МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»
Кафедра кадастра недвижимости, землеустройства и геодезии

БАЗЫ ГЕОПРОСТРАНСТВЕННЫХ ДАННЫХ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к проведению практических работ
для студентов, обучающихся по направлению подготовки:
05.04.03 Картография и геоинформатика
(программа: Геоинформационное моделирование)
всех форм обучения

Воронеж 2025

УДК 91
ББК 26.8я7

Составители
ст. преп. Васильчикова Е.В.

БАЗЫ ГЕОПРОСТРАНСТВЕННЫХ ДАННЫХ: методические указания для проведения практических работ для студентов, обучающихся по направлению подготовки: 05.04.03 Картография и геоинформатика (программа: Геоинформационное моделирование) всех форм обучения/ сост.: Е.В. Васильчикова; ВГТУ. – Воронеж, 2025. – 96с.

Методические указания разработаны для проведения практических работ по дисциплине «БАЗЫ ГЕОПРОСТРАНСТВЕННЫХ ДАННЫХ», обучающие компьютерным технологиях, которые используются для создания карт и оценки фактически существующих объектов.

Предназначены для студентов направления 05.04.03 Картография и геоинформатика (программа: Геоинформационное моделирование) всех форм обучения.

Методические указания подготовлены в электронном виде и содержатся в файле МУ_БГД_ПР.pdf.

УДК 91
ББК 26.8я7

Рецензент – Н.И. Самбулов, к.г.н., доц. кафедры кадастра недвижимости, землеустройства и геодезии ВГТУ

Оглавление

Практическая работа 1. Базовые операции с пространственными данными 4

Практическая работа 2. Интеграция данных различных типов и составления аналитических запросов в базе пространственных данных реляционного типа..13

Практическая работа 3. Построение ER-диаграмм для пространственных данных. 22

Практическая работа 4. Работа с пространственными запросами в SQL 34

Практическая работа 5. Интеграция данных различных типов в базу пространственных данных геоинформационных систем 38

Практическая работа 6. Анализ и визуализация данных в базе пространственных данных геоинформационных систем 61

Практическая работа 7. Анализ топологических отношений и пространственных данных 74

Практическаяработа 8. Создание картографической продукции на основе базы данных 89

Список литературы 95

Практическая работа 1. Базовые операции с пространственными данными

Цель работы — научиться рассчитывать соотношение различных явлений на регулярной сетке с использованием векторного оверлея.

Необходимая теоретическая подготовка: Оверлей пространственных объектов, геометрическое определение вероятности как отношения мер (площадей), соединение таблиц в реляционных базах данных, внешний и внутренний ключ соединения.

Необходимая практическая подготовка: Знание основных компонент интерфейса QGIS (менеджер источников данных, таблица слоёв, фрейм карты, менеджер компоновок). Работа с различными форматами источников пространственных данных. Настройка символики и подписей объектов. Владение базовыми ГИС-технологиями.

Исходные данные: Векторные контура типов земельного покрова, полученные на основе данных OpenStreetMap.

Результат: карта соотношения различных типов земельного покрова.

- Добавить на карту слои типов объектов и регулярную сетку, оформить их
- Произвести оверлей и слияние объектов в пределах ячеек
- Рассчитать площадь объектов
- Присоединить поля площади к таблице регулярной сетки и рассчитать площадь оставшихся объектов
- Визуализировать результат

Аннотация

Задание посвящено знакомству с пространственным анализом на основе векторных данных. Векторная модель представляет объекты в виде отдельных геометрических фигур с набором атрибутов. Она является объектно-ориентированной и удобна для анализа формы, размеров объектов, их взаимной конфигурации в пространстве. Одним из широко используемых методов анализа на основе векторных данных является оверлей.

При *оверлее* происходит наложение двух или более слоев, в результате чего образуется их геометрическая (пространственная) композиция. Полученные участки наследуют атрибуты от каждого слоя. Эта операция базируется на стандартных отношениях множеств, таких как пересечение, объединение и симметрическая разность.

С помощью оверлея можно, например, установить, как соотносятся площади объектов разных типов в пределах ячеек регулярной сетки. Это может быть важно при моделировании, например, локальных климатических зон или анализе экологической ситуации.

Визуальный анализ векторных слоев

В первую очередь при анализе данных следует провести их визуальную оценку, которая может натолкнуть на отыскание закономерностей во взаимном расположении объектов.

1. Распакуйте архив с материалами упражнения в свою рабочую директорию. Создайте проект QGIS в папке с распакованными материалами.

2. Добавьте на карту слои *Industrial*, *Hydro*, *Green*, *Buildings* из базы данных LandCover.gpkg. Присвойте этим слоям разные цвета для лучшего восприятия данных.

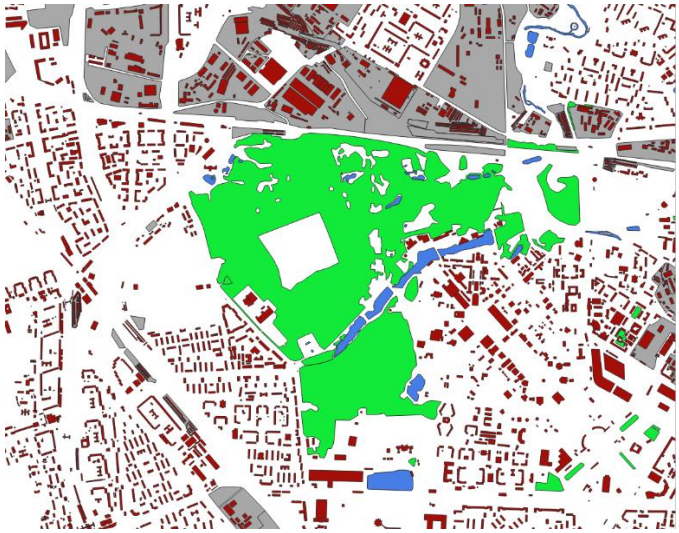


Рис. 1.1. Слои на карте в QGIS

3. Создайте регулярную сетку квадратов с помощью инструмента **Вектор – Анализ – Создать сетку…**. В открывшемся окне параметров инструмента выберите прямоугольный тип сетки, укажите вертикальный и горизонтальный шаг сетки в 1000 м, а в качестве экстента (охвата) слоя укажите существующий слой проекта *Buildings*. Сохраните сетку в отдельный файл Geopackage в вашей рабочей директории.

4. Измените стиль слоя сетки таким образом, чтобы отображались только границы полигонов. Установите толщину линии границы равной 0,26 мм. Переместите слой на верх таблицы слоёв. Результат должен выглядеть аналогично рисунку ниже.

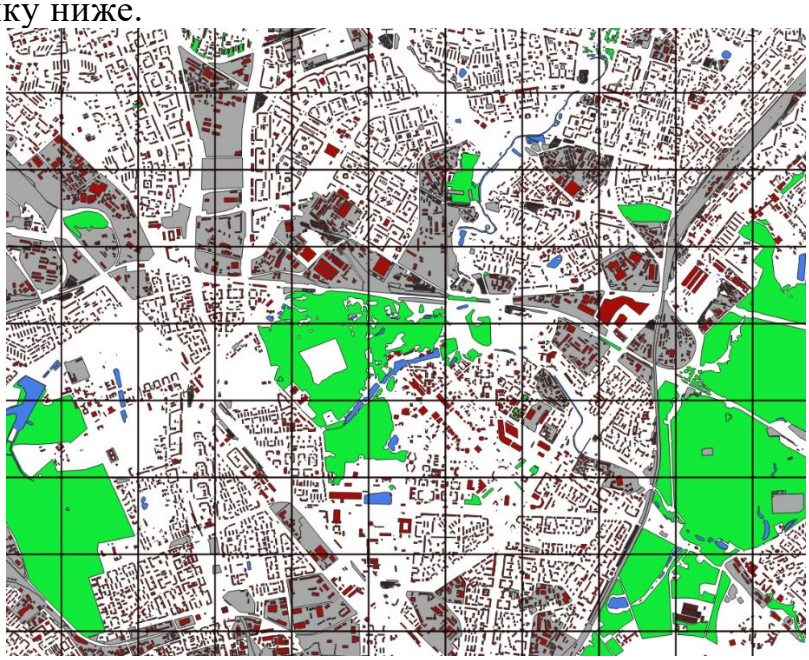


Рис. 1.2. Координатная сетка в проекте

5. Сделайте снимок экрана.

Расчёт площадей объектов

1. Для расчёта доли площади каждого типа объекта в пределах ячейки необходимо знать площадь самой ячейки. Для этого выберите слой в таблице слоёв, а затем нажмите кнопку **Калькулятор полей** . Эту кнопку можно найти на панели атрибутов либо непосредственно в таблице атрибутов.

2. В интерфейсе калькулятора полей укажите, что вы создаёте новое поле, введите имя поля CellArea и выберите подходящий тип поля.

3. Введите в форму выражения функцию *$area* и выполните расчёт.

Примечание: area() — системная функция QGIS, возвращающая площадь объекта. Значок $ означает, что функция будет применена к текущему объекту. Площадь вычисляется в единицах измерения площади, предусмотренных для системы координат источника данных. Для прямоугольных систем координат это, как правило, метры.

4. При использовании Калькулятора полей автоматически включается режим редактирования. После расчёта площади сохраните изменения и выключите режим редактирования.

5. Откройте таблицу атрибутов слоя сетки и сделайте снимок экрана.

После того, как вы сделаете снимок экрана, закройте окно таблицы атрибутов.

Оверлей векторных наборов данных

1. Для того чтобы рассчитать площади каждого типа объекта в пределах ячеек регулярной сетки, необходимо выполнить операцию пересечения (оверлея) слоев. Для этого воспользуйтесь инструментом **Вектор — Геообработка — Пересечение**. В качестве входного набора данных (*Input layer*) укажите регулярную сетку, а в качестве пересекающего набора данных (*Overlay layer*) — слой с растительностью. Результат сохраните во временный слой.

Временный слой в QGIS хранится в выделенной директории среди системных файлов. Если не сохранять временные файлы, они будут удалены после закрытия окна QGIS. Временные слои обозначаются значком справа от названия в таблице слоёв.

2. Результирующий слой будет называться *Пересечение* или *Intersect*. Переименуйте его так, чтобы было понятно, в результате пересечения каких именно данных он получился. Изучите, чем полученный набор данных отличается от исходных с точки зрения геометрии и семантики.

3. Повторите аналогичную операцию для оставшихся трёх слоёв.

4. После завершения всех операций увеличьте масштаб изображения так, чтобы в основной фрейм карты попадало 12-16 ячеек сетки, а также откройте таблицу атрибутов последнего из полученных слоёв. Расположите окно таблицы атрибутов так, чтобы оно было видно одновременно с геометрией объектов, и сделайте снимок экрана.

4. Обратите внимание, что в полученных слоях в пределах каждой ячейки мы видим, как правило, не один объект, а несколько. Мы можем

рассчитать площадь каждого из них, но нам нужна суммарная площадь всех объектов.

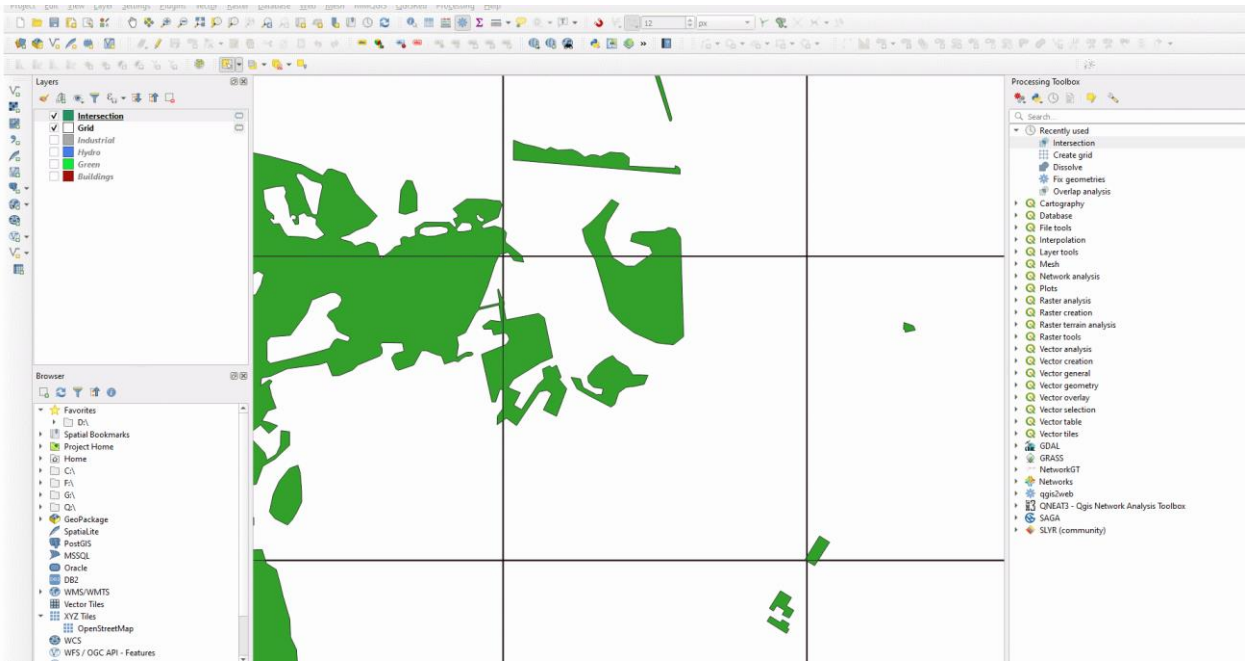


Рис. 1.3 Оверлей векторных наборов данных

Для получения общей площади можно объединить все объекты в пределах ячейки в один объект. Этого можно добиться с помощью операциии **объединения по признаку** (англ. *Dissolve*). Суть этой операции состоит в объединении объектов, имеющих совпадающие значения какого-либо атрибута (или нескольких атрибутов). После оверлея полигонов у каждого объекта растительности появился уникальный идентификатор ячейки регулярной сетки (id), его можно использовать для объединения по признаку.

5. Запустите инструмент **Вектор – Геообработка – Объединение по признаку…**. В качестве параметров инструмента необходимо указать исходный слой (возьмите пересечение сетки и растительности), а также столбец (или столбцы) таблицы атрибутов, по которому будет осуществляться слияние («поле классификации»). Результат сохраните в тот же файл GeoPackage, в который записана регулярная сетка; задайте имя таблицы green_intersect_dissolve

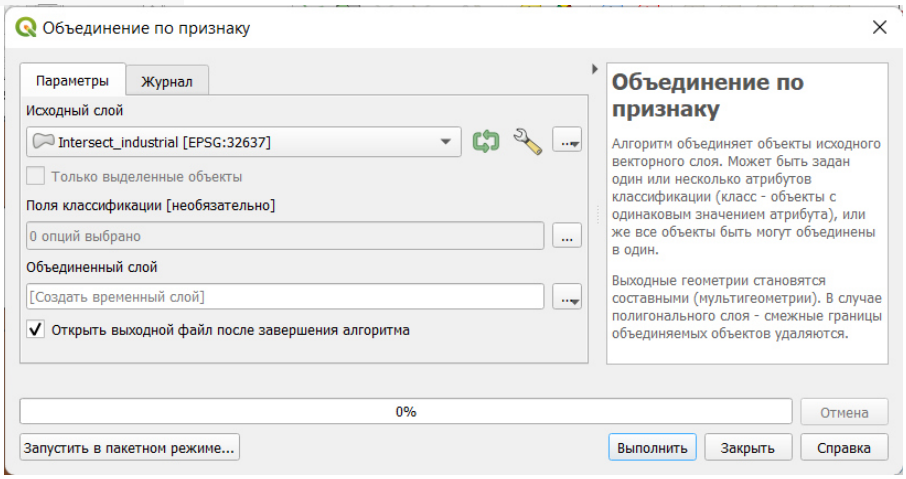


Рис. 1.4. Объединение по признаку

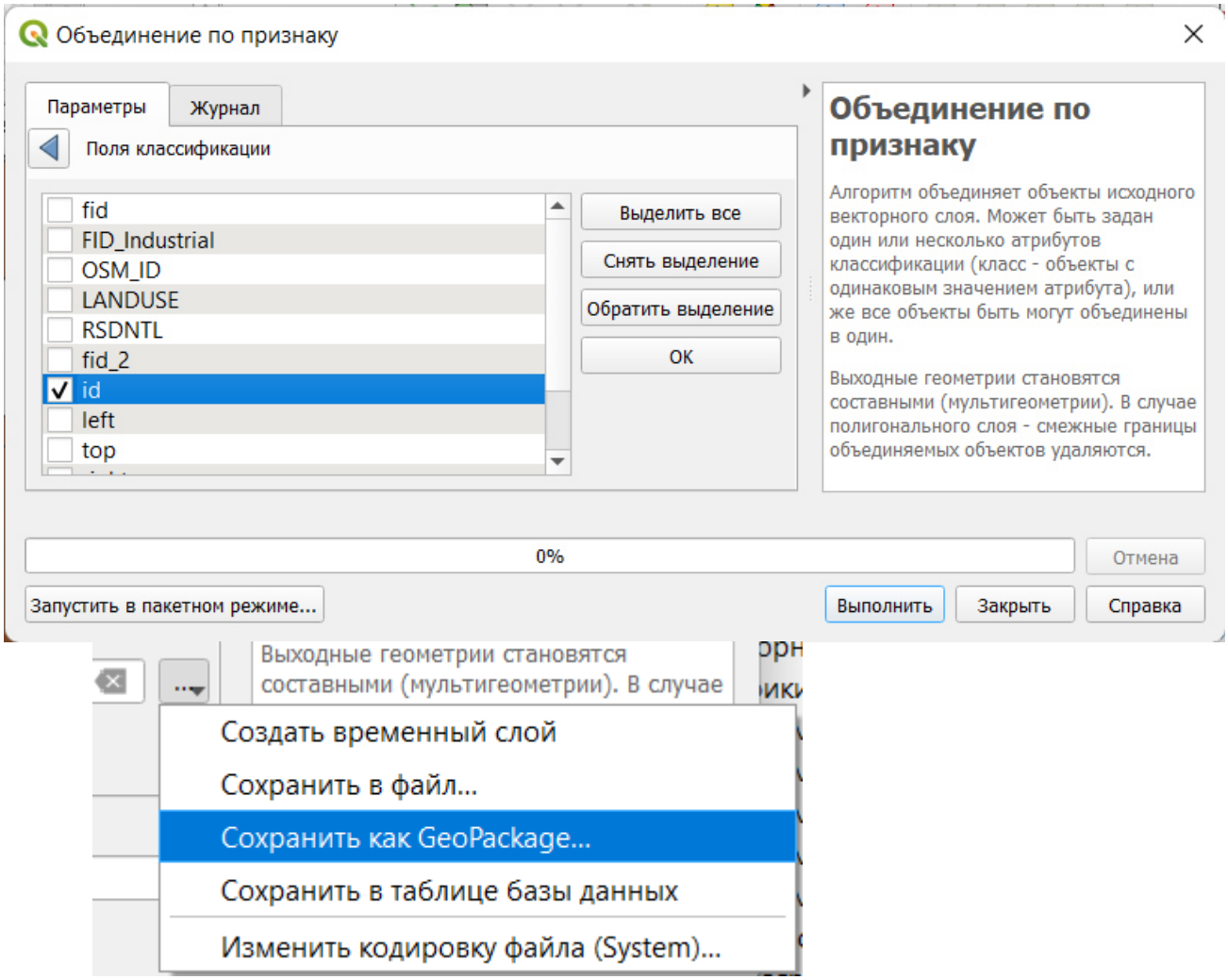


Рис. 1.5. Объединение по признаку id и сохранение файла

Результирующий временный слой будет называться *Объединенный слой* или *Dissolved* – переименуйте его так, чтобы было понятно, о каком типе объектов идёт речь.

5. Повторите аналогичную операцию для оставшихся трёх слоёв.

Подсказка: можно не закрывать интерфейс инструмента объединения по признаку, а просто менять необходимые параметры и нажимать «Выполнить»

Примечание: из-за ошибок в коде QGIS в некоторых случаях объединение по признаку может срабатывать некорректно: в целевой набор не записываются объекты, и таблица атрибутов не содержит записей. Если вы столкнётесь с такой проблемой, сохраните объединенный слой в новый файл GeoPackage или во временный слой.

6. Рассчитайте площади всех объектов в каждом объединённом слое с помощью калькулятора полей. Используйте осмысленные имена полей, различные для разных наборов данных (например, buildings_area).

7. Откройте таблицу атрибутов любого из полученных на этом этапе слоёв и сделайте снимок экрана.

Соединение таблиц по ключевому полю

На данном этапе у вас должны быть 4 слоя с известными площадями объектов в ячейках и собственно слой ячеек, для которых рассчитана площадь.

Для дальнейшего анализа нам необходимо соединить атрибутивные таблицы всех этих слоёв.

Соединение таблиц (table join) — операция, в результате которой к одной таблице временно добавляются столбцы из другой таблицы. Чтобы установить соответствие между строками целевой и присоединяемой таблицы, необходимо иметь в каждой таблице поля с общими для них значениями — так называемые *ключевые поля*. Например, это может быть числовой код объекта (как в нашем случае) или короткий текст (например, трёхбуквенный код единицы АТД по ISO). При соединении проверяется соответствие значения из поля для объединения и целевого поля. Если найдено совпадение, то строка из присоединяемой таблицы добавляется к атрибутам соответствующих объектов целевой таблицы. Если совпадения нет, то столбцы присоединяемой таблицы всё равно добавляются в целевую таблицу, но все значения в них будут пустыми (*NULL*).

Соединение в QGIS имеет реализуется по схеме M:1. Это значит, что одна и та же строка присоединяемой таблицы может быть добавлена ко многим строкам целевой таблицы, в то время как к одной строке целевой таблицы присоединяется только одна строка присоединяемой таблицы. Ключевые поля такого соединения имеют собственные специальные названия: ключевое поле из присоединяемой таблицы называется *первичным ключом*, а ключевое поле из целевой таблицы называется *внешним ключом*. На первичный ключ накладывается следующее ограничение: значения в этом поле должны быть уникальными. На внешний ключ такое ограничение не накладывается.

1. Откройте свойства слоя регулярной сетки. Перейдите на вкладку **Связи** (*Joins*).
2. Чтобы добавить новое соединение, нужно нажать кнопку «+» внизу. Откроется окно, в котором нужно выбрать слой, атрибутивную таблицу которого мы хотим присоединить, а также ключевые поля, хранящие общие значения – в нашем случае это поля *id*.
3. Установите галочку напротив опции **Присоединенные поля** (*Joined fields*), чтобы выбрать поля, которые нужно присоединить. Нам нужна только площадь объектов. Кроме того, можно включить **Пользовательский префикс имени поля** и удалить все символы из формы.

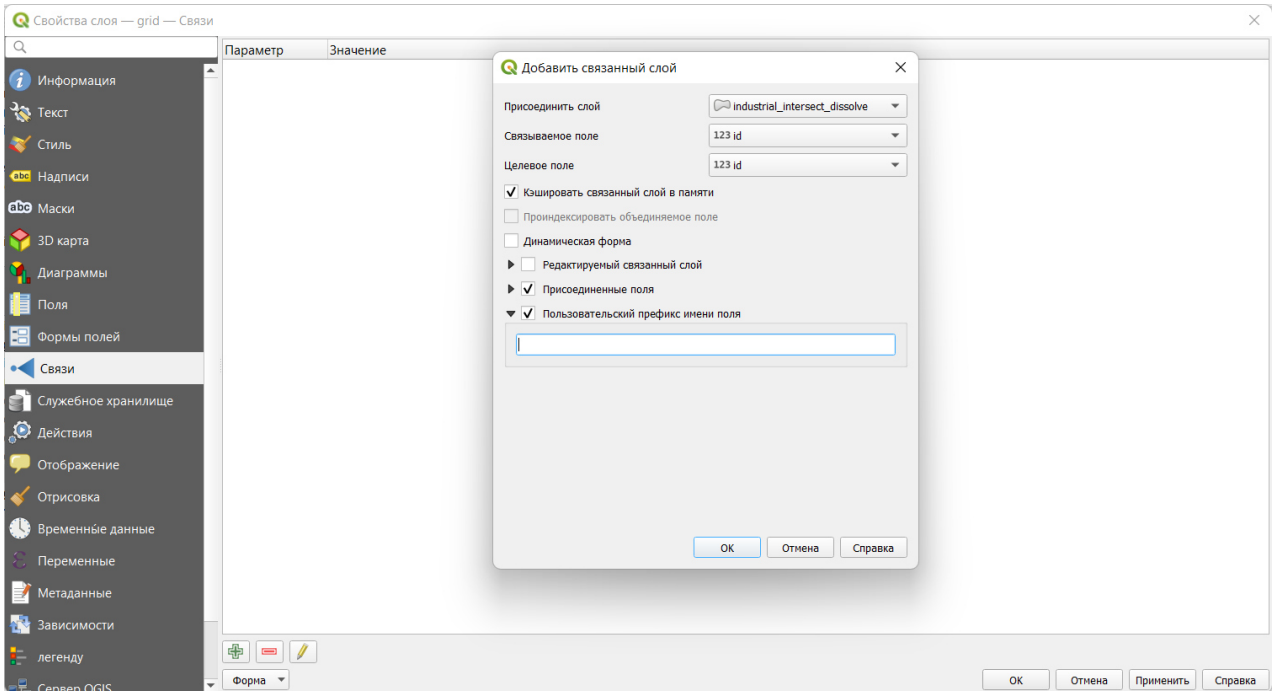


Рис. 1.6. Свойства связанного слоя

4. Проделайте аналогичную операцию для остальных трёх слоёв.

5. Ответьте на вопрос в бланке отчёта.

В отчёт: вопрос №1 — Что такое «поле для объединения» и «целевое поле» в QGIS? Как правильно называются эти поля в процедуре соединения таблиц? К каким таблицам (слоям) относится каждое из них?

5. После выполнения соединения откройте таблицу атрибутов регулярной сетки и прокрутите таблицу вправо таким образом, чтобы были видны присоединённые поля. Сделайте снимок экрана.

Расчёт площадей, не покрытых объектами

Четыре типа объектов покрывают далеко не всю территорию, поскольку есть и иные объекты, которые в нашем случае не берутся в расчёт. Для корректного анализа и визуализации необходимо рассчитать оставшуюся площадь для каждой ячейки. Это можно сделать, отняв из площади ячейки площади всех объектов каждого типа. Проще всего для такой задачи воспользоваться калькулятором поля. Однако здесь возникает проблема: в атрибутивной таблице встречаются пустые значения (NULL) в полях площадей, которые обозначают отсутствие данных. Такое значение получается в случае, если в ячейке регулярной сетки отсутствует соответствующий тип объектов. Со значением NULL мы не сможем корректно рассчитать оставшуюся площадь, поэтому необходимо заменить его на 00. Для этого воспользуемся запросами к атрибутивной таблице и калькулятором полей.

1. Сохраните слой регулярной сетки с присоединенными полями как новый набор объектов (GeoPackage), щёлкнув правой кнопкой мыши по нему и выбрав соответствующий пункт меню. Это нужно, чтобы вы не изменяли данные в исходных слоях.

2. Запустите калькулятор полей для вновь созданного набора данных.

3. В открывшемся окне укажите, что нужно не создавать новое поле, а обновить существующее. Выберите одно из полей, содержащих сведения о площади покрытий (зданиями, растительностью и др.)

4. В калькуляторе полей разверните блок **Условные** и найдите функцию if. Изучите справку к этой функции.

5. Составьте выражение, которое будет выдавать значение 0, если исходное значение поля — NULL, и исходное значение во всех остальных случаях. **Важно**: вы не можете использовать оператор = вместе со значением NULL, вместо = используйте оператор IS. Чтобы использовать названия полей в выражении, найдите в средней панели группу «Поля и значения». Добавляйте поля в выражение, кликая по их названиям дважды левой кнопкой мыши.

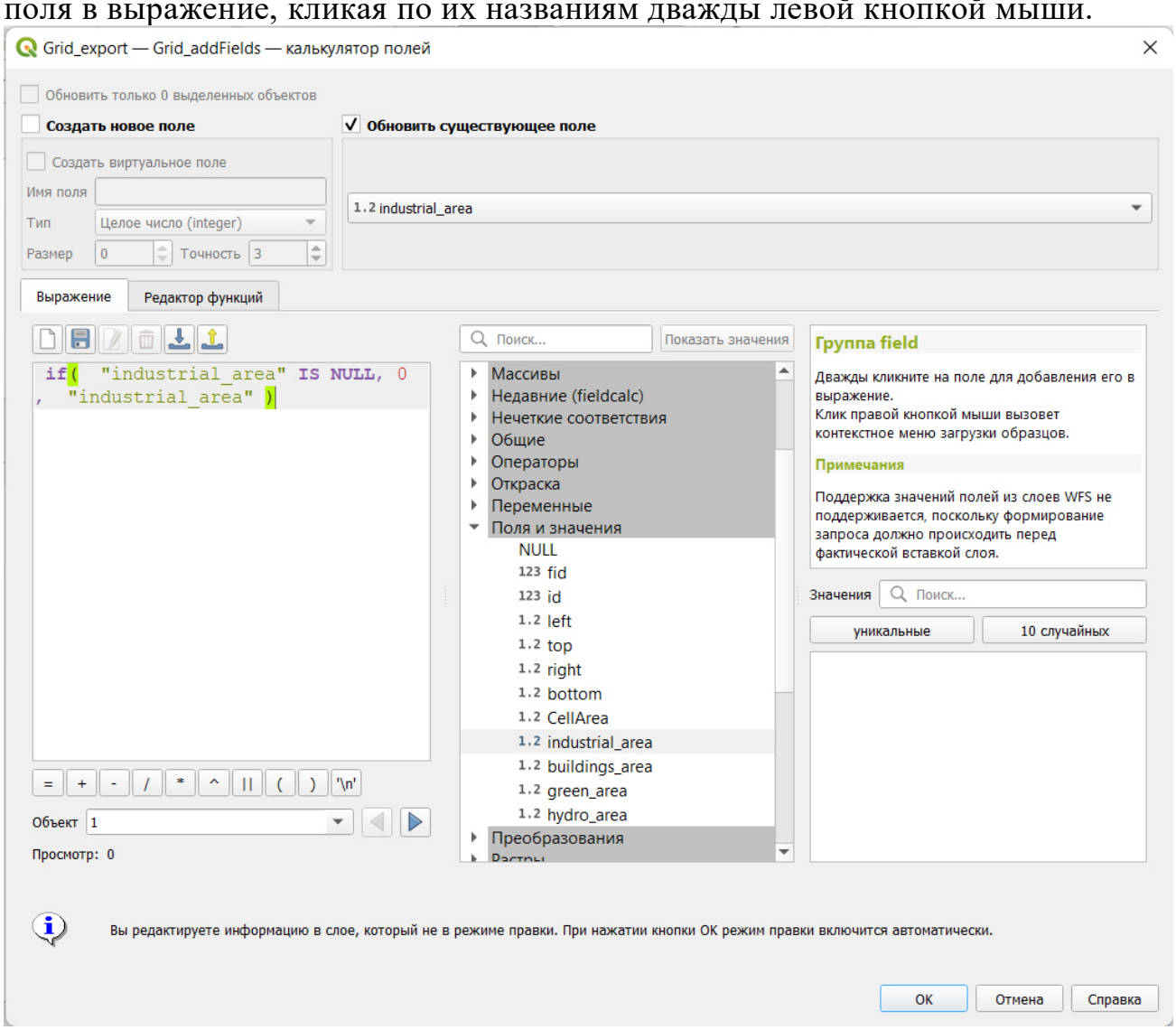


Рис. 1.7. Калькулятор полей

Если выражение составлено корректно, значения NULL будут заменены на 0

5. Проделайте аналогичную последовательность действий с выборкой и калькулятором полей для оставшихся трёх типов объектов.

6. Теперь используйте калькулятор полей, чтобы рассчитать площадь ячейки, не покрытую ни одним из типов объектов. Составьте выражение самостоятельно аналогично представленному на скриншоте ниже.

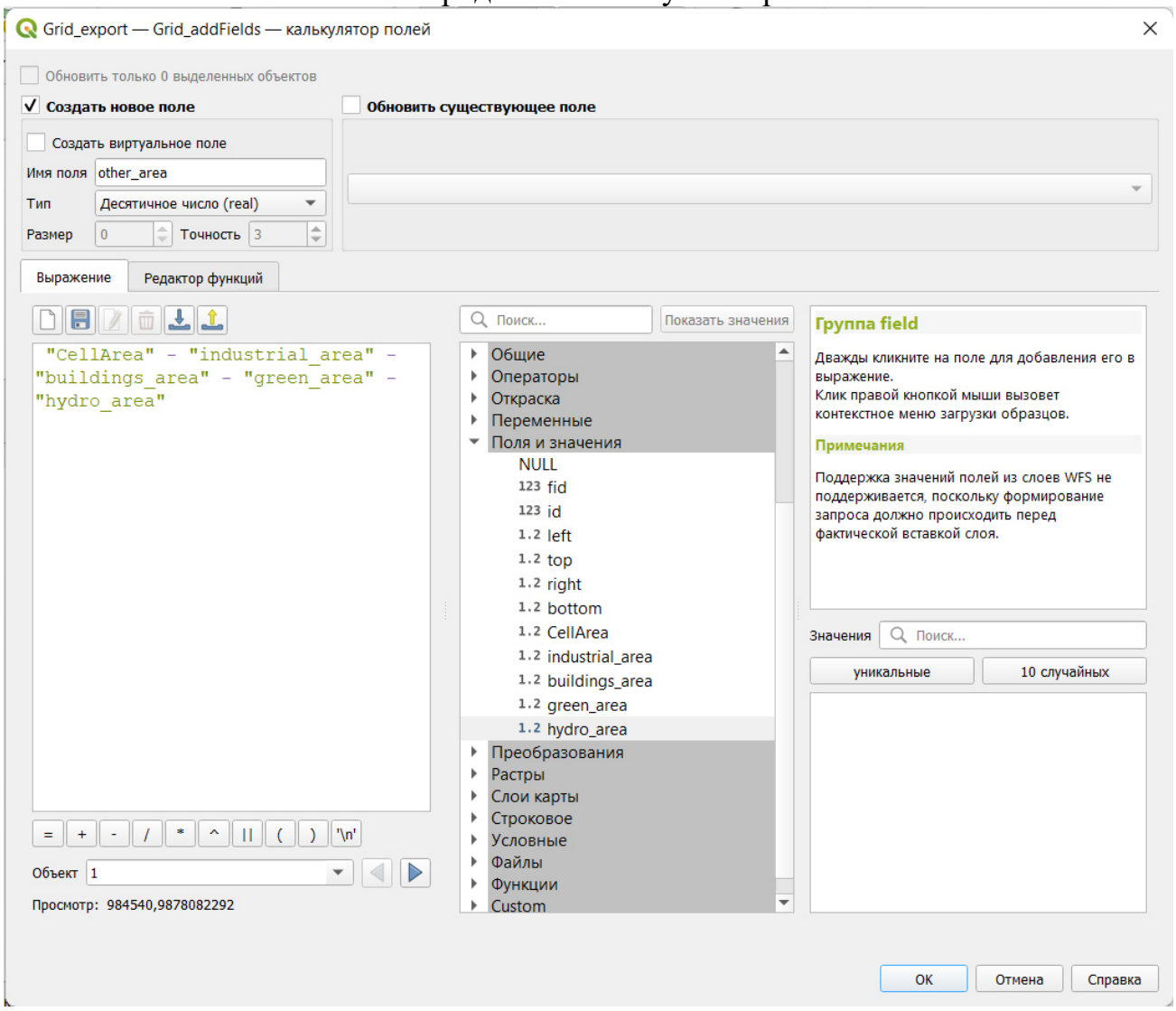


Рис. 1.8. Калькулятор полей

7. Выключите режим редактирования, сделайте снимок экрана с таблицей атрибутов и закройте окно таблицы атрибутов

Настройка символики

1. Визуализируйте доли покрытий с помощью картодиаграмм.

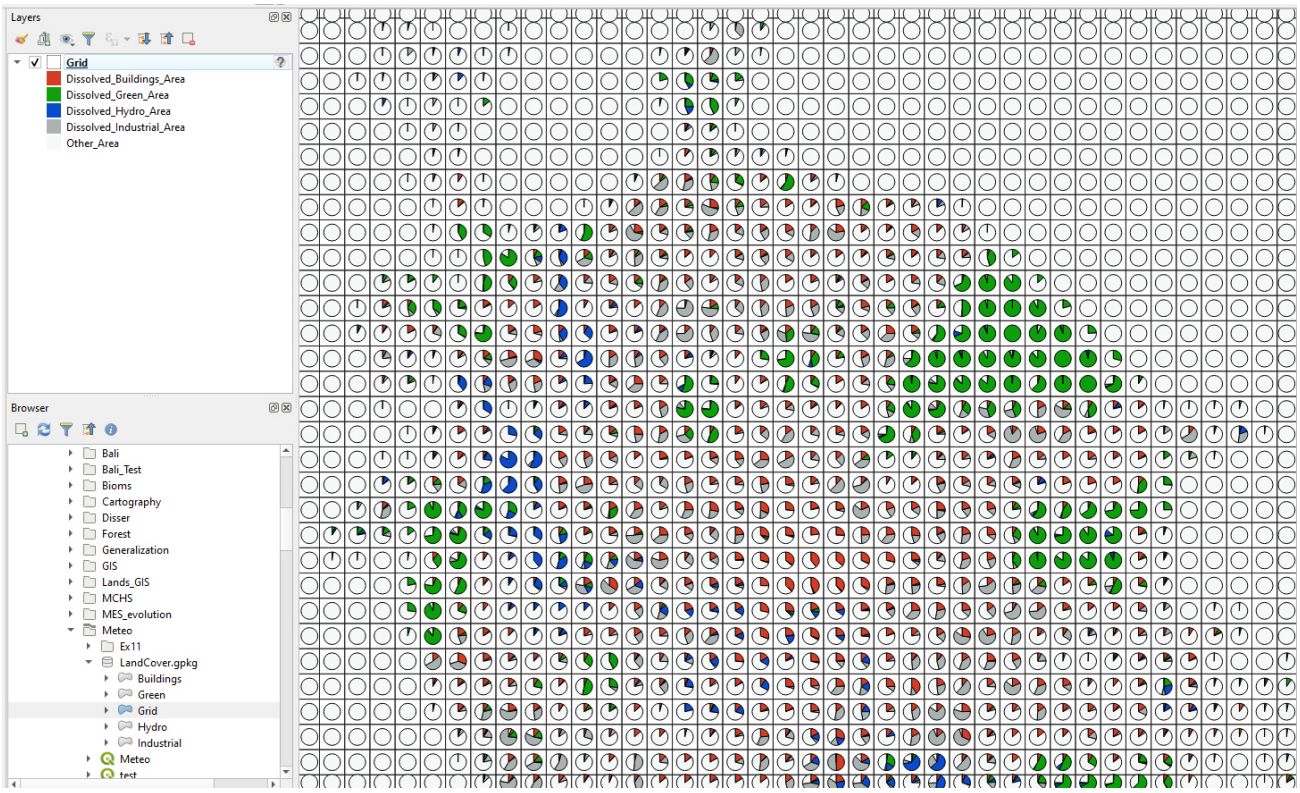


Рис. 1.9. Картодиаграммы

2. Оформите фрагмент вашего набора данных в виде картографического изображения. Вставьте полученную карту в отчётный файл.

Практическая работа 2. Интеграция данных различных типов и составления аналитических запросов в базе пространственных данных реляционного типа

Цель работы: усвоить основные принципы интеграции данных различных форматов, основные типы запросов к данным, освоить принципы связывания таблиц и фильтрации данных инструментами геоинформационной системы.

Задачи работы

1. Изучить:
- распространенные форматы данных для геоинформационных систем;
- алгоритм выполнения запросов к данным.
2. Научиться:
- работать с основными форматами данных геоинформационных систем;
- выполнять связывание данных из разных таблиц;
- выполнять пространственные запросы.

Перечень обеспечивающих средств

Для выполнения работы необходимо иметь методические указания по выполнению работы, а также программное обеспечение QGIS (версия 3.10 и выше).

Открытие таблиц в ГИС QGIS

Набор данных для выполнения заданий представляет собой пространственные данные о странах и городах мира, а также различную

статистическую информацию. Исходные данные представлены в форматах, которыми может оперировать ГИС QGIS.

Исходные данные о границах стран мира в виде площадных объектов представлены в стандартном формате хранения данных ГИС.

Для открытия карты с административно-территориальным делением стран мира (Countries) нужно в меню *Слой* выбрать пункты *Добавить слой - Векторный слой*. В открывшемся диалоге указать файл Countries.TAB и нажать кнопку *Добавить*. В результате будет открыто окно карты с площадными объектами (рис. 2.1). Семантическая информация об объектах представлена в таблице (правой кнопкой мыши нажать по слою и выбрать пункт меню *Открыть таблицу атрибутов*) (рис. 2.2).

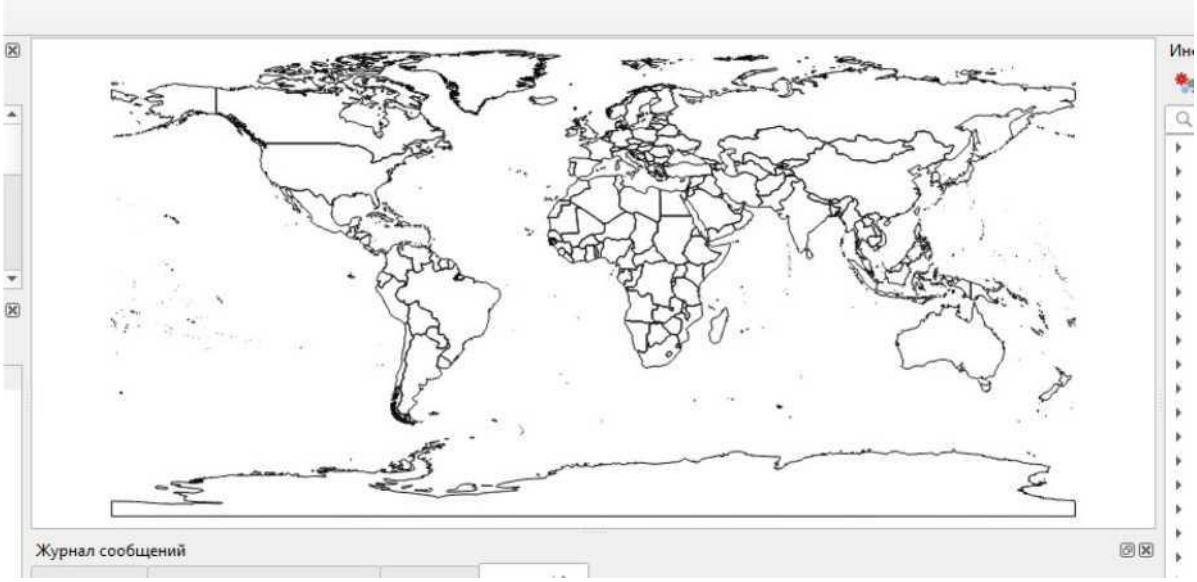
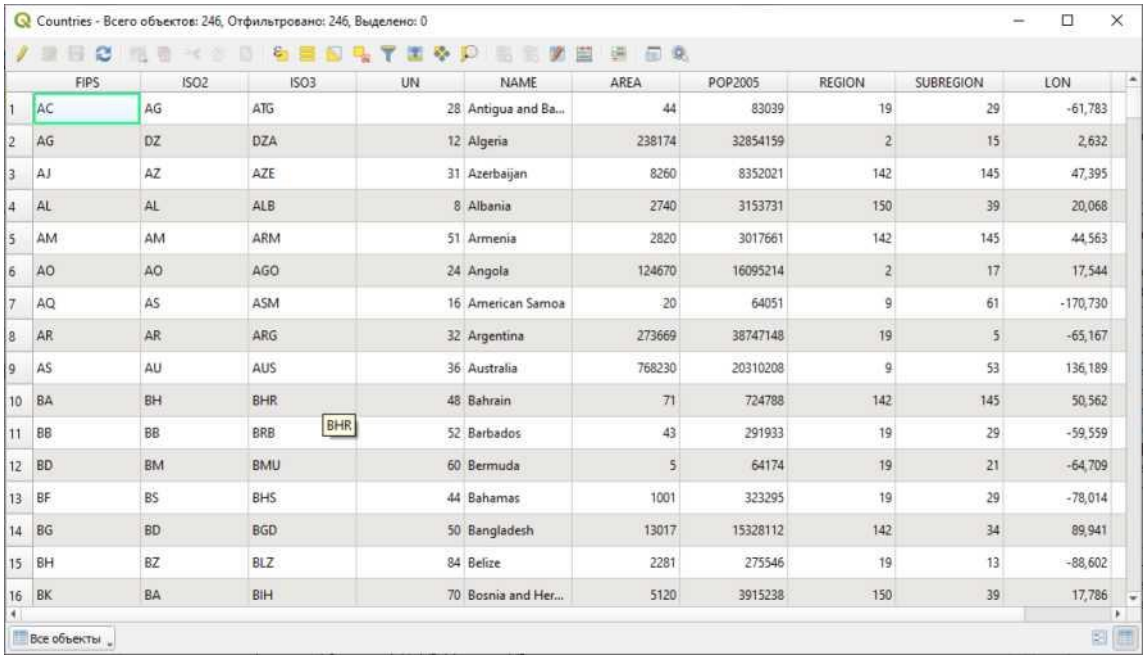


Рис. 2.1. Исходная политико-административная карта мира



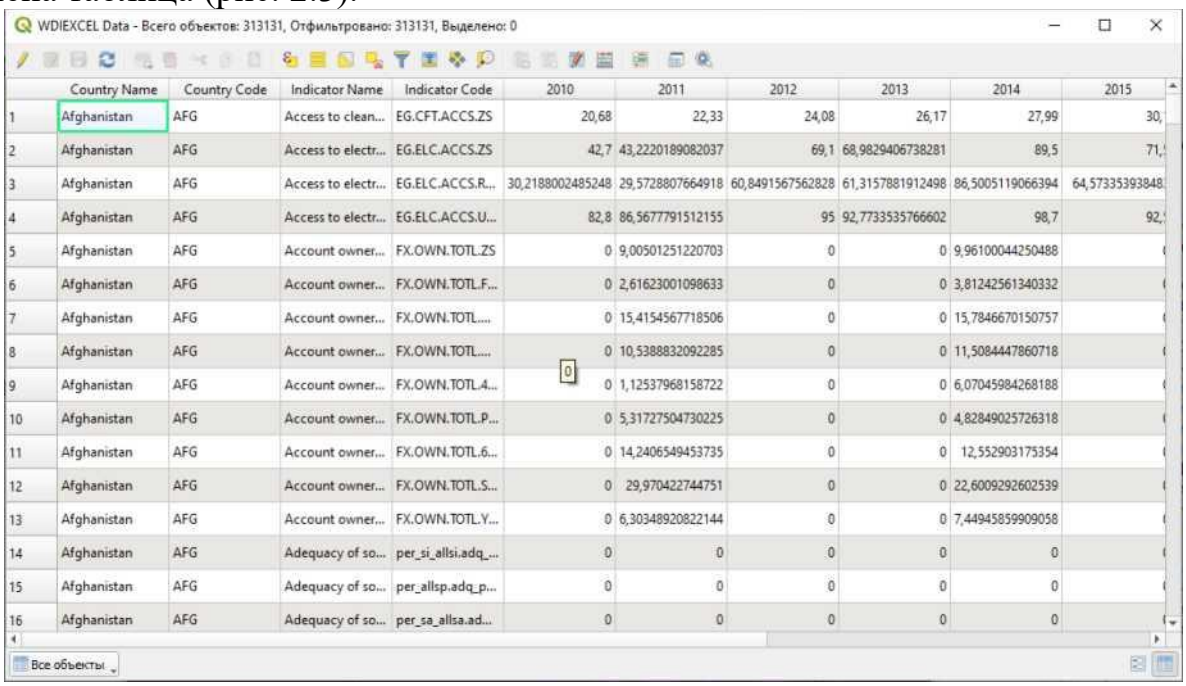
	FIPS	ISO2	ISO3	UN	NAME	AREA	POP2005	REGION	SUBREGION	LON
1	AC	AG	ATG	28	Antigua and Ba...	44	83039	19	29	-61,783
2	AG	DZ	DZA	12	Algeria	238174	32854159	2	15	2,632
3	AJ	AZ	AZE	31	Azerbaijan	8260	8352021	142	145	47,395
4	AL	AL	ALB	8	Albania	2740	3153731	150	39	20,068
5	AM	AM	ARM	51	Armenia	2820	3017661	142	145	44,563
6	AO	AO	AGO	24	Angola	124670	16095214	2	17	17,544
7	AQ	AS	ASM	16	American Samoa	20	64051	9	61	-170,730
8	AR	AR	ARG	32	Argentina	273669	38747148	19	5	-65,167
9	AS	AU	AUS	36	Australia	768230	20310208	9	53	136,189
10	BA	BH	BHR	48	Bahrain	71	724788	142	145	50,562
11	BB	BB	BRB	52	Barbados	43	291933	19	29	-59,559
12	BD	BM	BMU	60	Bermuda	5	64174	19	21	-64,709
13	BF	BS	BHS	44	Bahamas	1001	323295	19	29	-78,014
14	BG	BD	BGD	50	Bangladesh	13017	15328112	142	34	89,941
15	BH	BZ	BLZ	84	Belize	2281	275546	19	13	-88,602
16	BK	BA	BIH	70	Bosnia and Her...	5120	3915238	150	39	17,786

Рис. 2.2. Семантические данные политико-административной карты мира

Импорт данных в ГИС из Excel

Одним из распространенных источников статистической информации являются файлы Microsoft Excel. Их добавление в виде таблиц с данными также осуществляется через меню *Добавить слой - Векторный слой* и указания типа файла *Файлы Excel* (*.xls, *.xlsx). Импорт выполняется на примере файла *WDI.xlsx* (лист *Data*), содержащего ежегодные значения различных социально-экономических индикаторов стран мира.

При открытии файлов Excel выводится диалог, в котором можно указать один или несколько листов с интересующими данными. В результате будет получена таблица (рис. 2.3).



	Country Name	Country Code	Indicator Name	Indicator Code	2010	2011	2012	2013	2014	2015
1	Afghanistan	AFG	Access to clean...	EG.CFT.ACCS.ZS	20,68	22,33	24,08	26,17	27,99	30,
2	Afghanistan	AFG	Access to electr...	EG.ELC.ACCS.ZS	42,7	43,2220189082037	69,1	68,9829406738281	89,5	71,
3	Afghanistan	AFG	Access to electr...	EG.ELC.ACCS.R...	30,2188002485248	29,5728807664918	60,8491567562828	61,3157881912498	86,5005119066394	64,57335393848
4	Afghanistan	AFG	Access to electr...	EG.ELC.ACCS.U...	82,8	86,5677791512155	95	92,7733535766602	98,7	92,
5	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.ZS	0	9,00501251220703	0	0	9,96100044250488	
6	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.F...	0	2,61623001098633	0	0	3,81242561340332	
7	Afghanistan	AFG	Account owner...	FX.OWN.TOTL....	0	15,4154567718506	0	0	15,7846670150757	
8	Afghanistan	AFG	Account owner...	FX.OWN.TOTL....	0	10,5388832092285	0	0	11,5084447860718	
9	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.4...	0	1,12537968158722	0	0	6,07045984268188	
10	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.P...	0	5,31727504730225	0	0	4,82849025726318	
11	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.6...	0	14,2406549453735	0	0	12,552903175354	
12	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.S...	0	29,970422744751	0	0	22,6009292602539	
13	Afghanistan	AFG	Account owner...	FX.OWN.TOTL.Y...	0	6,30348920822144	0	0	7,44945859909058	
14	Afghanistan	AFG	Adequacy of so...	per_si_allsi.adq_...	0	0	0	0	0	
15	Afghanistan	AFG	Adequacy of so...	per_allsp.adq_p...	0	0	0	0	0	
16	Afghanistan	AFG	Adequacy of so...	per_sa_allsa.ad...	0	0	0	0	0	

Рис. 2.3. Просмотр импортированных данных из файла формата Excel

Импорт данных в ГИС из файлов *.csv

Формат *.csv сейчас является самым распространенным способом представления статистических данных. Для примера загрузим в QGIS csv- файл Countries_of_the_world, содержащий краткую сводную информацию по странам мира на текущий год. Для этого в меню *Слой* нужно выбрать пункты меню *Добавить слой - Добавить слой их текста с разделителями*. После выбора файла будет выдан диалог, позволяющий указать разделитель полей (если он отличается от стандартной запятой), кодировку, названия и типы данных колонок (рис. 2.4). Результат добавления таблицы в формате *.csv представлен на рис. 2.5.

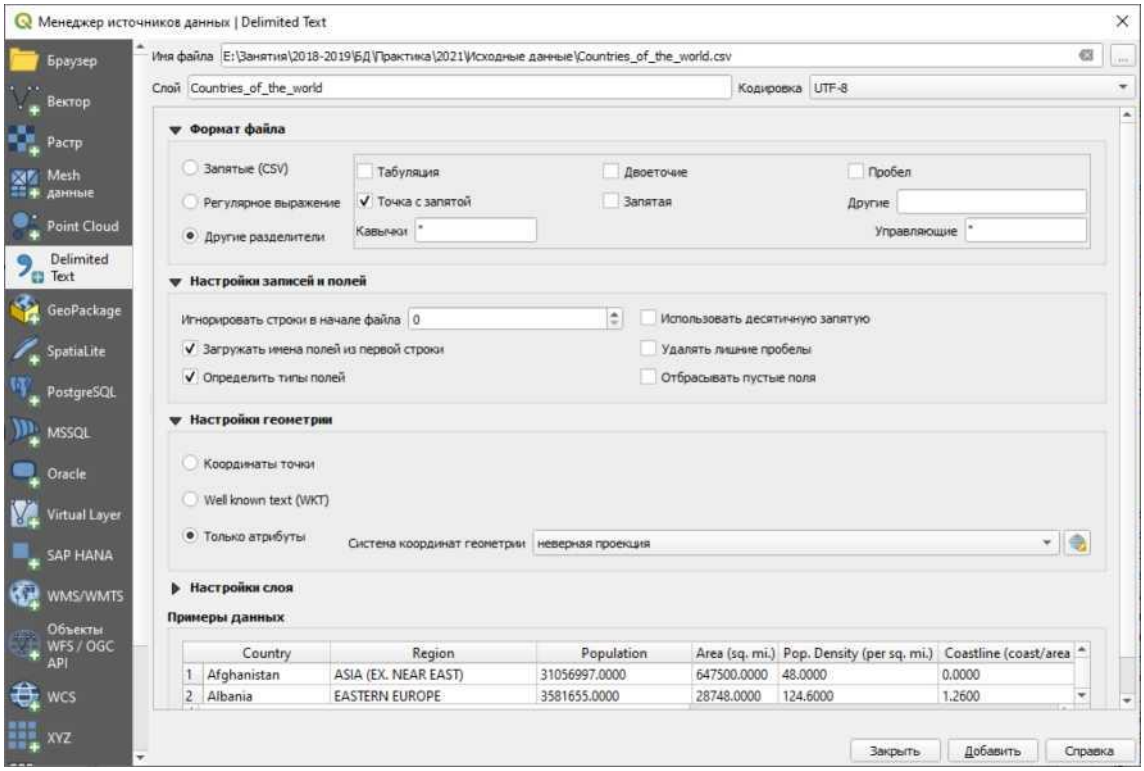


Рис. 2.4. Настройка параметров импорта csv-файла

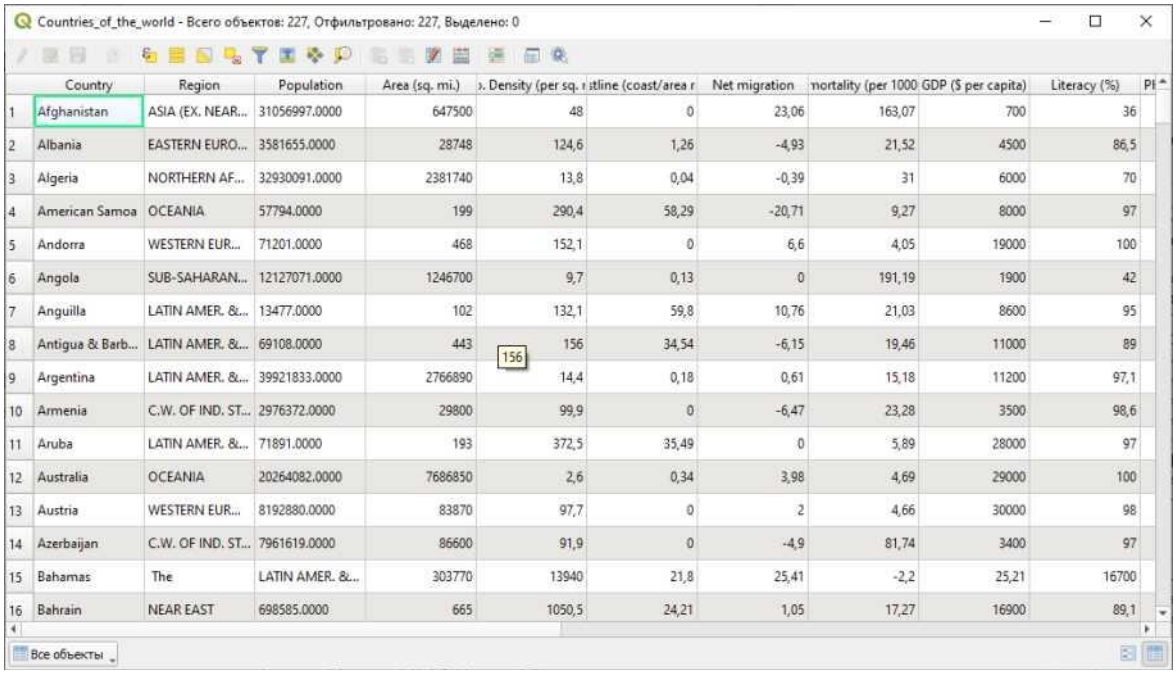


Рис. 2.5. Просмотр импортированных данных из csv-файла

Текстовые файлы могут содержать не только семантические данные, но и пространственные. В случае csv-файлов это, как правило, пары координат точечных объектов. QGIS позволяет создавать точечные объекты в импортированных таблицах на основе значений координат из определенных колонок. Рассмотрим порядок действий на примере файла, содержащего координаты и семантическую информацию наиболее крупных городов мира. Выберем для открытия csv-файл Cities.

В диалоге импорта файла с данными о городах нужно установить следующие параметры: разделитель - запятая, загружать имена полей из первой строки, координаты *X* и *Y* объектов из полей (с помощью выпадающих списков), содержащих значения широты и долготы, система координат - «WGS 84» (рис. 2.6).

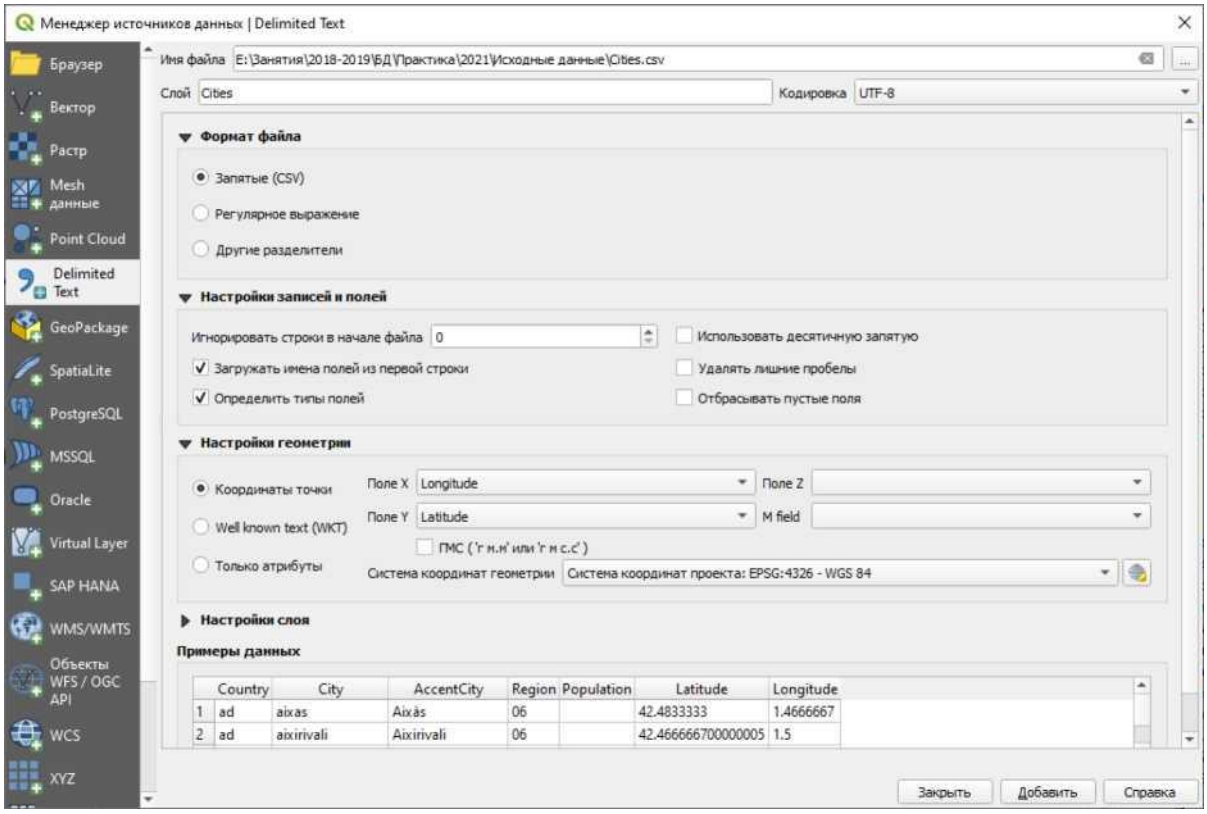


Рис. 2.6. Настройка параметров импорта csv-файла с данными о городах мира

Импорт данных в ГИС из SQLite

ГИС QGIS позволяет работать с наиболее распространенными файловыми базами данных и сетевыми СУБД, в том числе поддерживающими хранение и анализ пространственных данных. Для примера добавим информацию из файла открытого формата файловых баз данных - SQLite, содержащего статистические данные о демографии стран. В этом диалоге рекомендуется установить параметр *LIST_ALL_TABLE* в значение *Да*, чтобы получить список возможных к загрузке таблиц. В этом списке нужно выделить только таблицу с названием *Data*, остальные являются служебными (рис. 2.7).

Рис. 2.7. Диалог настройки параметров импорта данных из файла формата SQLite

Интеграция данных с картой

Поскольку пространственные данные в виде контуров стран мира и статистические данные из импортированных таблиц представляют собой различные источники, то практически всегда возникает необходимость объединения данных. Для осуществления этой операции необходимым является наличие в обеих таблицах колонок с одинаковыми данными; в случае представленных файлов такими колонками будут названия стран или их ISO-коды.

Для объединения таблиц по условию в QGIS используется механизм связей. Параметры связывания таблиц устанавливаются в свойствах слоя, раздел *Связи*. На рисунке ниже представлен пример запроса, объединяющего таблицы *Countries* и *Demographics* с условием идентичности трехсимвольных кодов стран (рис. 2.8).

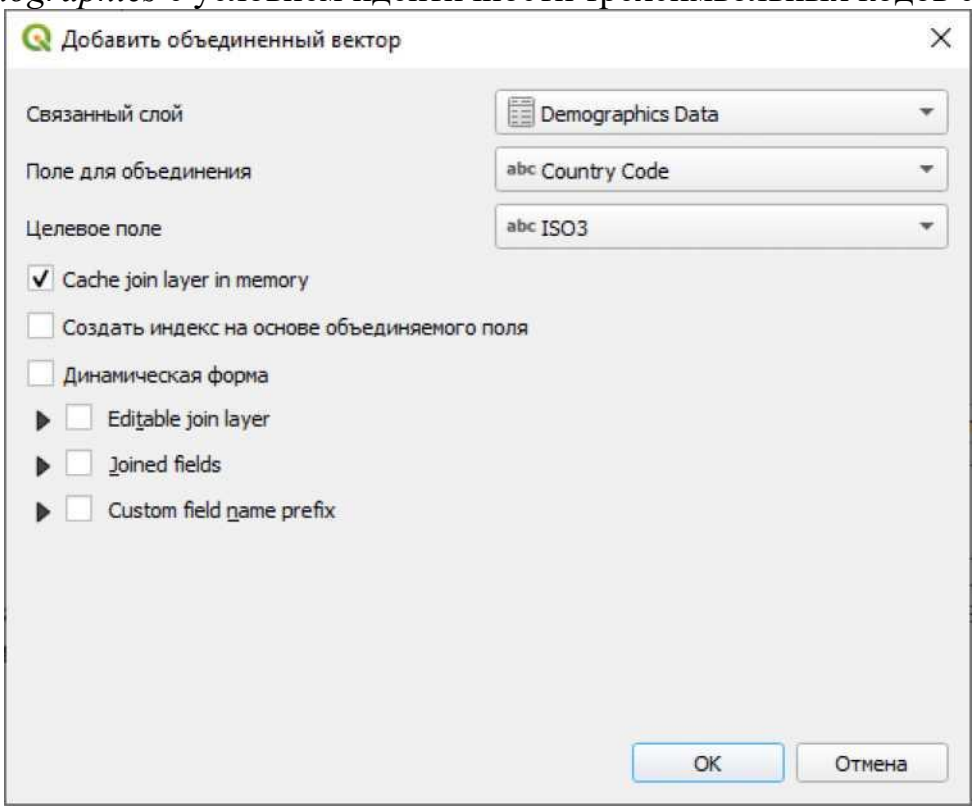


Рис. 2.8. Параметры связывания таблиц

В результате выполнения запроса на объединение таблиц будет выведена таблица, содержащая колонки из обеих таблиц исходных данных.

Запросы на выборку данных

Поскольку исходная таблица с городами содержит слишком много записей, то демонстрацию работы с запросами на выборку данных выполним на примере отбора населенных пунктов, лежащих на территории Российской Федерации с указанным (ненулевым) населением. Затем результаты этой выборки нужно будет выгрузить в отдельный файл.

Составим запрос на выборку всех населенных пунктов с ненулевым населением. Конструктор запросов вызывается кнопкой *Выделить объекты удовлетворяющие условию поиска*, находящейся в окне таблицы атрибутов слоя. Для отбора используется поле *Population (Население)*, показанное на рис. 2.9. Для последующего сравнения производительности СУБД и ГИС нужно засечь время выполнения запроса (в секундах) и внести в табл. 2.1.

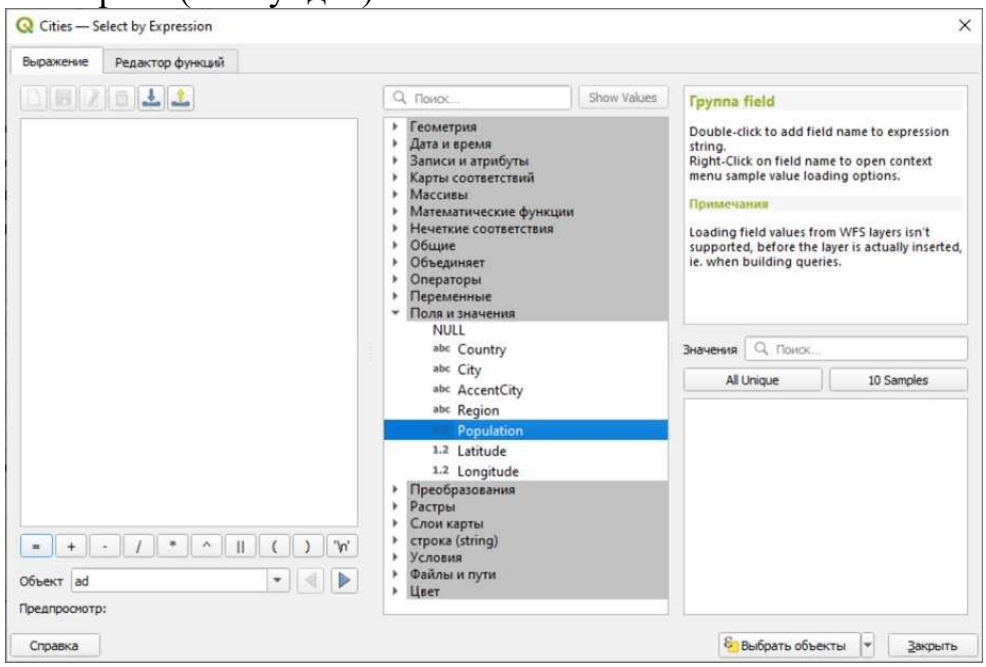


Рис. 2.9. Диалог отбора объектов по критерию

Таблица 2.1 – Статистика времени выполнения запроса

Используемое программное обеспечение	Время выполнения запроса на поиск городов с ненулевым населением, с
QGIS	

Созданную выборку нужно сохранить в отдельный файл (контекстное меню, вызываемое щелчком правой кнопки по слою, выбрать *Экспорт - Сохранить выделенные объекты*) (рис. 2.10).

Сохраняем результаты запросы в новом shape-файле *Cities_not_null* (рис. 2.11).

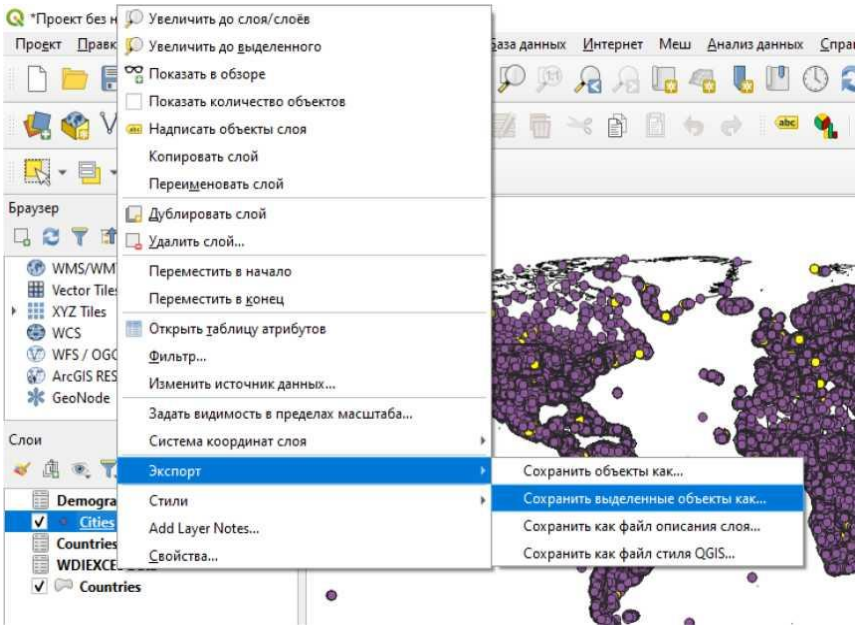


Рис. 2.10. Команда сохранения результатов выборки в отдельный файл

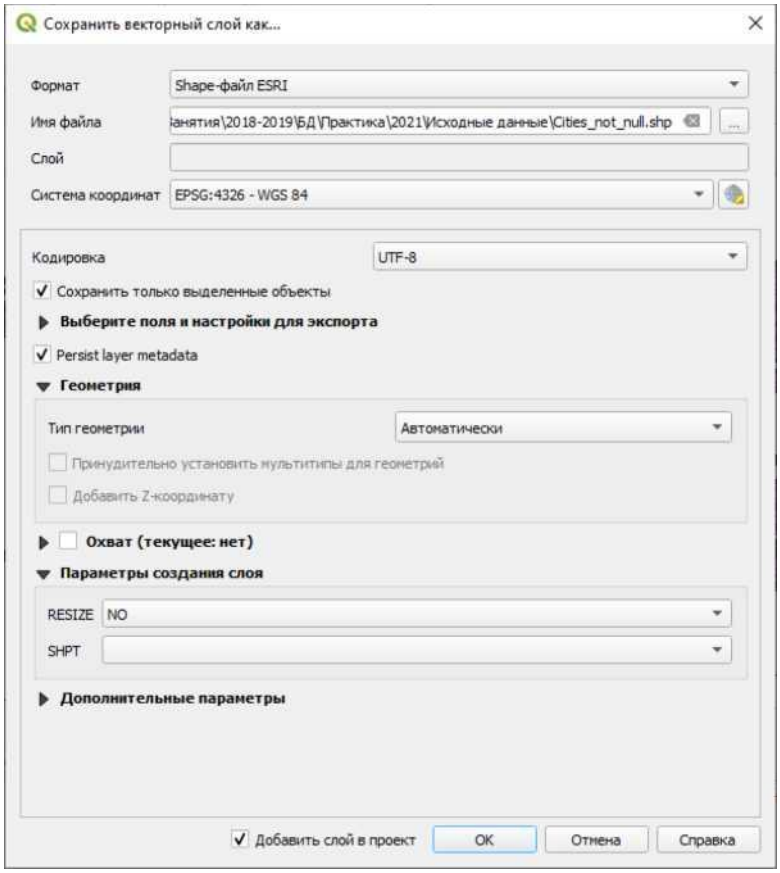


Рис. 2.11. Диалог указания параметров нового файла

Далее нужно закрыть исходную таблицу с городами и добавить созданную на предыдущем шаге (если это не было указано сразу в диалоге).

Запрос для выборки только тех городов, которые попадают на территорию Российской Федерации, формируется на основе пространственного отношения. Пространственный запрос находит объекты, полностью попадающие в указанные объекты одного из слоев (условие *are within*). Для осуществления выбора только внутри заданного полигона (из слоя *Countries*) нужно его предварительно

выделить (вручную либо через запрос по названию объекта) (рис. 2.12). Также для дальнейшего сравнения нужно засечь время выполнения запроса на выбор в границах полигона и внести результаты в табл. 2.2. Результаты выполнения запроса представлены ниже на рис. 2.13.

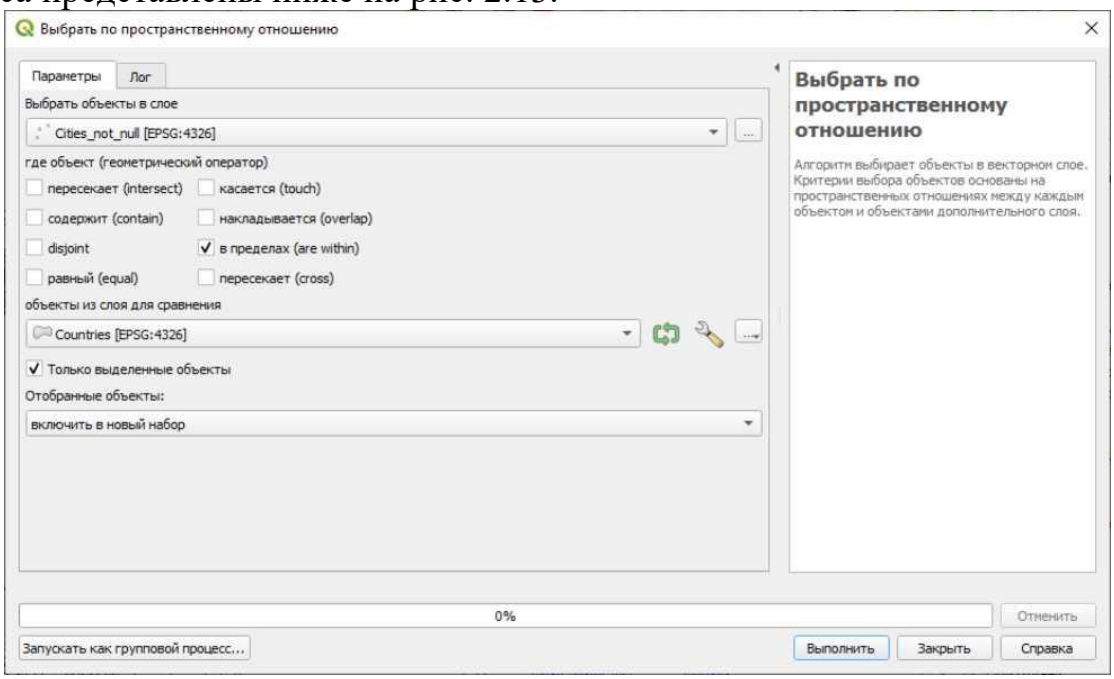


Рис. 2.12. Пространственный запрос на поиск точек внутри заданного полигона

Таблица 2.2 - Статистика времени выполнения запроса

Используемое программное обеспечение	Время выполнения запроса на поиск городов внутри указанной страны, с
QGIS	

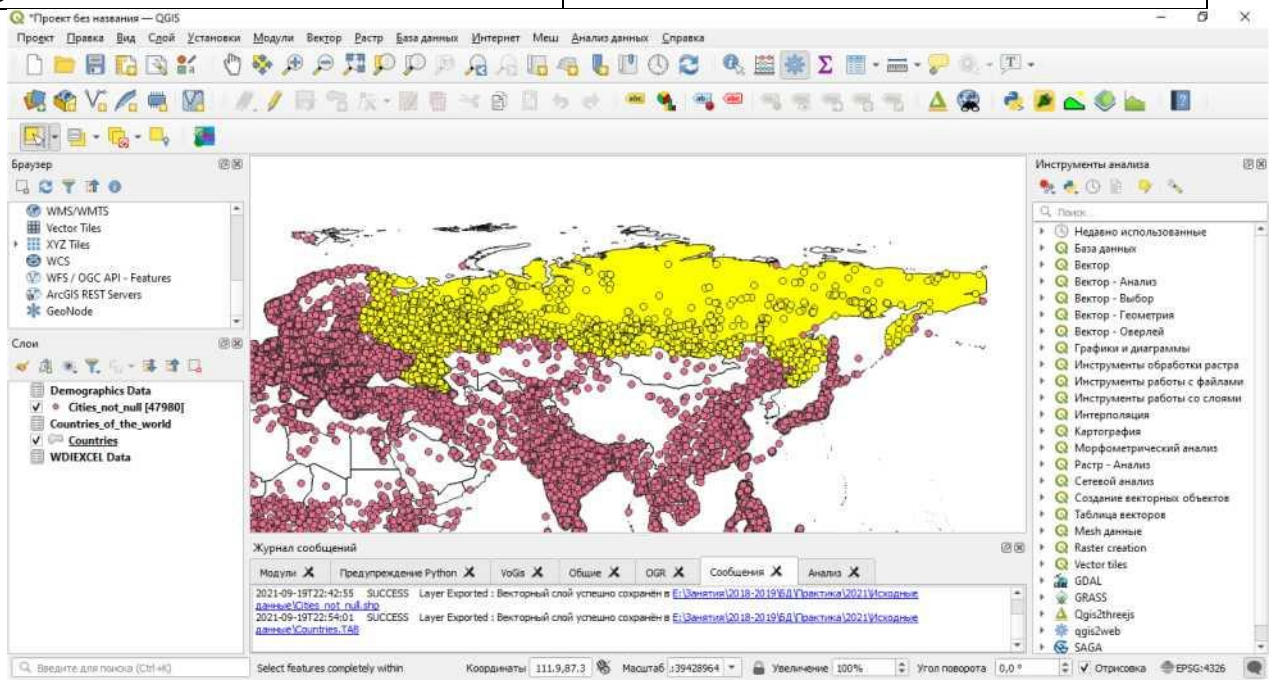


Рис. 2.13. Результат выборки

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Постановка задачи.
4. Описание выполненной работы со скриншотами и заполненными таблицами.

Практическая работа 3. Построение ER-диаграмм для пространственных данных.

Цель работы: усвоить основные принципы построения ER-диаграмм для пространственных данных

Задачи работы

1. Изучить:
- инфологическую модель данных;
- моделирование предметной области;
- ER-модель данных;
- CASE – средства;
- Цель инфологического проектирования предметной области;
- Модель «сущность-связь»;
- Построение ER-диаграмы.
2. Научиться:
- Строить ER-диаграмы

Наиболее распространенным средством моделирования данных являются диаграммы «сущность-связь» (ER). С помощью ER диаграмм осуществляется детализация хранимых данных, а также документируются информационные аспекты геоинофрмационной системы, включая идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их связей с другими объектами (отношений).

Рассмотрим классический способ построения моделей с точки зрения проектирования баз данных. При проектировании базы геоданных организацию данных рассматривают на трех уровнях: информационно-логическом (инфологическом), логическом и физическом. Этим уровням соответствуют этапы проектирования баз данных.

Инфологическая модель данных

Понятие о инфологической модели данных

Процесс проектирования базы данных длительный, требует обсуждения с заказчиком и специалистами в предметной области. При разработке корпоративных информационных систем проект базы данных является тем фундаментом, на котором строится вся система в целом.

Инфологическая модель должна включать такое формализованное описание предметной области, которое будет легко «читаться» как специалистами по базам данных, так и конечными пользователями, а также быть пригодной для оценки глубины и корректности работы проекта базы данных без

привязки к конкретной СУБД.

Проблема представления семантики данных является очень важной и актуальной для разработчиков информационных систем. Назначением инфологического моделирования является обеспечение возможности выражения семантики данных. Потребности проектировщиков баз данных в удобных средствах моделирования предметной области вызвали к жизни направление инфологического (семантического) моделирования данных. В семидесятых годах прошлого столетия были предложены несколько моделей данных, которые были названы информационно-логическими (семантическими) моделями.

Одной из наиболее популярных информационных моделей данных является модель «сущность-связь» или **ER-модель** (**Entity Relationship**). На использовании разновидностей ER-модели основано большинство современных подходов к проектированию реляционных баз данных. Модель была предложена Питером Ченом в 1976 году. В настоящее время модель Чена «сущность-связь» стала фактическим стандартом при инфологическом моделировании баз данных. Общепринятым стало сокращенное ее название - ER-модель.

Моделирование предметной области

Моделирование предметной области базируется на использовании графических диаграмм (нотаций) - ER-диаграмм (Entity Relationship Diagram), которые включают небольшое число разнородных компонентов. На основе анализа предметной области определяются основные объекты предметной области, строится схема базы данных в виде ER-диаграммы. На схеме в графической форме отображаются связи между объектами и характеристики этих связей. Затем по четким правилам осуществляется переход от ER-диаграмм к таблицам базы данных, наполнение таблиц атрибутами и проверка их на выполнение условий нормализации. Определяются ключевые атрибуты таблиц и связи между таблицами. Результатом проектирования является схема реляционной базы данных.

В связи с наглядностью представления концептуальных схем баз данных, ER-модели получили широкое распространение в CASE-системах, которые поддерживают автоматизированное проектирование реляционных баз данных. Большинство современных CASE-систем содержат инструментальные средства для описания данных в формализме модели «сущность-связь» и реализуют методы автоматического превращения проекта базы данных из ER-модели в реляционную модель. Затем выполняется превращение в логическую модель, которая отвечает конкретной СУБД. Примером CASE-системы является простая и универсальная программа **Erwin** фирмы PLATINUM, которая предназначена для автоматизированного создания реляционных баз данных.

Все CASE-системы имеют развитые средства документирования процесса разработки базы данных, автоматические генераторы отчетов позволяют подготовить отчет о текущем состоянии проекта базы данных с подробным описанием объектов базы данных и их отношений, как в графическом виде, так и в виде готовых стандартных печатных отчетов, что существенно облегчает

ведение проекта. На данный момент не существует единственной общепринятой системы обозначений для ER-модели и различные CASE-системы используют различные графические нотации, знание одной из них, например, нотации Чена, позволяет легко понять и другие нотации.

ER-модель данных

ER-модель данных служит основой многих методик системного анализа и проектирования, CASE-средств и репозиториев программного обеспечения. ER-модель является основой для таких программных пакетов, как Repository Manager MVS компании IBM и CDD Plus компании Digital Equipment Corporation. ER-модель данных была принята ANSI в качестве метамодели для Каталога системных информационных ресурсов (IRDS), ER-подход занимает первое место среди методологий проектирования баз данных и, согласно опросам компаний из списка Fortune 500, является одной из лучших методологий в развитии информационных систем.

Прежде, чем приступать к созданию системы автоматизированной обработки информации, разработчик должен сформировать понятия о предметах, фактах и событиях, которыми будет оперировать данная система. Для того, чтобы привести эти понятия к той или иной модели данных, необходимо заменить их информационными представлениями. Модель «сущность-связь» основывается на некой важной семантической информации о реальном мире и предназначена для логического представления данных. Она определяет значения данных в контексте их взаимосвязи с другими данными. Важным является тот факт, что из модели «сущность- связь» могут быть порождены все существующие модели данных (иерархическая, сетевая, реляционная, объектная), поэтому она является наиболее общей.

CASE - средства

ER-модель повлияла на разработку большинства основных CASE-инструментов, включая ERWIN, Designer, PowerDesigner и даже Microsoft Visio, а также стандарт IDEF1X. В конце 1980-х и начале 1990-х на основе ER-модели данных были разработаны такие программные продукты, как репозиторий DB2 и AD Cycle компании IBM. На ER-модели данных основаны и системы репозиториев других вендоров, таких как CDD.

Концепция гипертекста, которая делает World Wide Web чрезвычайно популярным, очень похожа по своей сути на ER-модель. ER-модель также лежит в основе некоторых работ по объектно-ориентированному анализу, методологий проектирования и семантического Web. Язык моделирования UML также имеет корни в ER-модели.

Цель инфологического проектирования предметной области

Целью инфологического моделирования является обеспечение разработчика базы данных концептуальной схемой базы данных на уровне представлений о предметной области.

Как инструмент инфологическое моделирование использует различные варианты ER-диаграмм. С их помощью определяются важные для предметной области объекты (*сущности*), их свойства (*атрибуты*) и отношения между ними

(связи). Инфологическое моделирование отображает тот факт, что сущности имеют связи между собой, а атрибуты принадлежат сущностям.

Инфологическая модель данных отображает данные с точки зрения их смысла и в контексте пользователя. Модель скрывает технические детали и подчеркивает наиболее важную суть, с точки зрения предметной области и пользователя. Чаще всего на практике инфологическое моделирование используется на стадии инфологического проектирования базы данных. При этом в терминах модели проводится разработка концептуальной схемы базы данных, которая затем вручную приводится к реляционной схеме. Этот процесс выполняется с использованием методик, в которых достаточно четко обусловлены все этапы такого превращения.

Сущности и атрибуты

Модель «сущность-связь» имеет несколько базовых понятий, которые образуют начальные «кирпичики», из которых строятся уже более сложные объекты по предварительно определенным правилам. Основными конструктивными элементами модели «сущность-связь» являются: ***сущность, экземпляр сущности, атрибуты, связи и ассоциативные сущности***.

В модели «сущность-связь» предметная область представляется сущностями.

Сущности - это объекты, которые существуют физически или концептуально. Их характеризуют атрибуты (**attributes**) и связи (**relationships**) с другими сущностями.

В примере «Государственный парк» сущностями являются: **ЛЕС**, **ОЗЕРО**, **УЧАСТОК** (лесонасаждение), **ДОРОГА**, **ДИРЕКТОР**, **СОТРУДНИК**, **ПОСТРОЙКА**.

Сущности описываются атрибутами

Атрибут выражает определенное свойство объекта. С точки зрения БД (физическая модель) сущности соответствует таблица, экземпляру сущности - строка в таблице, а атрибуту - колонка таблицы.

Атрибут сущности - это некоторая характеристика сущности, которая описывает одно из ее свойств. Атрибут имеет имя и принимает значение из некоторого множества значений (например, у сущности **КНИГА** могут быть атрибуты: **КОД КНИГИ**, **РАЗДЕЛ ЛИТЕРАТУРЫ**, **НАЗВАНИЕ**, **АВТОРЫ**, **ЦЕНА**).

ПРИМЕР. **НАЗВАНИЕ** - это атрибут сущности **ЛЕС**. Атрибут (или набор атрибутов), который однозначно идентифицирует экземпляр сущности, называется ***ключом***.

ПРИМЕР. Атрибут **КД (код дороги)** название сущности **ДОРОГА** является ключом. Все экземпляры сущности **ДОРОГА** в этой базе данных имеют уникальные коды.

Атрибуты могут быть однозначными и многозначными.

ПРИМЕР. **ПОРОДА** (преобладающая порода деревьев на участке) - однозначный атрибут сущности **УЧАСТОК**.

Понятие ***многозначных атрибутов*** можно пояснить на следующем

примере. Сущность **ПОСТРОЙКА** имеет атрибут **НОМЕР ТОЧКИ**. Он служит уникальным идентификатором пространственного положения экземпляра сущности.

На карте отдельные экземпляры сущности **ПОСТРОЙКА** отображаются точками. Постройка может занимать два и более различных геометрических мест. Тогда атрибут **НОМЕР ТОЧКИ** является многозначным. Это справедливо и для других сущностей.

ПРИМЕР. Необходимо сохранить информацию о высоте деревьев **ЛЕСА**. Поскольку она может изменять свое значение в пределах сущности **ЛЕС**, необходимо моделировать высоту как многозначный атрибут.

Связи

Третья конструкция в модели «сущность-связь» - это связь. Посредством связей сущности взаимодействуют или соединяются друг с другом. В одной связи может быть задействовано большое количество сущностей. Будем рассматривать только бинарные связи. Такие связи соединяют две сущности. Существует три вида связей: «один к одному», «многие к одному», «многие ко многим».

Один к одному (1:1). Каждый экземпляр одной сущности может соотноситься лишь с одним экземпляром другой сущности. Например, связь **ОБСЛУЖИВАЕТ** между сущностями **СОТРУДНИК** и **ОЗЕРО** является связью вида «один к одному»; **ОЗЕРО** может обслуживаться только одним **СОТРУДНИКОМ**, а **СОТРУДНИК** может обслуживать только одно **ОЗЕРО**.

Многие к одному (M:1). Связь «многие к одному» потенциально может соединять множество экземпляров одной сущности с одним экземпляром другой сущности, участвующей в связи. **ПРИНАДЛЕЖИТ** - это связь вида «многие к одному» между сущностями **ПОСТРОЙКА** и **ЛЕС** при условии, что каждая постройка может находиться на территории только одного леса, но таких, построек может быть много.

Многие ко многим (M:M). Иногда ряд экземпляров одной сущности должен быть связан с рядом экземпляров другой сущности.

Построение ER-диаграмы

На этапе инфологического проектирования базы данных необходимо определить инфологическую модель предметной области. Задача создания модели отделить понятийный аппарат приложения от деталей реализации.

В примере «Государственный парк» атрибутами являются: КОД СОТРУДНИКА, ФАМИЛИЯ, НОМЕР ТЕЛЕФОНА, КОД ОЗЕРА, ОСНОВНОЙ ВИД ВЫЛАВЛИВАЕМОЙ В ОЗЕРЕ РЫБЫ и ПЛОЩАДЬ ОЗЕРА.

Решение задачи

Предположения при создании ER-диаграммы: все сотрудники имеют работу, некоторые озера не обслуживаются. Следовательно, класс принадлежности сущности **СОТРУДНИК** обязательный, а сущности **ОЗЕРО** необязательный.

Диаграмма «сущность-связь» примера «Государственный парк» приведена на рисунке 3.1. На ней показаны семь сущностей: **УЧАСТОК (ЛЕСОНАСАЖДЕНИЕ), ОЗЕРО, ДОРОГА, ПОСТРОЙКА, ЛЕС,**

ДИРЕКТОР и **СОТРУДНИКИ.**

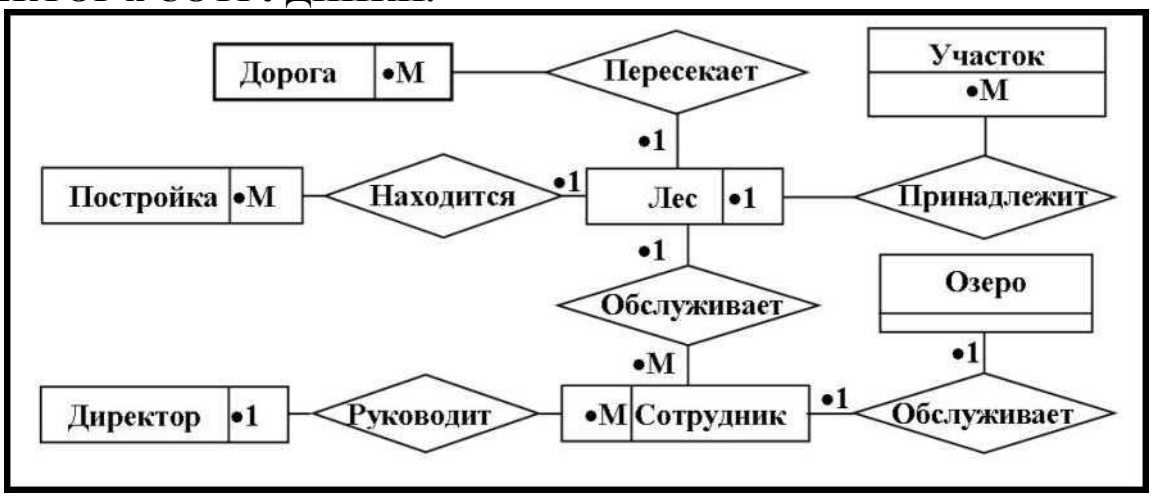


Рисунок 3.1 - Диаграмма «сущность-связь» для примера «Государственный парк»

На диаграмме «сущность-связь» сущности представлены как прямоугольники, связи представлены как ромбы. Иногда указываются и атрибуты. Атрибуты обозначаются овалами, которые соединены с прямоугольниками при помощи прямых линий.

Кардинальность связи (1:1, M:l или M:M) указывается около ромба или внутри прямоугольника. Ключевые атрибуты подчеркиваются, многозначные представляются двойными овалами.

Теперь необходимо пояснить принятые в диаграмме обозначения. Класс принадлежности сущности к связи может быть обязательным и необязательным.

При ***обязательном классе*** принадлежности каждый экземпляр сущности обязательно должен быть связан с другой сущностью.

При ***необязательном классе*** принадлежности не требуется, чтобы каждый экземпляр сущности был связан с каким-либо экземпляром другой сущности. В БГД хранится информация обо всех экземплярах сущностей, но только некоторые из них связаны с экземплярами другой сущности.

Для отображения характеристик связи на диаграмме может использоваться, например, символ «точка». Если точка внутри прямоугольника, то соответствующая прямоугольнику сущность имеет обязательный класс принадлежности. Если вне прямоугольника, то необязательный класс принадлежности. Цифры или буквы рядом с точками указывают на степень связи.

Сочетание трех типов связей с двумя классами принадлежности дают возможность описания множества различных вариантов связей в предметной области. На диаграмме (рис 1.31) представлены шесть связей. Сущность **ЛЕС** участвует в четырех из них. Сущность **ДИРЕКТОР** участвует только в одной связи - руководит. Ограничения кардинальности показывают, что каждый **СОТРУДНИК** обслуживает только один **ЛЕС**, но один лес может обслуживаться несколькими сотрудниками.

Отдельные связи являются по своей природе пространственными. К ним относятся следующие связи: **ПЕРЕСЕКАЕТ**, **НАХОДИТСЯ**, **ПРИНАДЛЕЖИТ**, **РУКОВОДИТ**, **ОБСЛУЖИВАЕТ**.

Отображение ER-диаграмм на реляционной модели

Общий подход к построению БД с использованием ER-метода состоит в выполнении следующих шагов:

1. Построение диаграммы ER-типа. В диаграмму должны быть включены все сущности и связи, важные с точки зрения интересов организации.
2. Анализ связей и определение их характеристик: степень связи и класс принадлежности.
3. Построение набора предварительных отношений с указанием предполагаемого первичного ключа для каждого отношения. Имя отношения - существительное в единственном числе.
4. Подготовка списка всех атрибутов (тех, которые не были перечислены в диаграмме ER-типа в качестве ключей сущности) и назначение каждого из этих атрибутов одному из предварительных отношений. Эти отношения должны находиться в нормальной форме Бойса Кодда (НФБК).
5. Проверка, все ли полученные отношения находятся в НФБК.
6. Построение схемы данных.
7. Уточнение ER-диаграмм, если полученные отношения не находятся в НФБК или некоторые атрибуты отсутствуют в предварительных отношениях.

Ранее были выполнены первые два шага процесса проектирования, третий шаг - построение предварительных отношений. Построение отношений выполняется по определенным правилам, которые будут рассмотрены далее.

Предварительные отношения для бинарных связей 1:1

Перечень общих правил генерации отношений из диаграмм ER-типа можно получить, опираясь на класс принадлежности и степень связи.

Правило 1. Если степень бинарной связи 1:1 и класс принадлежности обеих сущностей является обязательным, то требуется только одно отношение. Первичным ключом этого отношения может быть ключ любой из двух сущностей.

ПРИМЕР 1. Сотрудник обслуживает озеро (см. рис. 3.1)

Получаем следующее отношение (табл. 1.1).

СОТРУДНИК (**КС,** Фамилия, Телефон, КО, Рыба, Площадь).

Таблица 3.1 - Первоначальный вид таблицы «Сотрудники»

КС	**Фамилия**	**Телефон**	**КО**	**Рыба**	**Площадь**
С1	Иванов	234566	О1	Карась	8
С2	Андреев	233367	О2	Карп	6
С3	Суслов	226785	О3	Щука	3
С4	Репин	274564	О4	Лещ	9

Гарантируется однократное появление каждого значения **КС** (код сотрудника) и **КО** (код озера). Отношение никогда не будет содержать ни пустых данных, ни повторяющихся групп избыточных данных.

Правило 2. Если степень бинарной связи 1:1 и класс принадлежности

одной сущности является обязательным, а другой - необязательным, то необходимо построение двух отношений. Под каждую сущность выделяется одно отношение, при этом ключ сущности должен служить первичным ключом для соответствующего отношения. Кроме того, ключ сущности, для которого класс принадлежности является необязательным, добавляется в качестве атрибута в отношение, выделенное для сущности с обязательным классом принадлежности.

ПРИМЕР 2. Класс принадлежности сущности **СОТРУДНИК** - обязательный, а сущности **ОЗЕРО** - необязательный (рис. 3.2).

Рисунок 3.2 - ER-диаграмма для примера 2

Получаем отношения: **СОТРУДНИК** (**КС,** Фамилия, Телефон, КО) (табл. 3.2), **ОЗЕРО** (**КО,** Рыба, Площадь) (табл. 1.3).

Таблица 3.2 - Измененная таблица «Сотрудники»

КС	**Фамилия**	**Телефон**	**КО**
С1	Иванов	234566	О1
С2	Андреев	233367	О2
С3	Суслов	226785	О3
С4	Репин	274564	О4

Таблица 3.3 - Первоначальный вид таблицы «Озера»

КО	**Рыба**	**Площадь**
О1	Карась	8
О2	Карп	6
О3	Щука	3
О4	Лещ	9

Правило 3. Если степень бинарной связи равна 1:1 и класс принадлежности ни одной из сущностей не является обязательным, то необходимо использовать три отношения: по одному для каждой сущности и одно отношение для связи. Ключ каждой сущности используется в качестве первичного ключа соответствующего отношения. Отношение связи должно иметь в числе своих атрибутов ключи каждой сущности.

ПРИМЕР 3. Класс принадлежности сущностей **СОТРУДНИК** и **ОЗЕРО** - необязательный (рис. 3.3).

Получаем отношения: **СОТРУДНИКИ** (**КС**, Фамилия, Телефон) (табл. 3.4), **ОЗЕРО** (**КО**, Рыба, Площадь) (табл. 3.5), **ОБСЛУЖИВАЕТ** (**КС, КО**) (табл. 3.6).

Рисунок 3.3 - ER-диаграмма для примера 3

Таблица 3.4 - Окончательный вид таблицы «Сотрудники»

КС	Фамилия	Телефон
С1	Иванов	234566
С2	Андреев	233367
С3	Суслов	226785
С4	Репин	274564

Таблица 3.5 - Измененная таблица «Озера»

КО	Рыба	Площадь
О1	Карась	8
О2	Карп	6
О3	Щука	3
О4	Лещ	9

Таблица 3.6 - Озера, обслуживаемые сотрудниками

КС	КО
С1	О1
С3	О3
С4	О4

Проведя анализ диаграммы можно заключить, что оба отношения находятся в НФБК. Детерминанты КС и КО являются ключевыми атрибутами.

Предварительные отношения для бинарных связей 1:N

В такой ситуации используются два правила. Каждое из них определяется классом принадлежности М-связной сущности. Класс принадлежности односвязной сущности на результат не влияет. Рассмотрим правила построения отношений.

Правило 4. Если степень бинарной связи равна 1:М и класс принадлежности М-связной сущности является обязательным, то это значит, что достаточным является использование двух отношений, по одному на каждую сущность. Ключ каждой сущности служит первичным ключом для соответствующего отношения. Дополнительно ключ односвязной сущности должен быть добавлен как атрибут в отношение для М-связной сущности.

ПРИМЕР 4. Класс принадлежности сущностей **СОТРУДНИК** (**КС**, Фамилия, Телефон) - необязательный, **ОЗЕРО** (**КО**, Рыба, Площадь, КС) - обязательный (рис. 3.4).

Получаем отношения: **СОТРУДНИКИ** (**КС**, Фамилия, Телефон) (табл. 3.3), **ОЗЕРО** (**КО**, Рыба, Площадь, КС) (табл. 3.7).

Рисунок 3.4 - ER-диаграмма для примера 4

Таблица 3.7 - Окончательный вид таблицы «Озера»

КО	**Рыба**	**Площадь**	**КС**
O1	Карась	48	C1
O2	Карп	56	C2
O3	Щука	34	C1
O4	Лещь	68	C4

Правило 5. Если степень бинарной связи равна 1:N и класс принадлежности N-связной сущности является необязательным, то это значит, что необходимо формирование трех отношений: по одному для каждой сущности и одного отношения для связи. Причем ключ каждой сущности будет использован в качестве первичного ключа соответствующего отношения. Отношение связи должно иметь в числе своих атрибутов ключи каждой сущности.

ПРИМЕР 5. Класс принадлежности сущностей **СОТРУДНИК** (**КС**, Фамилия, Телефон) - необязательный, **ОЗЕРО** (**КО**, Рыба, Площадь, КС) - необязательный (рис. 3.5).

Получаем отношения: **СОТРУДНИКИ** (**КС**, Фамилия, Телефон) (табл. 3.3), **ОЗЕРО** (**КО**, Рыба, Площадь, КС) (табл. 3.6).

Рисунок 3.5 - ER-диаграмма для примера 5

Автоматизация отображения модели «сущность-связь» на реляционную модель

Выполнять преобразование модели «сущность-связь» в реляционную схему могут многие программные пакеты, называемые также CASE-средствами. Среди них - ERwin, Oracle Designer 2000, Rational Rose и другие. Такая возможность преобразования позволяет проектировщикам баз данных работать с инфологической моделью данных, концентрируясь на нуждах предметной области приложения. Если бы не присутствие пространственных атрибутов, можно было бы произвести бесшовное и интуитивное отображение модели «сущность-связь» на реляционную модель данных. Перечислим пять основных этапов отображения модели «сущность-связь» на реляционную модель данных:

1. Отобразить каждую сущность в отдельное отношение. Атрибуты

сущности отображаются в атрибуты отношения. Аналогично ключ сущности становится первичным ключом отношения.

2. Для связей с кардинальностью 1:1 сделать ключевой атрибут любой одной сущности внешним ключом другой сущности.

3. Если связь имеет кардинальность M:l, сделать первичный ключ отношения на стороне 1 внешним ключом отношения на стороне M.

4. Связи с кардинальностью M:M следует обрабатывать иначе. Каждую связь M:M отобразить на новое отношение. Имя этого отношения должно совпадать с названием связи, а первичный ключ отношения - состоять из пары первичных ключей задействованных сущностей. Если связь имеет какие-либо атрибуты, то они становятся атрибутами нового отношения.

5. Для многозначных атрибутов создать новое отношение, в состав которого должны входить два столбца: столбец, соответствующий многозначному атрибуту, и столбец, соответствующий ключу сущности, которая владеет этим многозначным атрибутом. Вместе многозначный атрибут и ключ сущности должны образовать первичный ключ нового отношения.

ПРИМЕР. Сущность **УЧАСТОК** (Лесонасаждение) имеет многозначный атрибут **НОМЕР_УЧАСТКА**, который является целочисленным идентификатором геометрического местоположения участка. **НОМЕР_ТОЧКИ** является многозначным атрибутом, поскольку лесонасаждение может занимать два непересекающихся геометрических многоугольника (например, лесонасаждение может разделять дорога). Поэтому необходимо использовать отношение **Геом_Насаждение**. Аналогично следует вводить отношения **Геом_Лес, Геом_Озеро, Геом_Постройка**.

Атрибут **ВЫСОТА** требует иного подхода. Прежде всего, необходимо отметить что высота - это многозначный атрибут. Поэтому, требуется новое отношение **Высота**. На рисунке 1.36 показано, что атрибутами этого нового отношения являются **Название_леса, Номер_точки** и **Высота**. Атрибут **Высота** содержит высоту лесонасаждения в точке **Номер_точки**. В этой таблице все три атрибута образуют первичный ключ.

Рисунок 3.6 - Схема представления участков лесонасаждений, точек и высот

Пространственные таблицы. Топология данных

В реляционной модели пространственные и пространственнозависимые атрибуты, присутствующие на диаграмме «сущность-связь», следует обрабатывать особым образом. Новые домены, например пространственные объекты, представляют как новые отношения. Первичные ключи этих отношений

используют как внешние ключи в тех отношениях, которые представляют сущности, содержащие атрибуты этих типов (доменов).

Как было описано ранее, **Номер_Точки,** и **Номер_Участка** - это новые домены, и их можно смоделировать с использованием отдельных отношений. Каждому из этих атрибутов соответствует одно отношение: **Точка** (Point) и **Многоугольник** (Polygon).

Таблица **Точка** имеет три атрибута: **Номер_Точки, Долгота** и **Широта**. Несмотря на то, что существует большое количество других систем координат, более распространенная географическая система (широта, долгота), и все прочие системыможно получить на ее основе.

Прямая линия конечного размера может быть описана двумя точками. Следовательно, атрибут **номер_точки** в **таблице Линия** является внешним ключом таблицы точек. Атрибут **порядковый_номер** указывает порядковый номер точек, образующих линию с атрибутом **номер_линии**.

В таблице **Участок** первый и последний порядковые номера должны указывать на один и тот же **номер_точки**.

В итоге можно сделать вывод о том, что модель «сущность-связь» находит широкое применение в моделировании данных на концептуальном уровне, хотя ее и не разрабатывали как особую форму представления данных: в основном данную модель используют для работы с простыми типами данных и конструкциями. Эту модель можно отобразить на модель реляционной базы данных, последовательно выполняя ряд подробно описанных действий.

Модель «сущность-связь» и реляционную модель необходимо обогащать с целью поддержки особых характеристик пространственных данных. Модель «сущность-связь» можно расширить путем введения пиктограмм, которые символически изображают различные типы пространственных данных и связей. Аналогично реляционную модель можно расширять посредством встраивания новых типов данных и связанных с ними операций.

Задание Постройте диаграмму сущность-связь, предлагаемой предметной области, используя нотацию Чена и нотацию Мартина. В отчёте необходимо описать предлагаемые сущности предметной области и их атрибуты с указанием ключевых атрибутов. Приведенное в задание описание предметной области является приблизительным. Допускается уточнять список предлагаемых сущностей и атрибутов.

Информационная система “Парки города” хранит информацию о парках. Каждый парк имеет собственное имя. В парке высажены определённые насаждения. База данных должна хранить информацию о количестве насаждений каждого типа. В парке могут находиться фонтаны и павильоны.

Атрибуты парка:

- наименование,
- площадь,
- место нахождения (адрес).

Насаждения парка

- тип культуры,

- наименование,
- средняя продолжительность жизни.

Фонтан

- шифр,
- дата постройки,
- расход воды (максимальный и нормальный),
- площадь.

Павильон

- наименование,
- тип (кафе, продуктовый, развлекательный, прокат вещей),
- занимаемая площадь.

Практическая работа 4. Работа с пространственными запросами в SQL

Цель работы: усвоить основные принципы создания запросов вSQL для работы с данными.

Задачи работы

1. Изучить:

- запросы, предназначенные для работы со структурой данных;
- запросы, используемые непосредственно в работе с данными;
- запросы, применяемые для предоставления или отмены прав доступа к БД.

2. Научиться:

- формировать SQL-команды для управления структурой данных
- формировать SQL-команды для работы с данными
- формировать SQL-команды для управления правами доступа
- формировать SQL-команды для операций с пространственными объектами

SQL-запросы чаще всего делятся на следующие три категории:

- запросы, предназначенные для работы со структурой данных (Data Definition Language - DDL), то есть для создания, описания и модификации структуры базы данных и ее элементов;

- запросы, используемые непосредственно в работе с данными (Data Manipulation Language - DML), с помощью которых можно добавлять, обновлять и удалять данные;

- запросы, применяемые для предоставления или отмены прав доступа к БД (Data Control Language - DCL).

SQL-команды для управления структурой данных

К этой категории относятся команды, работающие со структурой БД. К ним относятся CREATE - «создать» (например, CREATE TABLE (создать таблицу), CREATE USER (создать пользователя), ALTER - «модифицировать» (этот запрос используется при внесении изменений в саму БД или в ее часть), DROP - «удалить» (также относится к БД и ее частям).

Например, синтаксис SQL-команды для создания новой таблицы (CREATE TABLE):

CREATE TABLE <table_name1> (
<col_name1><col_type1>,
<col_name2><col_type2>,
<col_name3><col_type3>
PRIMARY KEY(<col_name1>),
FOREIGN KEY(<col_name2>)
REFERENCES <table_name2> (<col_name2>));

При создании таблиц обычно сразу задаются ограничения целостности данных для определенных столбцов в таблице. Чаще всего применяются следующие ограничения:

- NOT NULL - ячейка таблицы не может иметь значение NULL;
- PRIMARY KEY(col_name1, col_name2, ...) - первичный ключ;
- FOREIGN KEY (col_name_1, ..., col_name_n) REFERENCES table_name(col_name_1, ..., col_name_n) - внешний ключ. SQL-синтаксис позволяет задавать более одного первичного ключа. В таком случае формируется составной первичный ключ.

SQL-команды для работы с данными

К наиболее востребованным командам работы с данными относятся: SELECT (выборка данных), INSERT (вставка новых данных), UPDATE (обновление данных), DELETE (удаление данных), MERGE (слияние данных).

Общая структура запроса на выборку (SELECT) данных выглядит следующим образом (табл. 4.1):

SELECT < col_name1, col_name2, .>
FROM<table_name1, table_name2, ...>
WHERE ■ condinonl AND/OR/NOT -condition? AND/OR/NOT ...
GROUP BY < col_name1, col_name2, .>
HAVING <gropu_conditionl> AND/OR/NOT .
ORDER BY< col_namel, col_name2, .>
LIMIT <int_digit>;

Таблица 4.1 - Элементы SQL-запроса

Ключевое слово	Параметры	Обязательность
SELECT	Список столбцов или * для выбора всех столбцов	Да
FROM	Список таблиц	Да
WHERE	Набор условий, объединенных логическими операторами	Нет
GROUP BY	Список столбцов для группировки результатов выборки с помощью агрегирующих функций	Нет
HAVING	Набор условий, объединенных логическими операторами для сгруппированных данных	Нет (используется только совместно с GROUP BY)

ORDER BY	Список столбцов, по которым сортируются результаты запроса	Нет
LIMIT	Ограничение количества записей, выводимых в результате запроса	Нет

Команда INSERT INTO <table_name> в SQL отвечает за добавление данных в таблицу:

INSERT INTO <table_name> (<col_namel>, <col_name2>, <col_name3>, .) VALUES (<valuel>, <value2>, <value3>, .);

При добавлении данных в каждый столбец таблицы можно не указывать названия столбцов.

INSERT INTO <table_name> VALUES (<value1>, <value2>, <value3>, ...);

UPDATE - это SQL-команда для обновления данных в таблице:

UPDATE <table_name> SET <col_name1> = <value1>, <col_name2> = <value2>, ... WHERE <condition>;

SQL-команда DELETE FROM используется для удаления данных из таблицы:

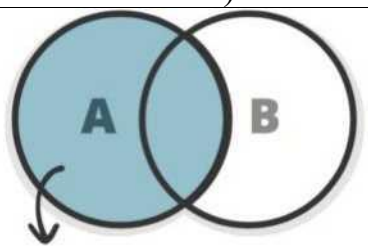
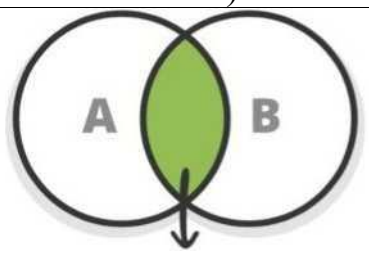
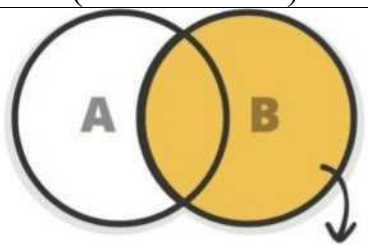
DELETE FROM <table_name> WHERE <condition>;

Также для связанных запросов (с помощью общих атрибутов) к нескольким таблицам используется оператор JOIN. Общий синтаксис запроса приведен далее:

SELECT<col_name1>, <col_name2>, ...
FROM <table_name1>
JOIN <table_name2>
ON <table_name1.col_name> = <table2.col_name>;

В табл. 3.2 ниже показаны различные способы объединения в SQL.

Таблица 3.2 Схемы запросов JOIN

Левостороннее объединение (LEFT JOIN).	Внутреннее объединение (INNER JOIN).	Правостороннее объединение (RIGHT JOIN).
		
Результатом будут все строки из таблицы А и те строки из таблицы В, которые имеют те же значения, что и в таблице А для выбранных столбцов	Результатом будут только те строки из таблицы *A* и *B*, которые имеют одинаковые значения для выбранных столбцов	Результатом будут все строки из таблицы *B* и те строки из таблицы *A*, которые имеют те же значения, что и в таблице *B* для выбранных столбцов

SQL-команды для управления правами доступа

К командам, работающим с правами доступа, относятся GRANT - разрешение пользователю на проведение определенных операций с БД или данными, REVOKE - отзыв выданного разрешения, DENY - установка запрета, имеющего приоритет над разрешением.

Общий синтаксис команды GRANT:

- privilege_name - это список команд, разрешение на выполнение которых выдается пользователю (например, SELECT, INSERT);
- object_name - это имя объекта базы данных, на который выдаются разрешения (например, таблица или хранимая процедура);
- user_name - это имя пользователя, которому выдаются разрешения (может быть указано несколько пользователей одновременно, через запятую);
- role_name - указывается вместо конкретных пользователей и означает группу пользователей объединенных по какому-то критерию.

GRANT privilege_name
ON object_name
TO {user_name |PUBLIC |role_name}
[WITH GRANT OPTION];

Общий синтаксис команды для удаления разрешений:

REVOKE privilege_name
ON object_name
FROM {user_name |PUBLIC |role_name}

SQL-команды для операций с пространственными объектами

Для получения характеристик (например, количество узлов, площадь, периметр и т. д.) и выполнения операций с пространственными объектами СУБД предоставляет дополнительные команды, которые могут быть использованы в стандартных SQL-запросах.

Расширения СУБД для работы с пространственными данными добавляют дополнительные типы данных: точки, линии, полигоны, мультиобъекты, поверхности, геопривязанные растры.

Рассмотрим синтаксис наиболее популярных команд, используемых в SQL-запросах, при работе с пространственными данными.

Проверка пересечения (ST_Intersects), содержания (ST_Contains) или входимости (ST_Within) двух наборов пространственных объектов:

ST_Intersection (или ST_Contains, или ST_Within) (поле геометрии набора объектов №1, поле геометрии набора объектов №2);

Вычисление длины линейных объектов:

ST_Length(поле геометрии);

Вычисление периметра полигональных объектов:

ST_Perimeter(поле геометрии);

Вычисление площади объекта:

ST_Area(поле геометрии);

Вычисление центроида объекта:

ST_Centroid(поле геометрии);

Вычисление расстояния между ближайшими точками объектов:

ST_Distance(поле геометрии набора объектов №1, поле геометрии набора объектов №2);

Сшивка пространственных объектов:

ST_Union(поле геометрии набора объектов №1, поле геометрии набора объектов №2);

Построение буферной зоны для одного или более пространственных объектов:

ST_Buffer(поле геометрии, радиус буферной зоны в единицах измерения системы координат таблицы данных, количество сегментов буферной зоны);

При использовании эти операторы задействуются непосредственно в тексте запроса. Например, далее представлен текст запроса для построения минимально-описывающих полигонов вокруг каждого из наборов объектов таблицы:

SELECT line, ST_ConvexHull(ST_Union(geom)) AS hull
FROM s
GROUP BY line;

Индивидуальные задания

Практическаяс сайта qgistutorials.com - Анализ количества точек в полигонах

Задача: Посчитать число землетрясений в России (NAME='Russia')

Пусть слои называются **countries, signif**.

Добавить виртуальный слой (Virtual Layer).

В поле Запрос написать:

select countries.*, ***count***(signif.geometry) **as** cnt
from countries **join** signif **on** ***st_intersects***(countries.geometry, signif.geometry)
where countries.NAME='Russia'
group by countries.geometry;

Примечание. Запрос может выполняться очень долго!

Практическая работа 5. Интеграция данных различных типов в базу пространственных данных геоинформационных систем

Цель работы: изучить основные этапы импорта табличных и пространственных данных в реляционную СУБД, усвоить понятие агрегирующих функций языка SQL, принципы составления, выполнения и получения данных для оптимизации SQL-запросов, в том числе пространственных, к нескольким таблицам, создания представлений и их использования в геоинформационных системах.

Задачи работы

1.Изучить:

- способы импорта данных в реляционные СУБД;
- принципы составления и оценки SQL-запросов;
- принципы создания и использования представлений в реляционных СУБД.

2. Научиться:

- выполнять агрегирующие операции к данным в реляционных СУБД с помощью SQL-запросов;
- оценивать производительность запросов и ускорять их с помощью индексов;
- выполнять пространственные запросы к нескольким таблицам.

Перечень обеспечивающих средств

Для выполнения работы необходимо иметь методические указания по выполнению работы, а также программное обеспечение PostgreSQL (версия 13 и выше), PostGIS (версия 3 или выше), pgAdmin (версия 4 или выше).

Создание базы данных

Запустить интерфейс к СУБД pgAdmin 4. Развернуть список серверов (пункт *Servers*). Найти сервер с именем *PostgreSQL 13* и подключиться к нему, щелкнув два раза кнопкой мыши или выбрав из контекстного меню пункт *Connect Server* (рис. 5.1). Пароль пользователя - тот, который был задан при установке (по умолчанию *postgres*).

Развернуть список *Databases* и открыть у него контекстное меню для создания базы данных (рис. 5.2).

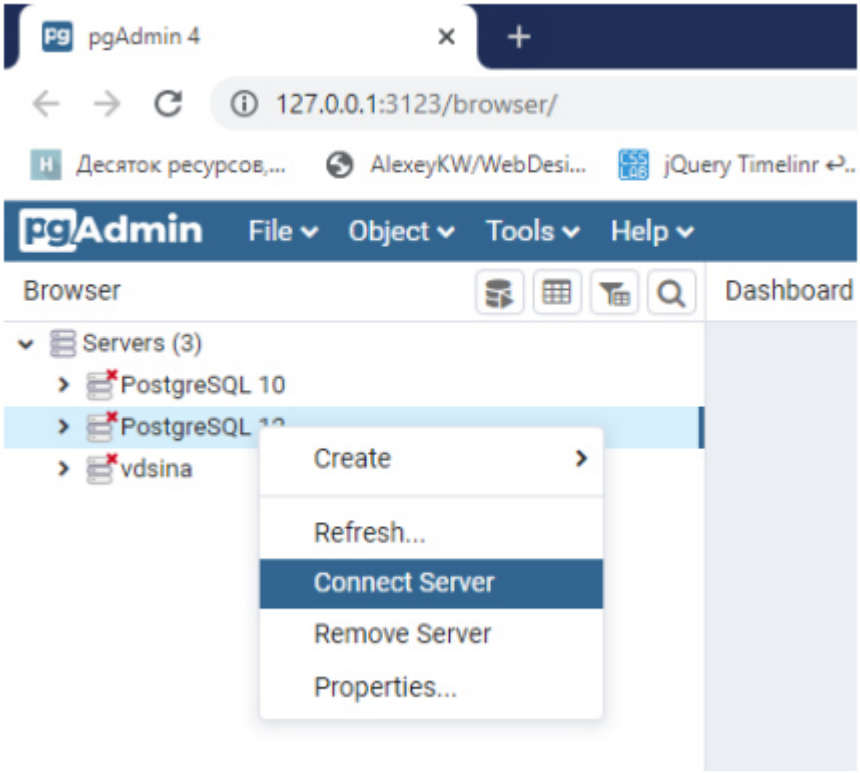


Рис. 5.1. Контекстное меню сервера баз данных

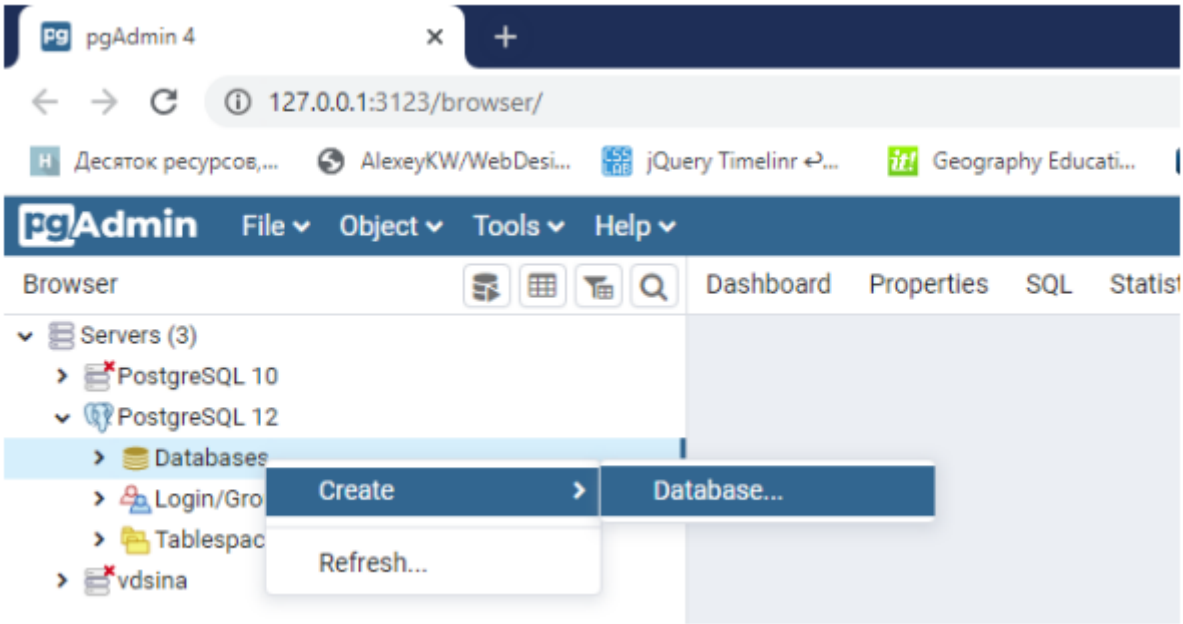


Рис. 5.2. Контекстное меню создания базы данных

Выбрать *Create* и далее *Database*. В открывшемся диалоге ввести в качестве названия базы данных свою фамилию транслитом и нажать на кнопку *Save* (рис. 5.3). В результате должна создаться новая база данных.

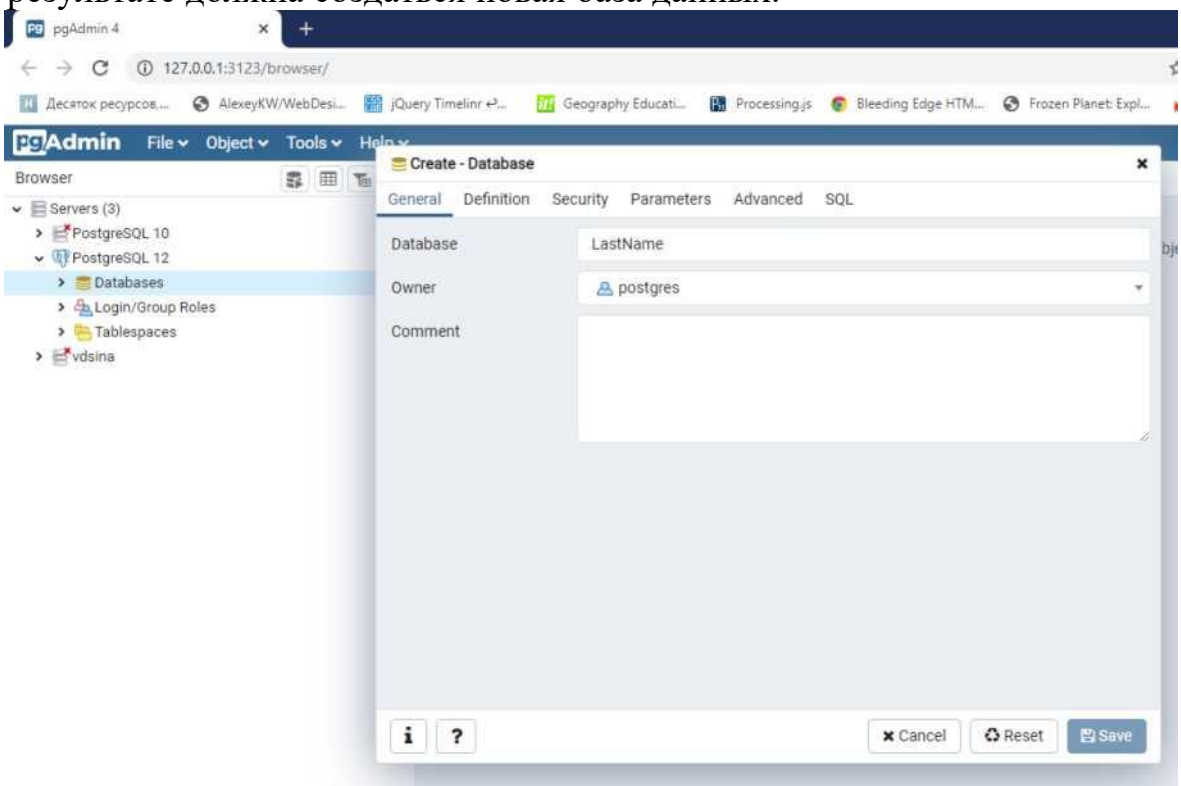


Рис. 5.3. Параметры создаваемой базы данных

Загрузка векторных данных в PostgreSQL

Для активации возможности работы с геоданными в созданной базе данных требуется предварительно активировать расширение *PostGIS*. Для этого нужно открыть вкладку для написания запросов *Query Tool* в меню *Tools* (рис. 5.4).

Затем вписать команды:

CREATE EXTENSION postgis;

CREATE EXTENSION postgis_topology; и нажать на кнопку *Execute/Re- fresh*

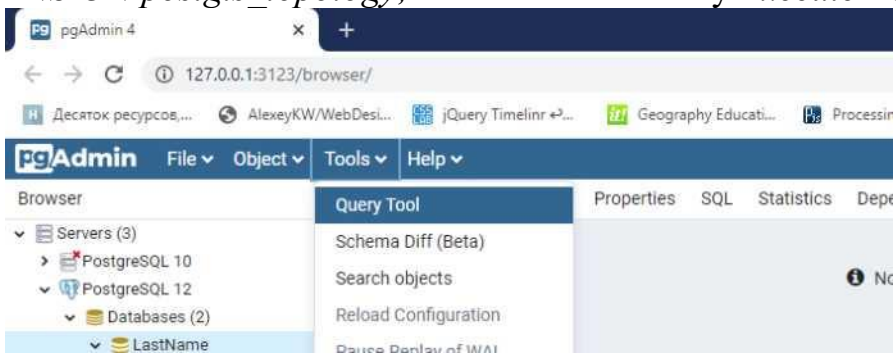


Рис. 5.4. Меню инструментов pgAdmin

В случае успешного выполнения будет выдано сообщение *Query returned successfully in...* В этом случае можно выполнить загрузку векторных данных по странам мира (*Countries.TAB*) из предыдущей работы в базу данных, используя утилиту *PostGIS Shapefile and DBF Loader*. Для получения SHP можно воспользоваться QGIS. Необходимо вызвать команду главного меню QGIS *Слой - Добавить слой - Добавить векторный слой.* Затем открыть контекстное меню слоя и выбрать пункт *Экспорт - Сохранить объекты как. (*рис. *5.5).*

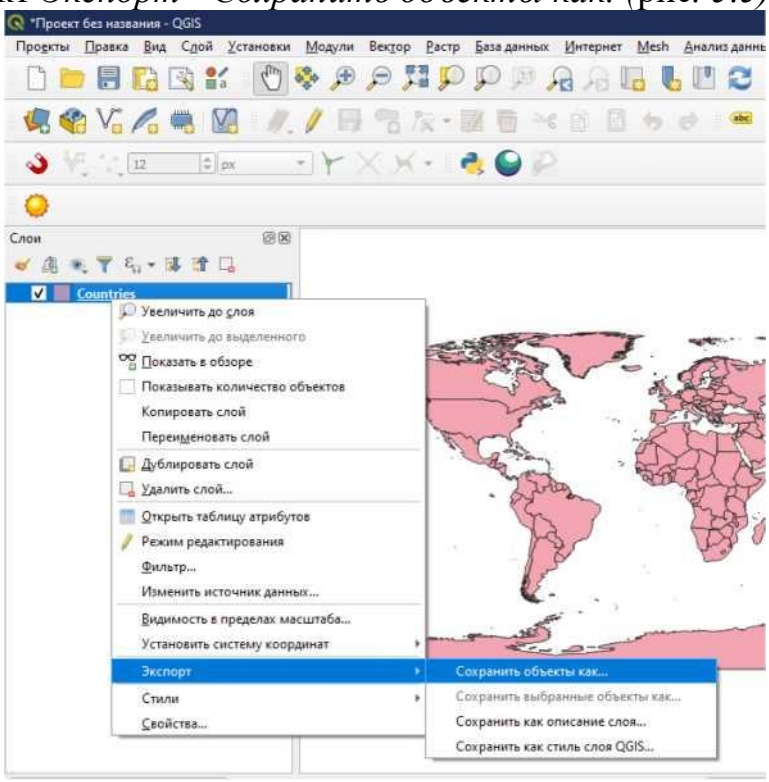


Рис. *5.5.* Меню экспорта слоев

Указать формат сохранения *Shape-файл,* расположение файла (для последующего импорта в PostgreSQL в пути не должно быть кириллических символов) и кодировку *UTF-8* (рис. 5.6).

Далее нужно запустить программу *PostGIS Shapefile and DBF Loader*,

которая устанавливается вместе с PostGIS. В открывшемся диалоговом окне нужно прописать параметры соединения в диалоге по кнопке *View connection detail*. Имя базы данных указывается то же самое, что было выбрано в начале работы (рис. 5.7).

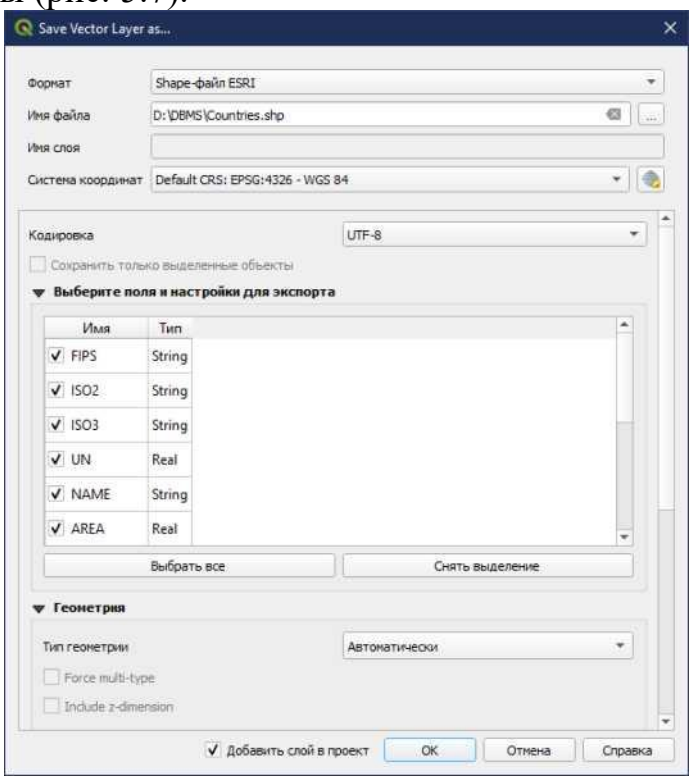


Рис. 5.6. Параметры экспорта векторного слоя

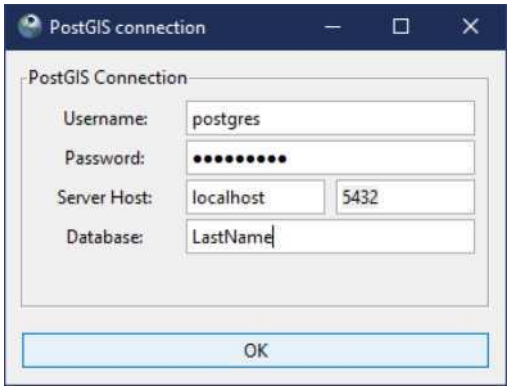


Рис. 5.7. Параметры подключения к СУБД PostgreSQL

Затем по нажатию кнопки *Add file* указать сконвертированный shp- файл и нажать кнопку *Import* (рис. 5.8).

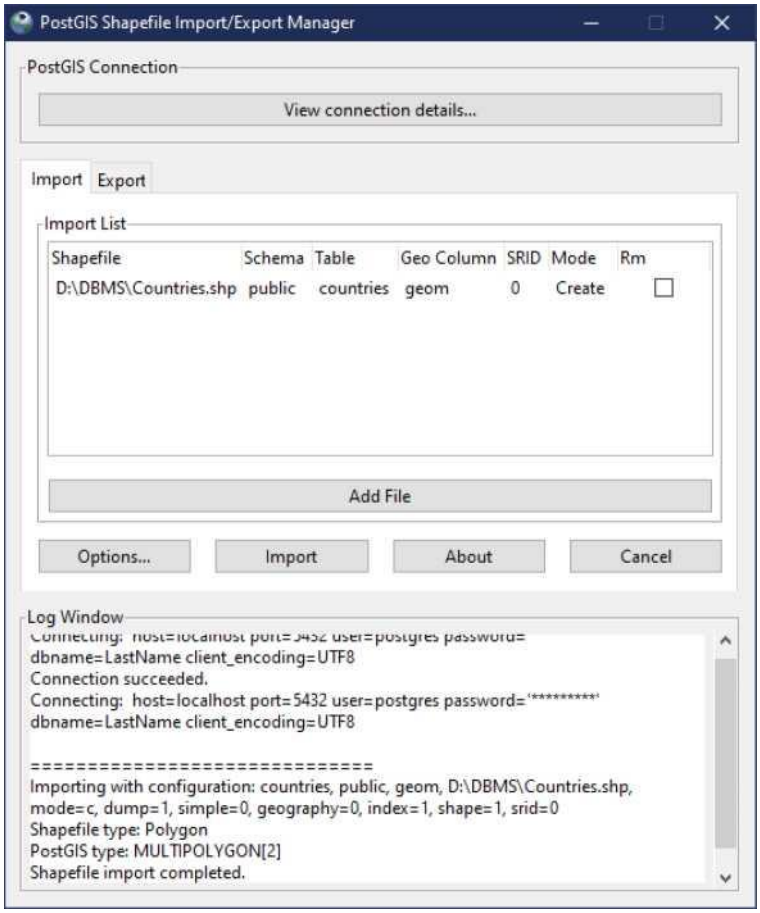


Рис. 5.8. Параметры загрузки SHP-файла в базу данных PostGIS

Для проверки корректности импорта созданную утилитой таблицу нужно открыть в QGIS. Для этого выбирается пункт главного меню *Слой - Добавить слой - Добавить слой PostGIS*. Затем создается новое соединение, для этого необходимо нажать кнопку *Создать*.

Пункты диалога заполняются в соответствии со скриншотом ниже (рис. 5.9), изменяется только имя базы данных.

После заполнения рекомендуется нажать на кнопку *Проверка соединения* и убедиться, что все параметры прописаны верно. Далее нужно нажать на кнопку *Подключиться* и дождаться появления списка схем данных. Затем развернуть схему данных и выбрать в списке таблицы с контурами стран (рис. 5.10).

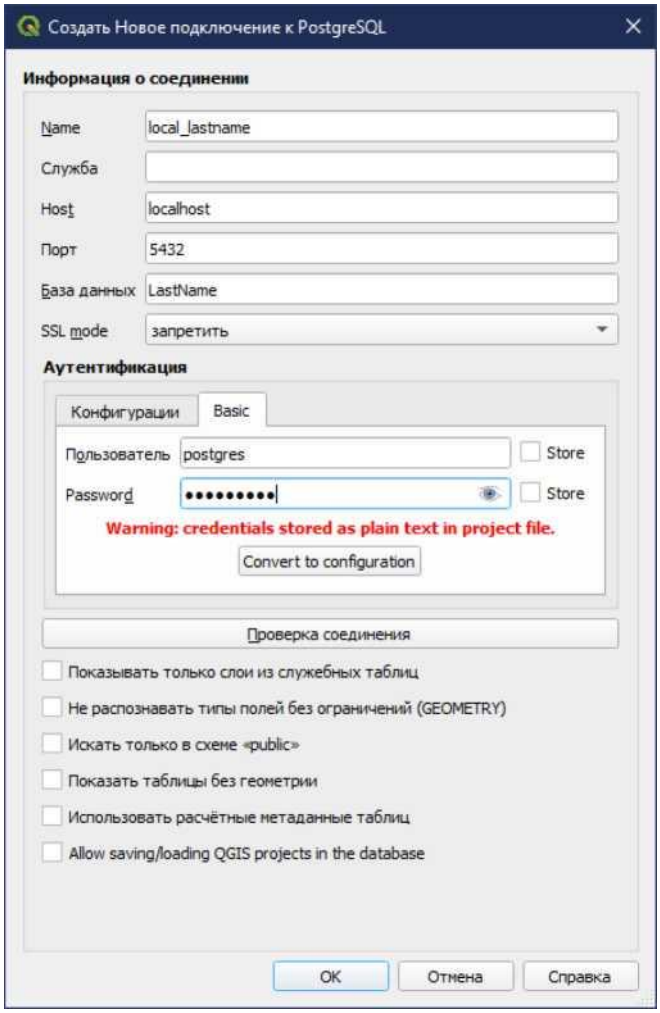


Рис. 5.9. Параметры соединения с сервером PostgreSQL

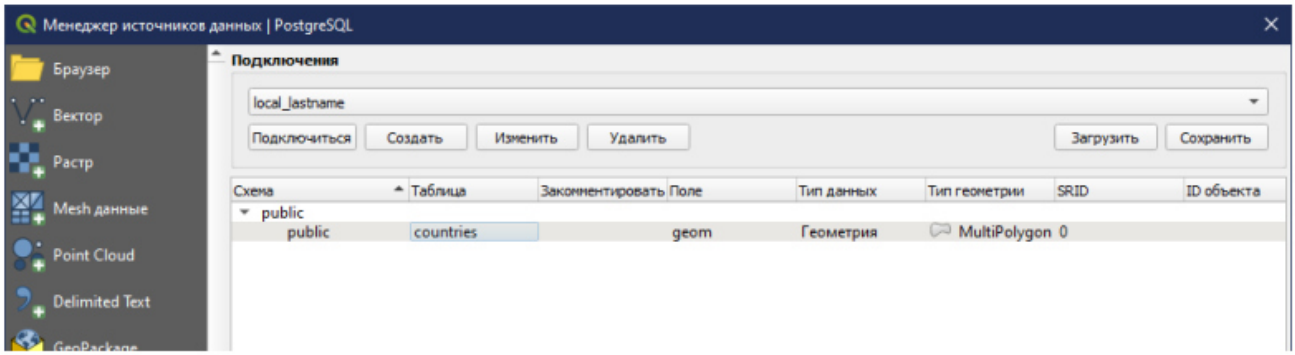
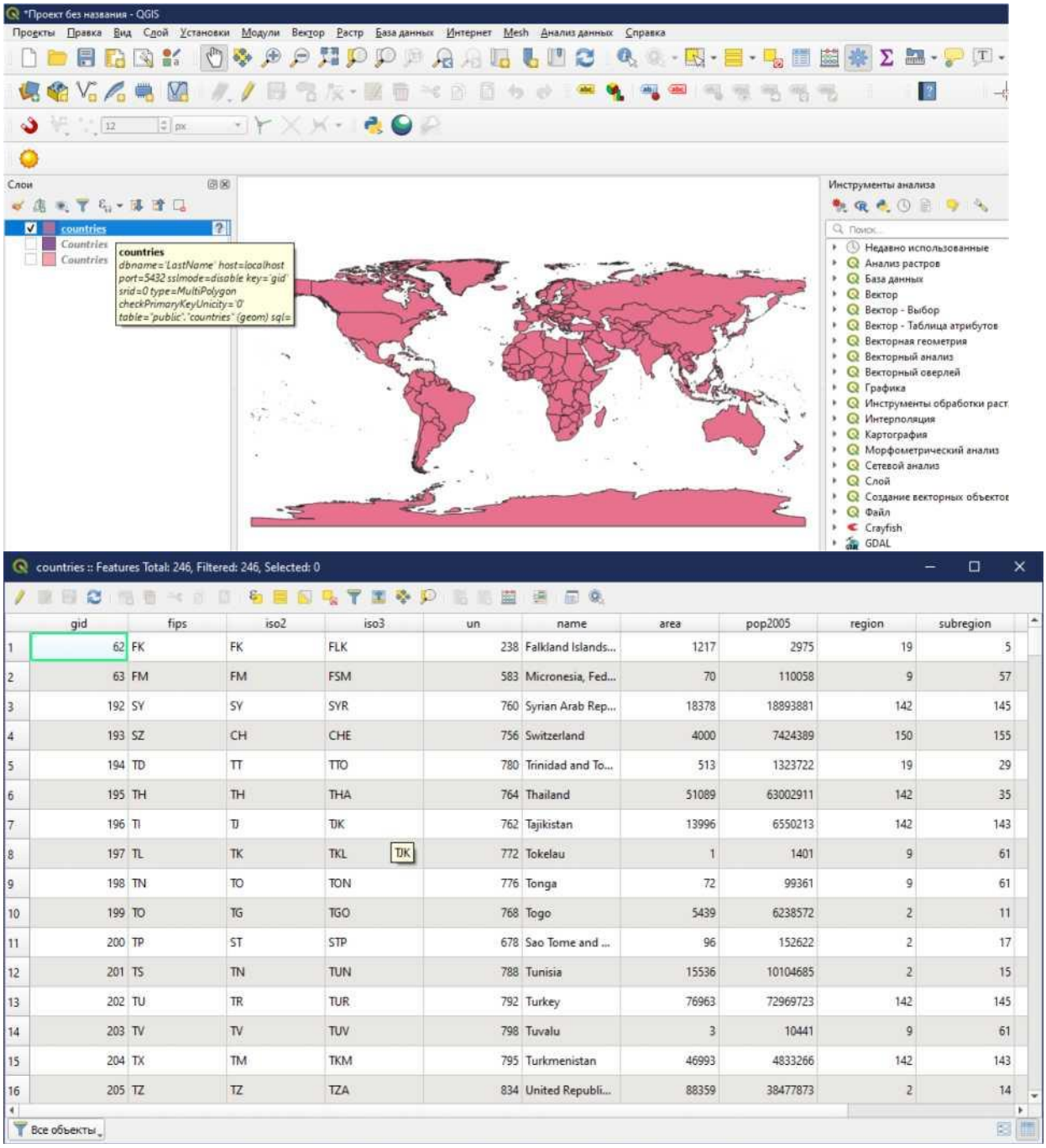


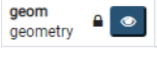
Рис. 5.10. Выбор слоя PostGIS для отображения

Убедиться, что геометрия и семантика отображаются корректно (рис. 5.11).



	gid	fips	iso2	iso3	un	name	area	pop2005	region	subregion
1	62	FK	FK	FLK	238	Falkland Islands...	1217	2975	19	5
2	63	FM	FM	FSM	583	Micronesia, Fed...	70	110058	9	57
3	192	SY	SY	SYR	760	Syrian Arab Rep...	18378	18893881	142	145
4	193	SZ	CH	CHE	756	Switzerland	4000	7424389	150	155
5	194	TD	TT	TTO	780	Trinidad and To...	513	1323722	19	29
6	195	TH	TH	THA	764	Thailand	51089	63002911	142	35
7	196	TI	TJ	TJK	762	Tajikistan	13996	6550213	142	143
8	197	TL	TK	TKL	772	Tokelau	1	1401	9	61
9	198	TN	TO	TON	776	Tonga	72	99361	9	61
10	199	TO	TG	TGO	768	Togo	5439	6238572	2	11
11	200	TP	ST	STP	678	Sao Tome and ...	96	152622	2	17
12	201	TS	TN	TUN	788	Tunisia	15536	10104685	2	15
13	202	TU	TR	TUR	792	Turkey	76963	72969723	142	145
14	203	TV	TV	TUV	798	Tuvalu	3	10441	9	61
15	204	TX	TM	TKM	795	Turkmenistan	46993	4833266	142	143
16	205	TZ	TZ	TZA	834	United Republi...	88359	38477873	2	14

Рис. 5.11. Отображение слоя PostGIS в QGIS

Альтернативным вариантом просмотра является добавленный в 12-й версии PostgreSQL *Geometry Viewer*. Для этого сначала нужно отобразить данные из таблицы *Countries*: развернуть поочередно списки *Имя_базы_данных - Schemas - Public - Tables,* щелкнуть правой кнопкой по таблице *Countries* и выбрать пункт *View/Edit Data - All Rows*. После этого в основном окне будет выведен запрос и его результаты. Для отображения контуров стран нужно пролистать вправо до колонки с координатами объектов (*geom*) и в ее заголовке нажать на значок просмотра 

(рис. 5.12).

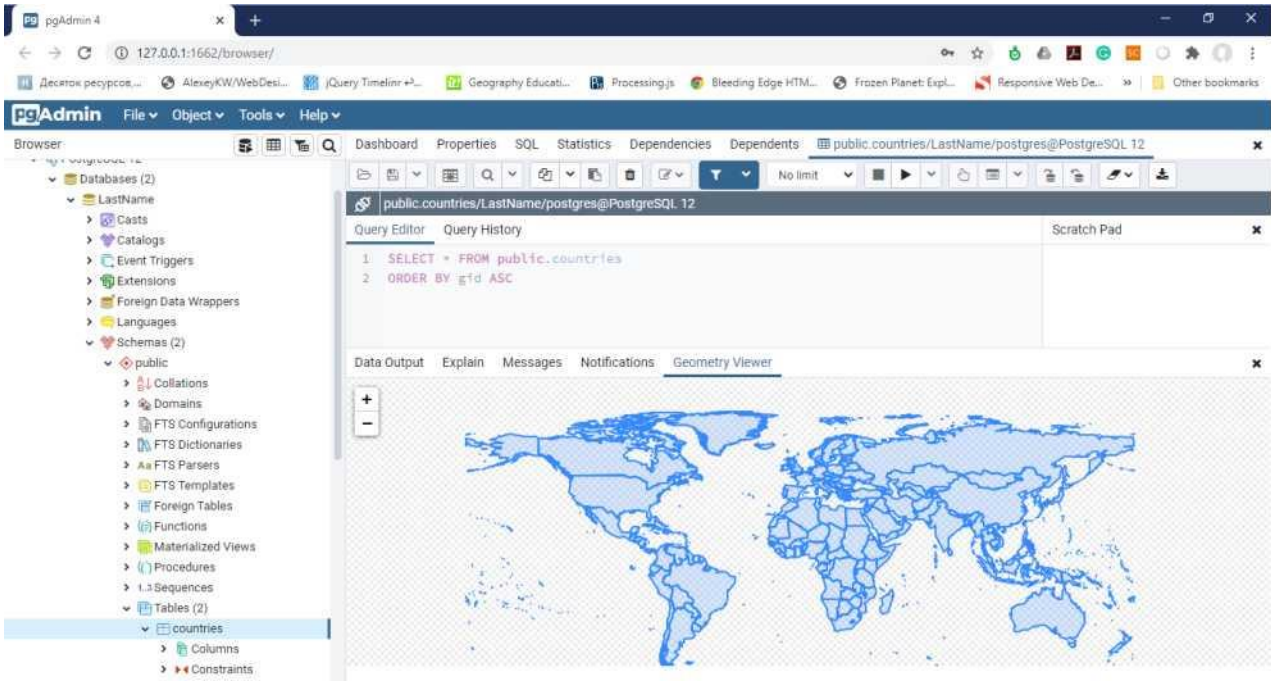


Рис. 5.12. Отображение пространственных данных в pgAdmin

Импорт внешних таблиц в базу данных

Следующим шагом нужно добавить информацию из Excel-файла *WDI.xlsx*. Для этого основной лист с данными нужно сохранить в формате CSV (при сохранении также требуется, чтобы в пути не было кириллических символов). Перед выполнением операций импорта в сохраненном ранее CSV-файле нужно заменить запятую на точку, которая будет являться разделителем целой и дробной частей в значениях показателей (рекомендуется вместо стандартного блокнота использовать Notepad++ или аналогичные программы).

После сохранения нужно создать структуру таблицы для импорта данных. Для этого в контекстном меню пункта *Tables* выбирается пункт *Create* и затем *Table* (рис. 5.13).

Название таблицы рекомендуется использовать такое же, как и у исходного Excel-файла. Названия колонок (не рекомендуется использовать пробелы и спецсимволы) и типы данных на вкладке *Columns* указываются в соответствии со структурой исходного Excel-файла. Также для колонок с указанием года (с 1960 по 2017) нужно в начале добавить букву *y* (year), чтобы впоследствии упростить структуру запроса (иначе эти колонки в запросах нужно было бы обрамлять кавычками). После завершения создания структуры таблицы нужно нажать на кнопку *Save (*рис. 5.14*)*.

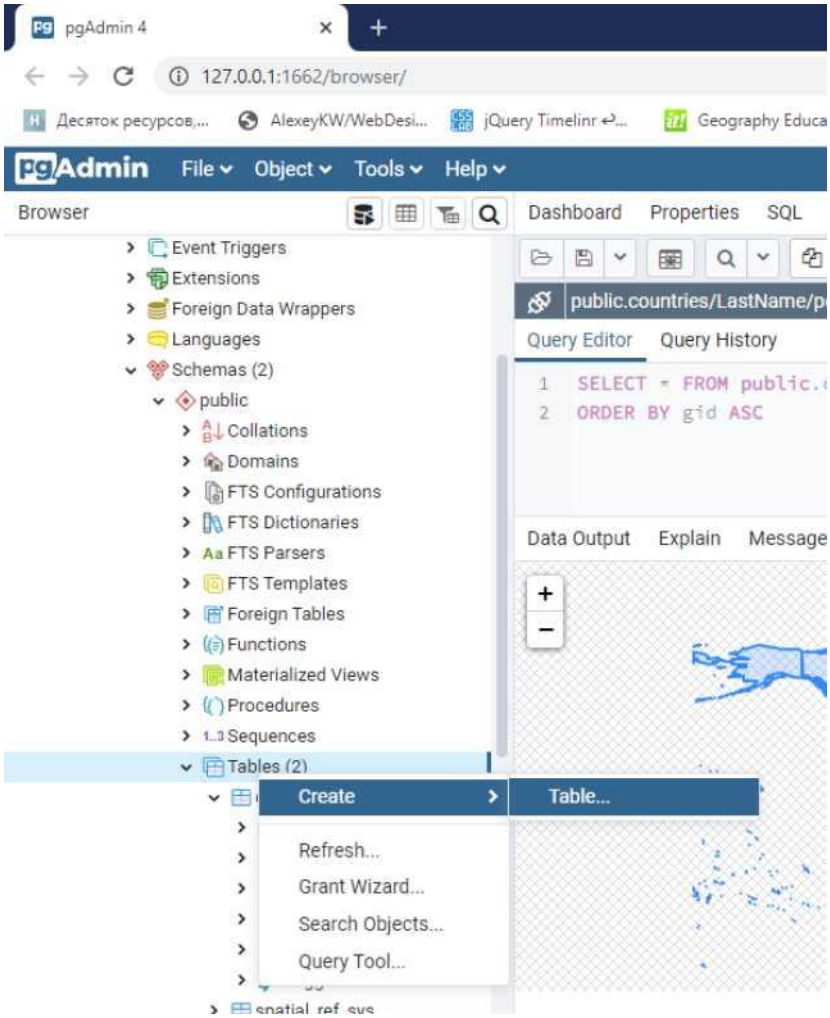


Рис. 5.13. Создание новой таблицы в pgAdmin

Для импорта данных из CSV-файла используется команда *COPY* с указанием имени таблицы, куда будут записаны данные, имени файла-источника, разделителя колонок и наличия заголовка в файле. Команда указывается в новом окне запроса *Tools - Query Tool*. Также в указании пути вместо одинарного используется двойной слэш (рис. 5.15).

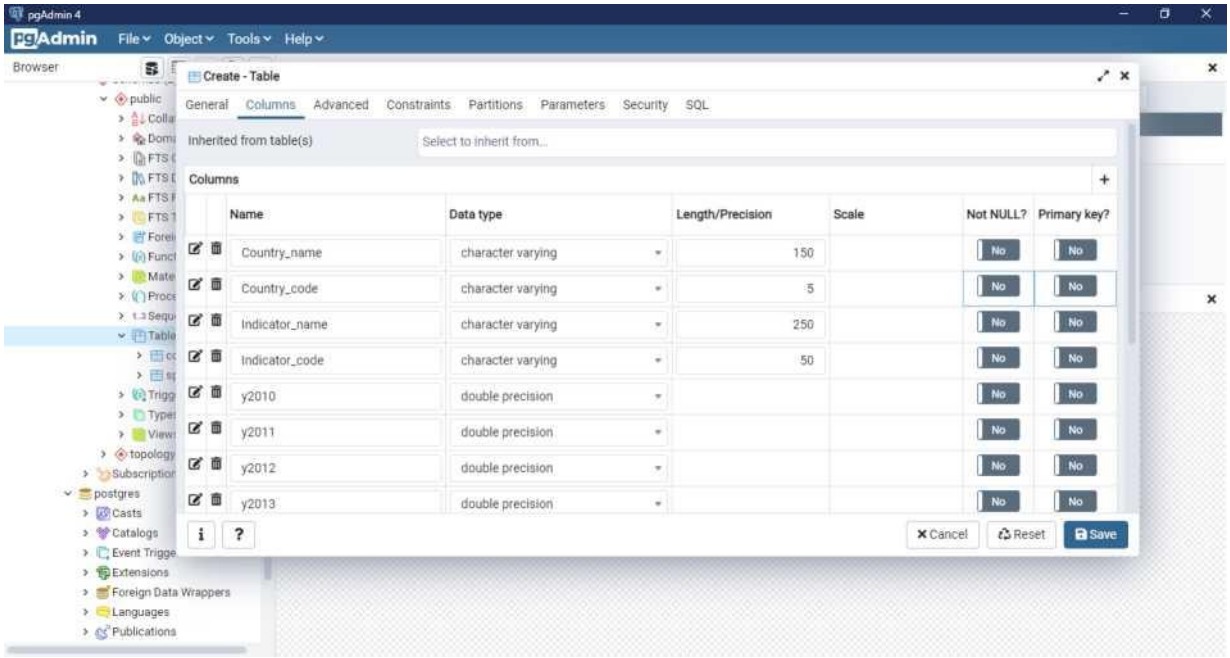


Рис. 5.14. Настройка колонок для создаваемой таблицы

Общая структура команды:

COPY "World_indicators" FROM '\\path\\to\\csv\\file' WITH CSV HEADER DELIMITER ';';

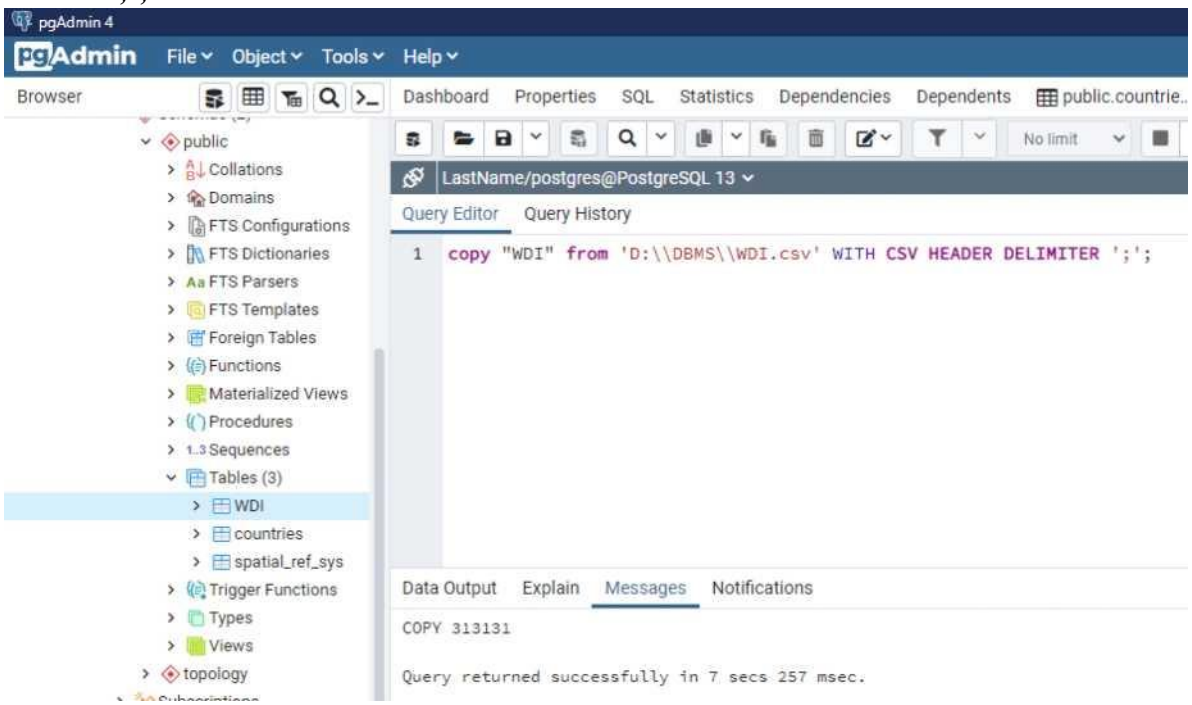


Рис. 5.15. Запрос для импорта CSV-файла в таблицу базы данных

После завершения импорта нужно проверить правильность записанных данных. В интерфейсе pgAdmin это можно сделать через контекстное меню *View / Edit Data* либо составив запрос *SELECT*.

Те же операции нужно сделать для CSV-файла со статистикой по странам мира (*Countries_of_the_world.csv*). Из названий столбцов нужно убрать спецсимволы, а сами названия можно сократить.

Важно обращать внимание на содержимое столбцов CSV-файла и

выставлять правильные типы данных у соответствующих колонок таблицы базы данных (рис. 5.16).

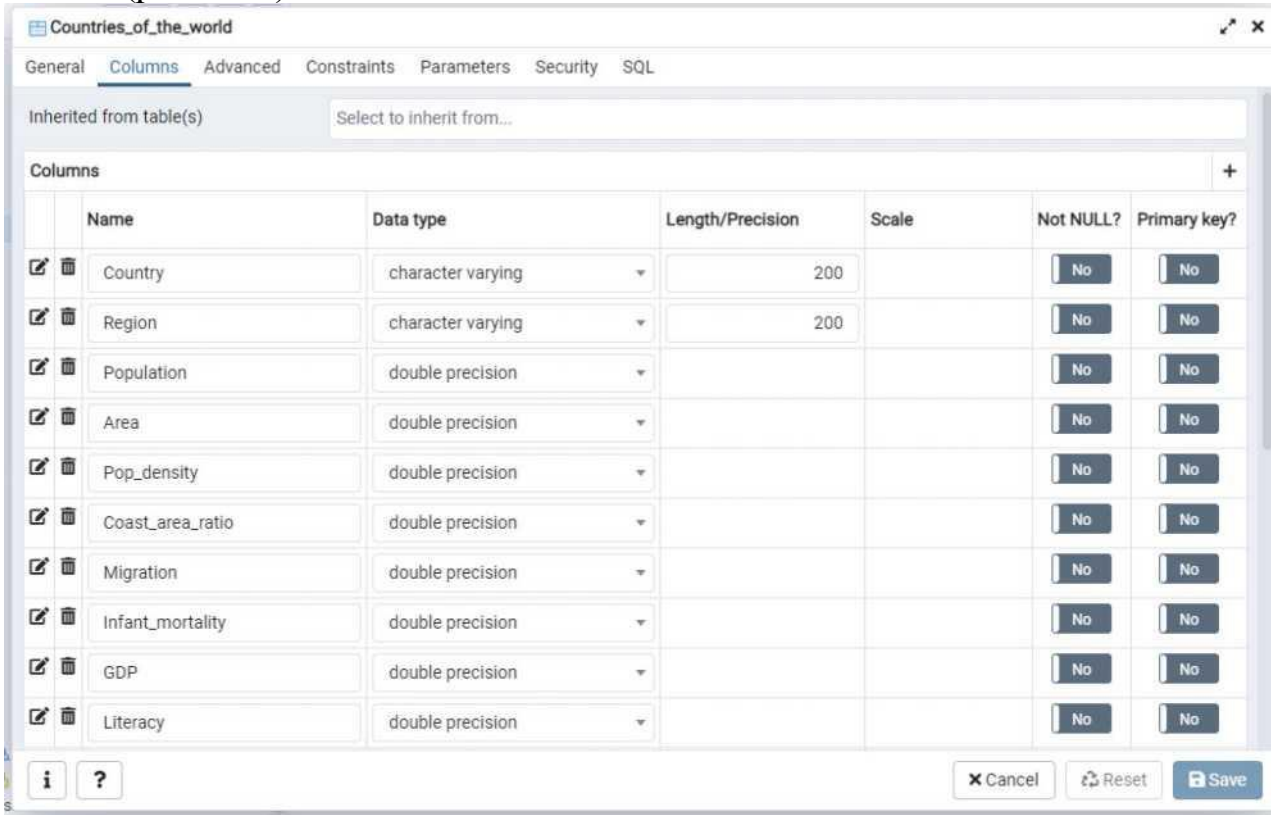


Рис. 5.16. Настройка колонок для таблицы Countries_of_the_world

Запрос на импорт данных аналогичен предыдущему, только разделителем колонок файла в данном случае является запятая; то есть нужно изменить параметр *DELIMETER* либо совсем его убрать (рис. 5.17).

После завершения импорта нужно проверить правильность записанных данных о странах мира (рис. 5.18).

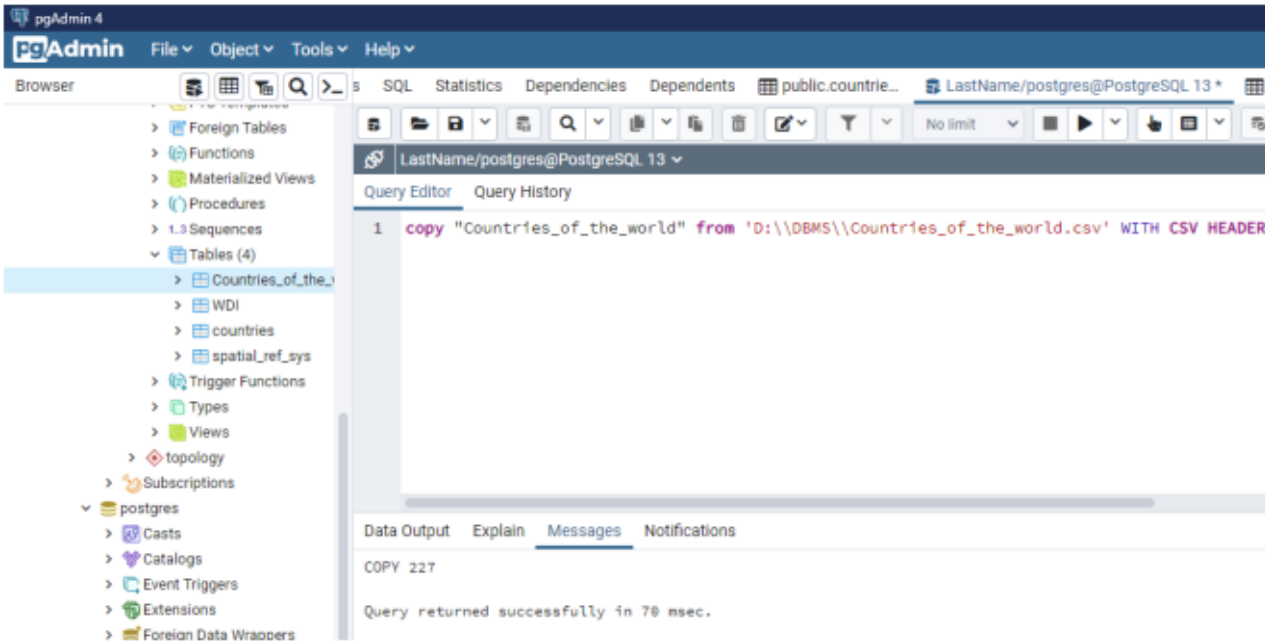


Рис. 5.17. Запрос для импорта файла *Countries_of_the_world.csv*

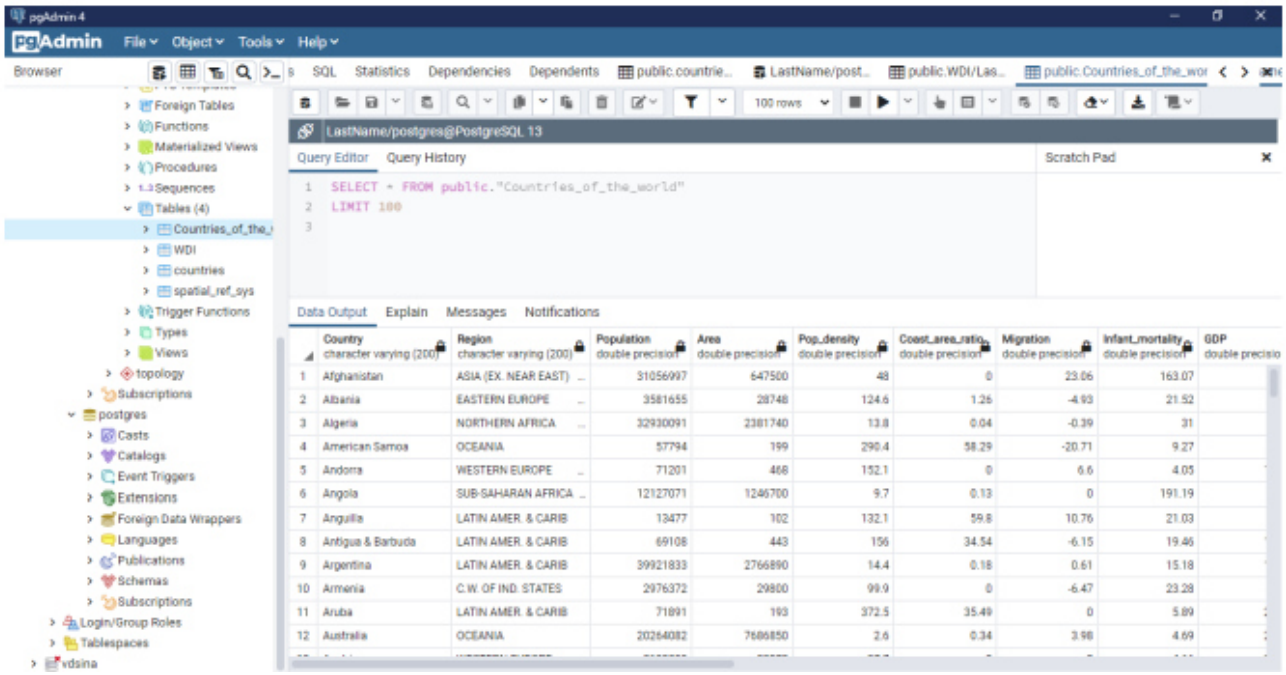


Рис. 5.18. Просмотр результатов импорта

Для импорта городов сначала выполняются стандартные операции по импорту CSV-файла, аналогичные предыдущим. Нужно также обратить внимание на то, что в этом файле разделителем колонок является запятая.

Затем нужно создать в таблице с городами столбец с геометрий точечной локализации (тип столбца *geometry*) и указанием EPSG-кода системы координат (рис. 5.19):

ALTER TABLE “Cities” ADD COLUMN geom geometry(Point, 4326);

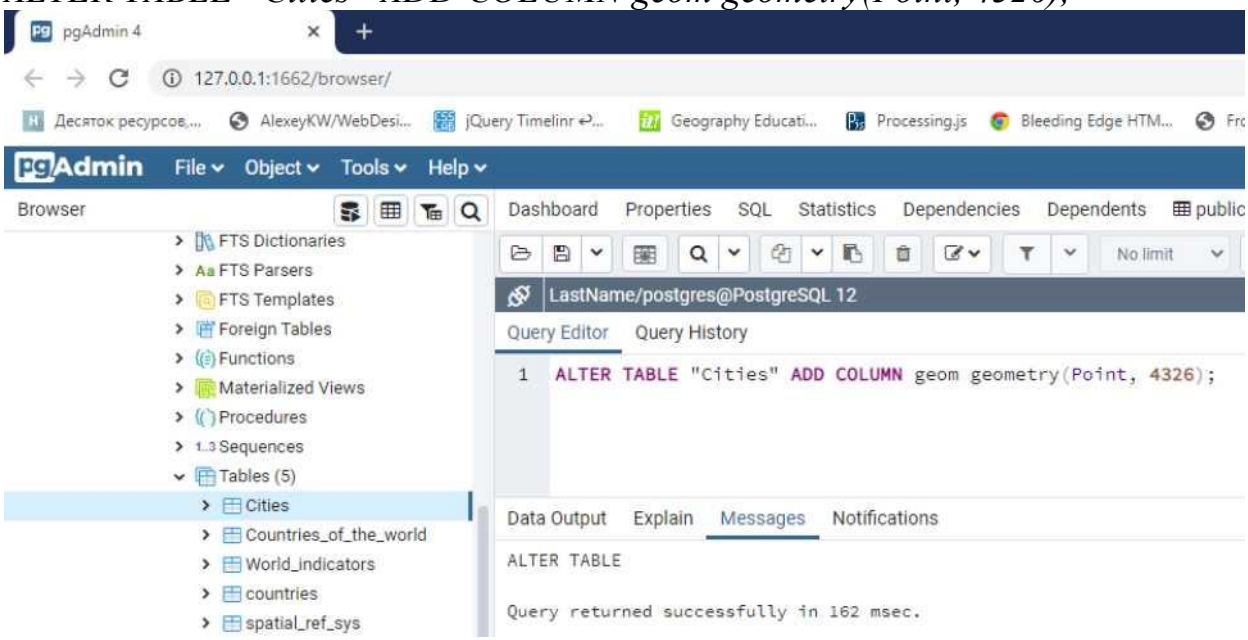


Рис. 5.19. Запрос на создания столбца для хранения координат объектов

Сами объекты создаются командой *ST_MakePoint* с выбором колонок, содержащих координаты, и также указанием системы координат (рис. 6.20).

UPDATE “Cities” SET geom = ST_SetSRID(ST_MakePoint("Longitude", "Latitude"), 4326);

Для проверки правильности создания точечных объектов также нужно воспользоваться QGIS или Geometry Viewer (во втором случае из-за большого количества объектов рекомендуется для отображения брать первые или последние 100 объектов *View/Edit Data - First/Last 100 Rows)* (рис. 5.21).

Для загрузки данных о демографии из файла SQLite можно воспользоваться генератором SQL-скриптов QGIS. Для его использования нужно вызвать контекстное меню слоя и выбрать пункт *Экспорт - Сохранить объекты как*. В открывшемся диалоге указать формат *PostgreSQL SQL дамп*, указать имя и местоположение создаваемого файла (рис. 5.22).

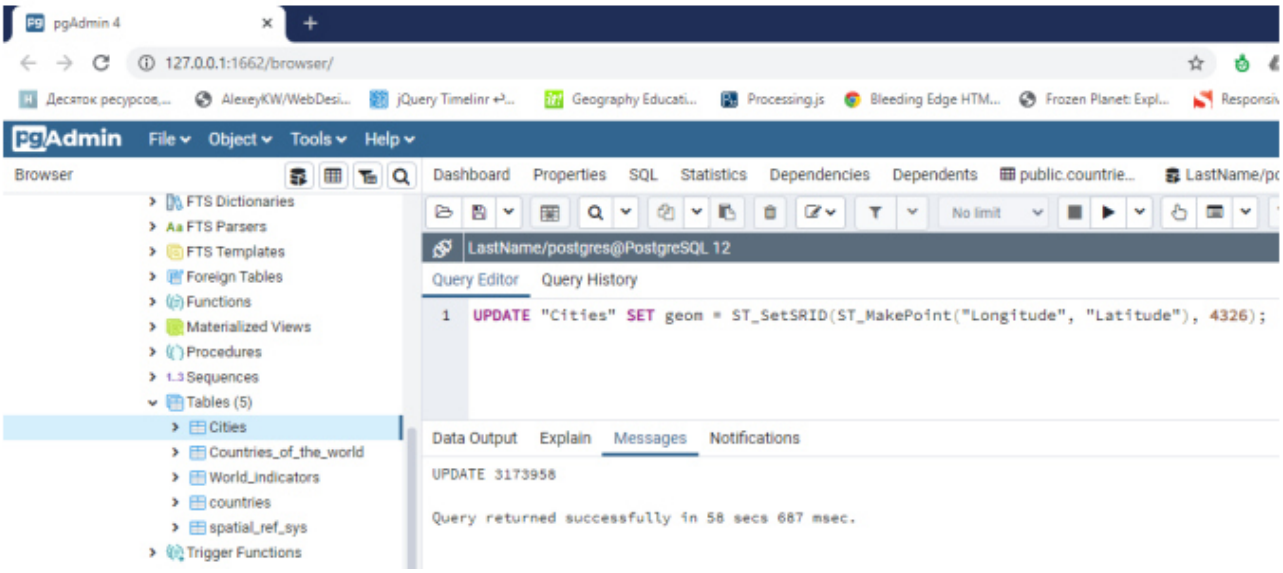


Рис. 5.20. Запрос на создание точечных объектов на основе данных в таблице

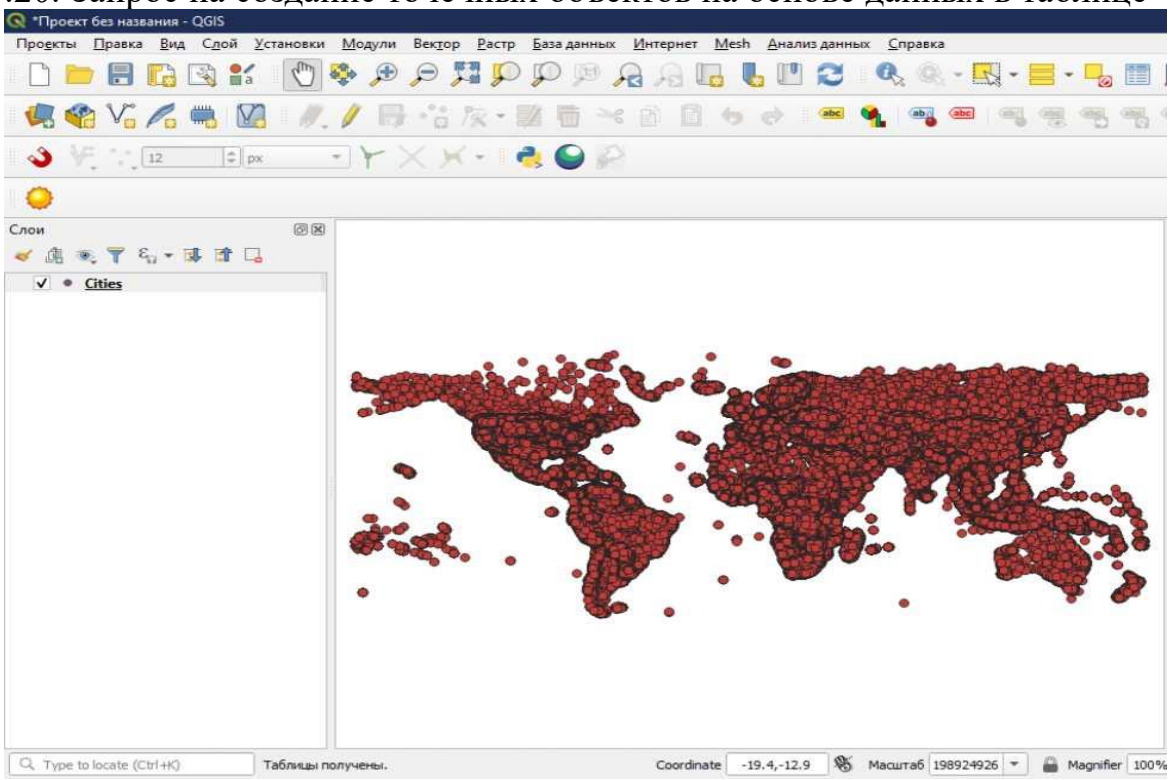


Рис. 5.21. Проверка корректности создания точечных объектов в QGIS

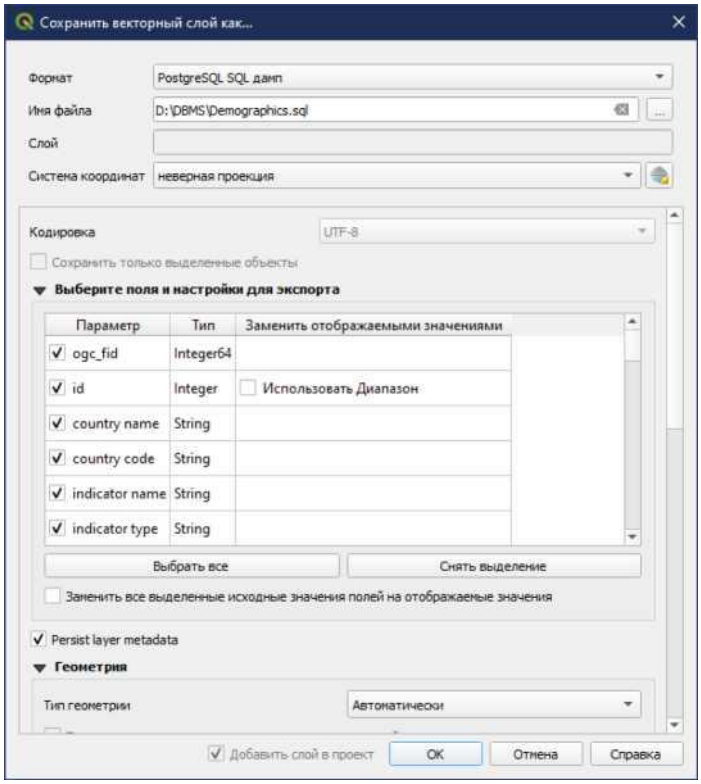


Рис. 5.22. Параметры выгрузки таблицы *Demographics* в SQL-файл

После сохранения файла нужно перейти в pgAdmin, открыть новое окно запросов (*Query Tool)* и загрузить SQL-скрипт с помощью кнопки *Open File* (рис. 5.23).

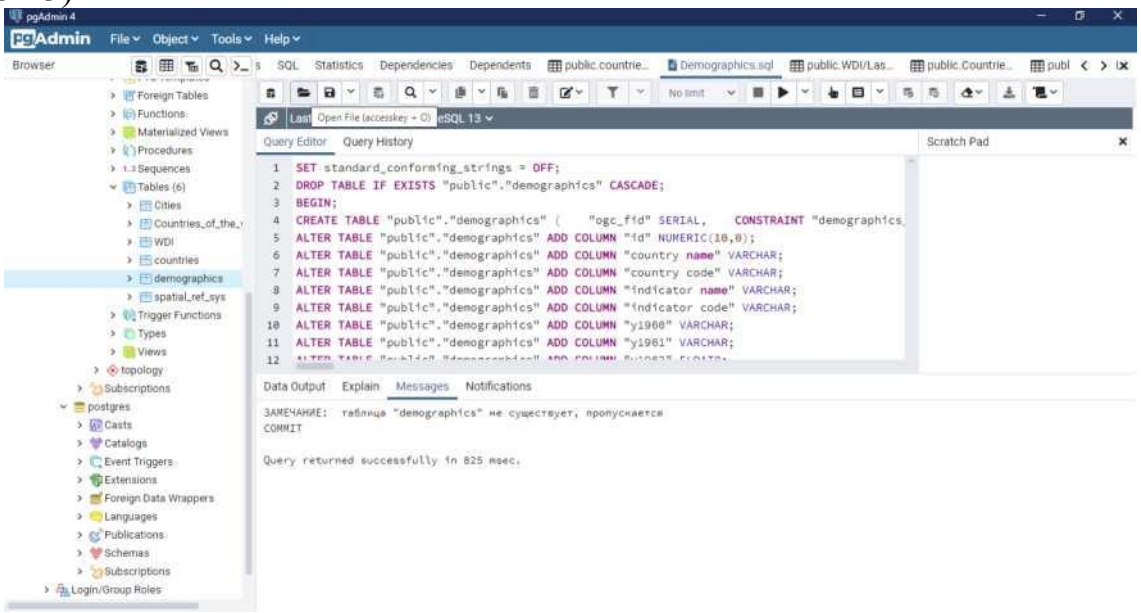


Рис. 5.23. Загрузка SQL-скрипта в pgAdmin

Затем выполнить скрипт и проверить правильность структуры таблицы и данных (рис. 5.24).

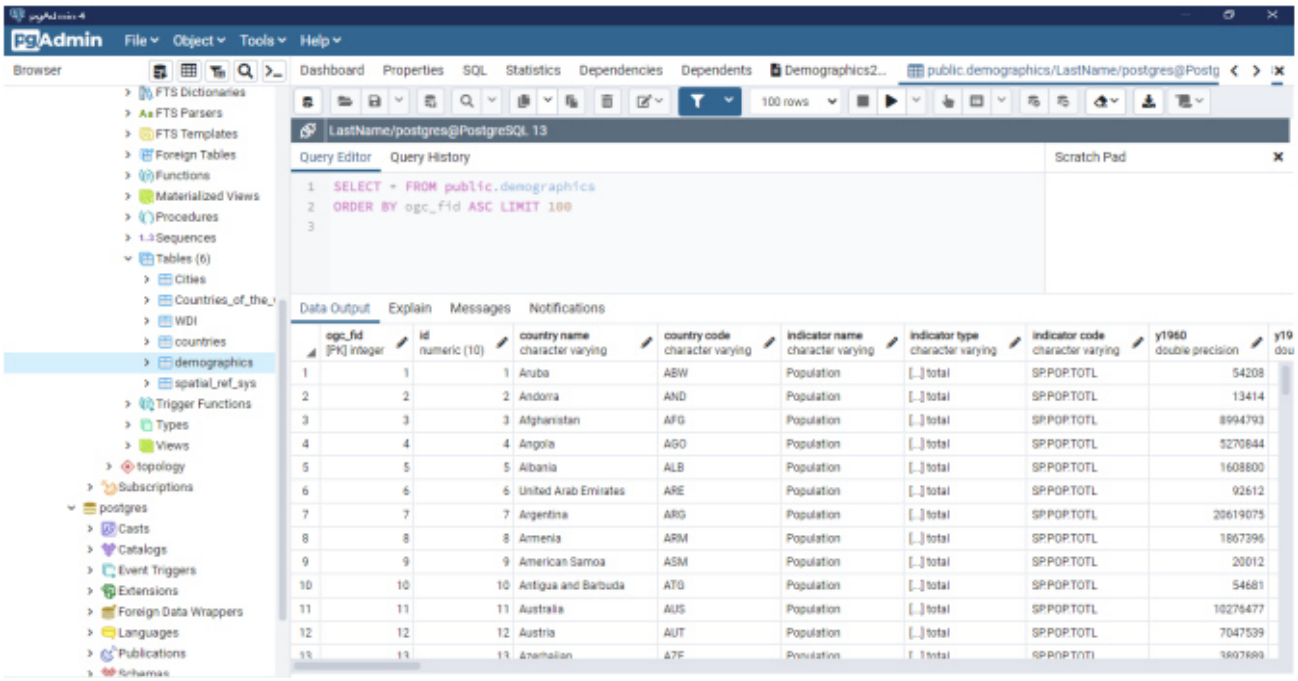


Рис. 5.24. Проверка результатов импорта таблицы *Demographics* из SQL-файла

Оценка производительности запроса и создание индексов

Составить запрос на выборку всех населенных пунктов с нулевым населением и оценить скорость выполнения этого запроса, используя *EXPLAIN ANALYZE*. Для этого нужно указать ключевые слова *EXPLAIN ANALYZE* перед текстом запроса. Дополнить результаты аналогичного запроса из таблицы практической работы № 1, указав значения Planning time и Execution time (табл. 5.1).

Таблица 5.1 - Статистика времени выполнения запроса

Программное обеспечение и параметры выполнения	Время выполнения запроса на поиск городов с ненулевым населением, с
QGIS	...
PostgreSQL Planning time	
PostgreSQL Execution time	

Далее создадим индекс для таблицы с городами и оценим влияние индексов на скорость выполнения запроса. Для этого в контекстном меню категории *Indexes* выбираем *Create - Index* (рис. 5.25).

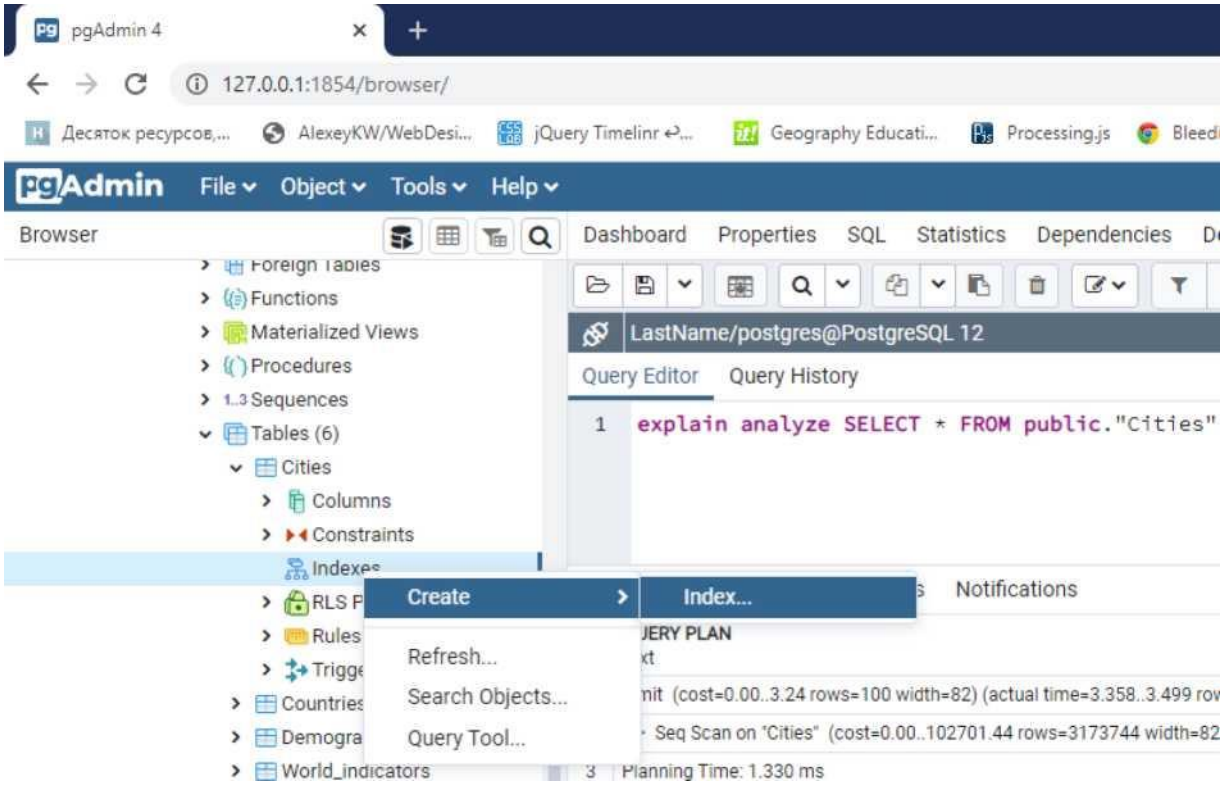


Рис. 5.25. Контекстное меню создания индекса

В открывшемся диалоге указывается название индекса. Затем на вкладке *Definition* выбирается тип индекса (нужно оставить стандартный *B-Tree*) и колонка, для которой создается индекс. В данном случае это колонка с количеством населения города - *Population* (рис. 6.26).

Оценить скорость выполнения предыдущего запроса на поиск населенных пунктов с нулевым населением после создания индекса, также используя *EXPLAIN ANALYZE*, и дополнить табл. 6.2 новыми измерениями.

Для визуализации результатов запросов, выполняемых PostgreSQL в QGIS, можно использовать вставку данных в новые таблицы либо воспользоваться представлениями (View). Рассмотрим второй вариант.

Для отображения в QGIS представления с геоданными (также, как и сами исходные таблицы) должны содержать ключевое поле (*PRIMARY*

KEY). Первичный ключ к таблице с городами добавляется следующим запросом:

ALTER TABLE cities ADD COLUMN id SERIAL PRIMARY KEY;

Если при создании таблица была названа с большой буквы (например, *Cities*), то в запросах имя этой таблицы нужно брать в двойные кавычки (“*Cities*”) (рис. 5.27).

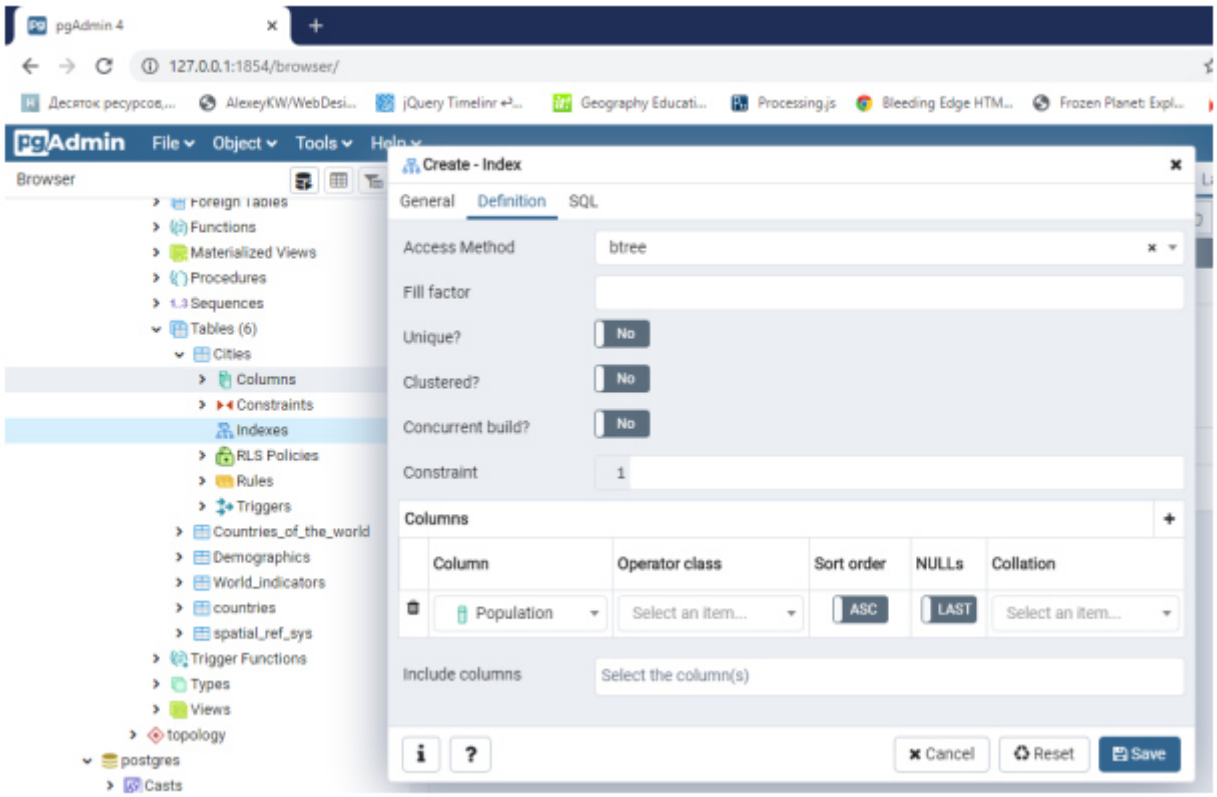


Рис. 5.26. Диалог настройки параметров индекса

Таблица 5.2 -Статистика времени выполнения запроса при использовании индексов

Программное обеспечение и параметры выполнения	Время выполнения запроса на поиск городов с ненулевым населением, с
QGIS	...
PostgreSQL Planning time	...
PostgreSQL Execution time	...
PostgreSQL Planning time после создания индекса	
PostgreSQL Execution time после создания индекса	

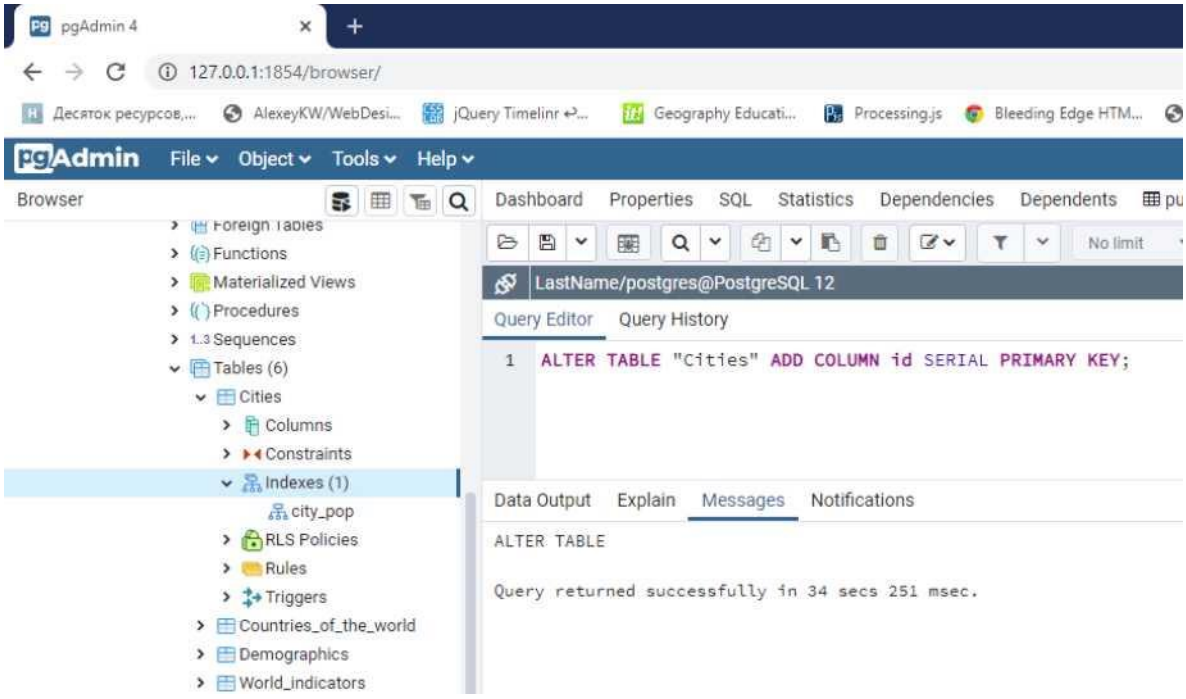


Рис. 5.27. Запрос на создание и автоматическое заполнение ключевого поля

Далее переходим к созданию представления: открываем контекстное меню в категории *Views,* далее *Create - View* (рис. 5.28).

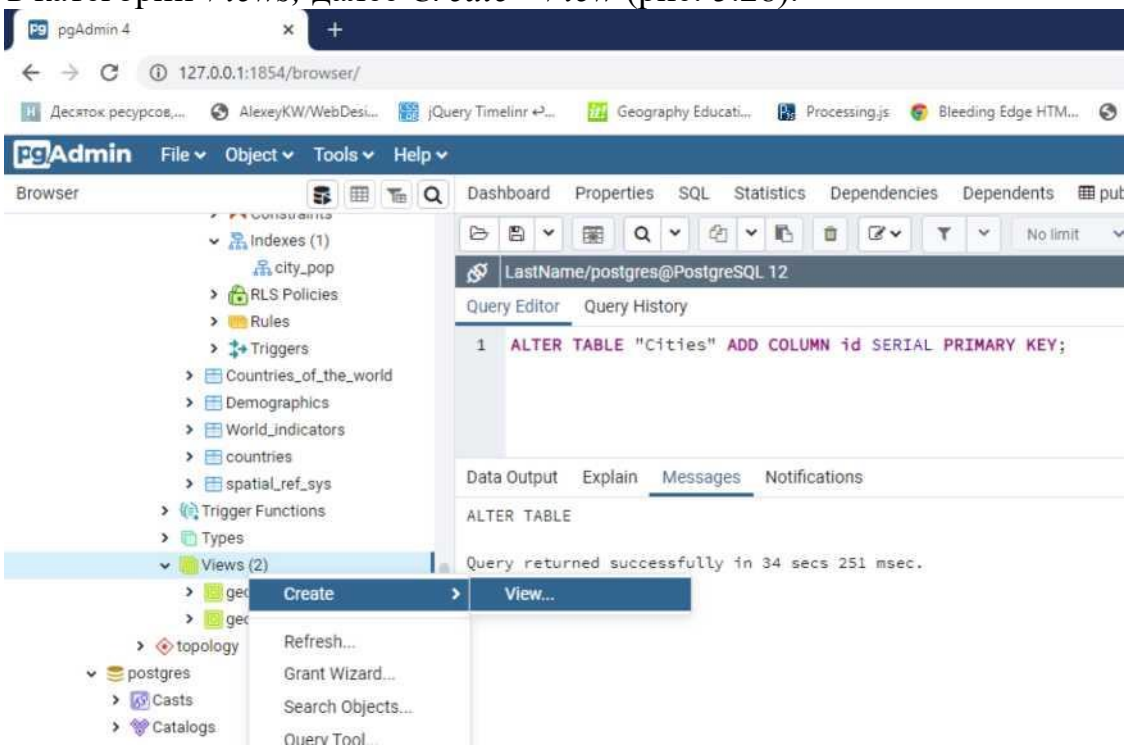


Рис. 5.28. Контекстное меню создания представления

Имя представления указывается произвольное. В поле *Code* прописывается запрос *SELECT* (рекомендуется заранее проверить его правильность в отдельном окне *Query Tool*) (рис. 5.29).

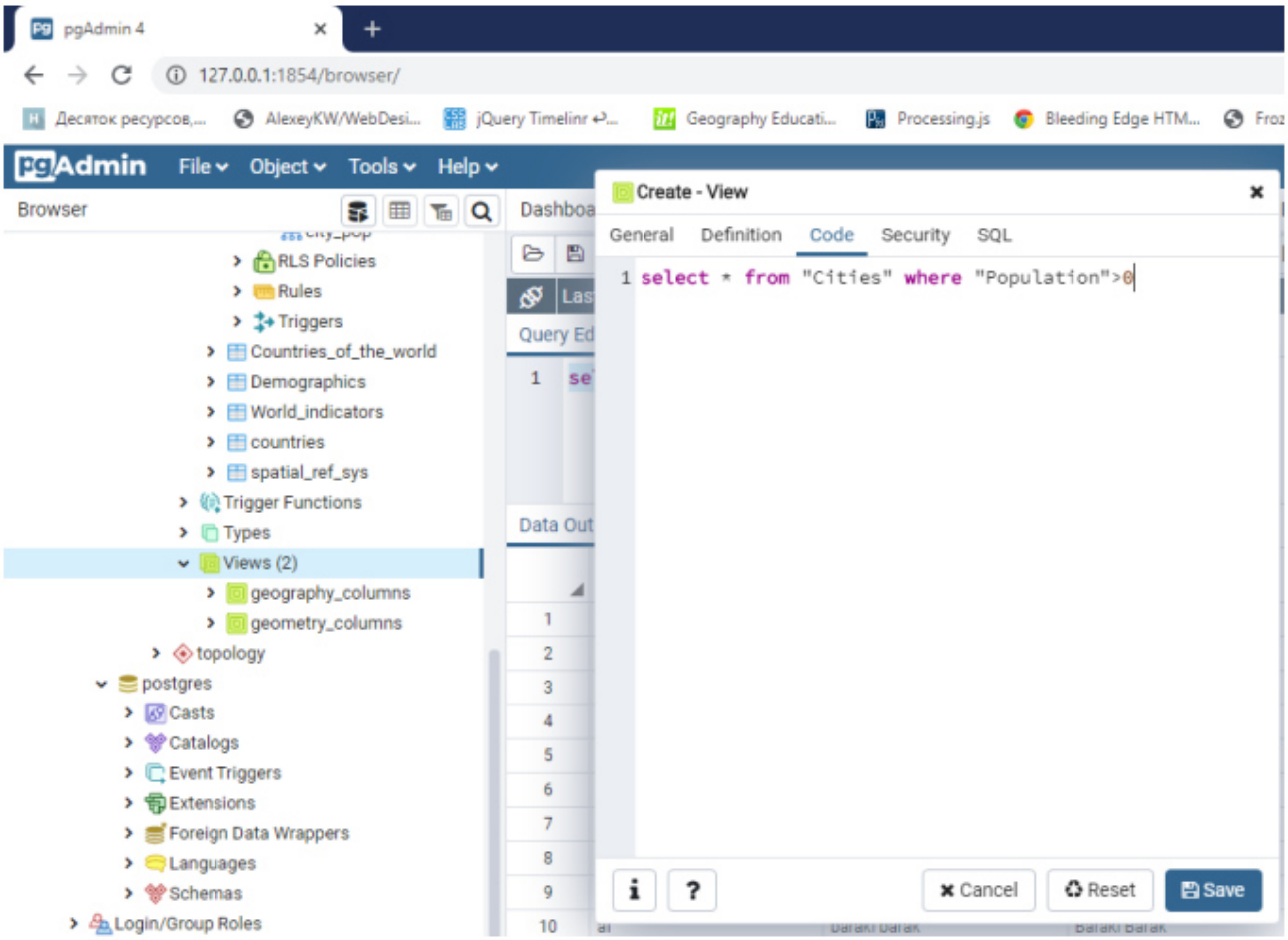


Рис. 5.29. Диалог указателя SQL-запроса для отбора данных представления

После подключения к базе данных в общем списке должно появиться созданное представление, и для добавления его как слоя QGIS нужно в списке *ID объекта* выбрать ранее созданный ключ (рис. 5.30 и 5.31).

Далее оценим влияние индексов на выполнение пространственных запросов. Для этого с помощью функции *ST_Contains* (или *ST_Within*) нужно выбрать города, которые имеют ненулевое население (из исходной таблицы *Cities*, а не из представления) и находятся на территории России (геометрия из таблицы *Countries*).

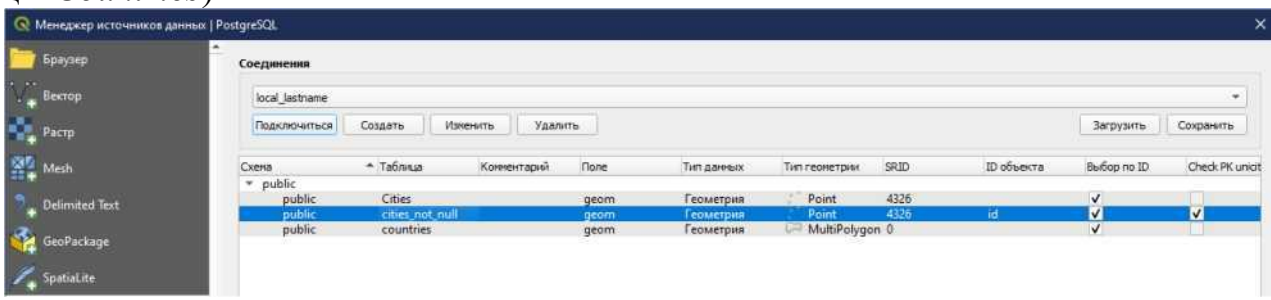


Рис. 5.30. Диалог настройки параметров подключения представления в QGIS

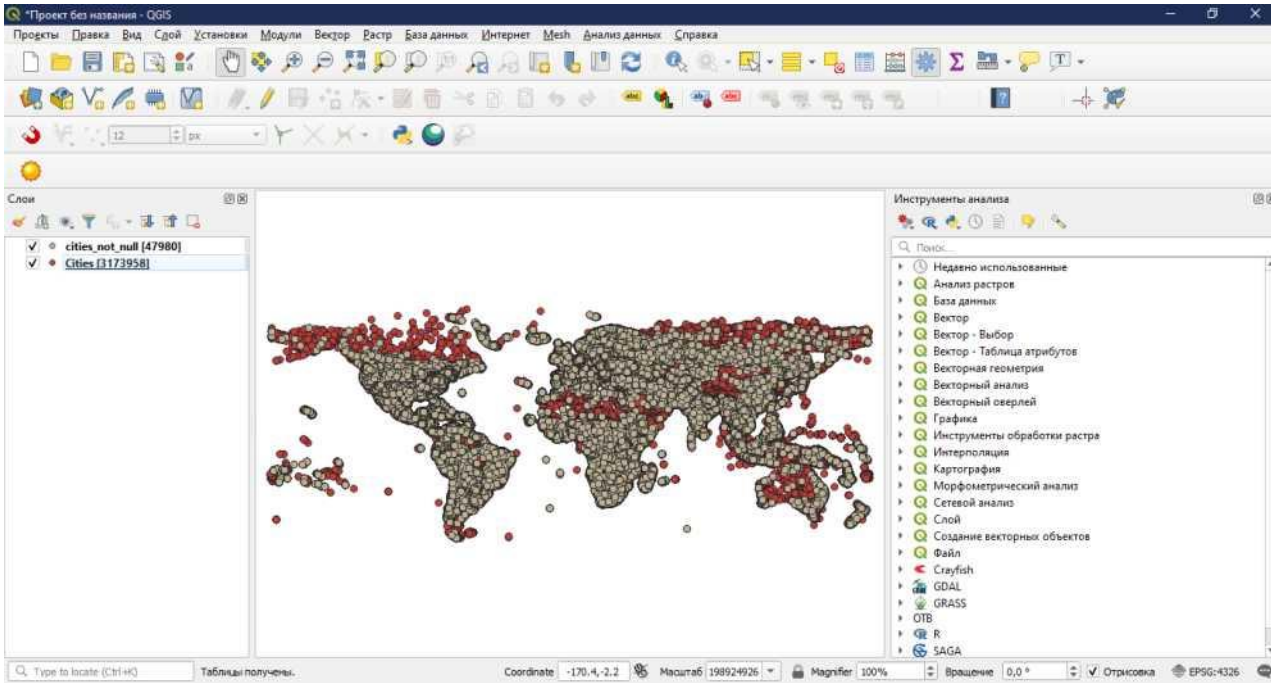


Рис. 5.31. Просмотр представления в QGIS

Для выполнения этого запроса требуется, чтобы у всех таблиц были указаны идентичные системы координат, поэтому у таблицы с контурами стран (*Countries*) потребуется с помощью запроса прописать конкретную систему координат WGS84 - *SELECT UpdateGeometrySRID('countries', 'geom',4326);*.

Запрос нужно выполнить три раза: для исходной таблицы; для таблицы с индексом *B-Tree*, созданным для колонки с геометрией; для таблицы с индексом *GIST*, созданным для колонки с геометрией. Результаты нужно занести в табл. 5.3.

Таблица 5.3 - Статистика времени выполнения запроса при использовании индексов

Программное обеспечение и параметры выполнения	Время выполнения запроса на поиск городов внутри указанной страны, с
QGIS	
PostgreSQL Planning time	
PostgreSQL Execution time	
PostgreSQL Planning time после создания индекса B-Tree	
PostgreSQL Execution time после создания индекса B-Tree	
PostgreSQL Planning time после создания индекса GIST	
PostgreSQL Execution time после создания индекса GIST	

Индивидуальные задания

Запрос № 1. Выбрать города с ненулевым населением в указанном радиусе от определенного города. Для расчета дистанции используется

функция *ST_DistanceSphere(поле_с_геометрией_анапизируемых_объектов, точка_отсчета).* Для использования вложенных запросов в функции *ST_DistanceSphere* их нужно брать в круглые скобки. Город и радиус поиска взять на основе номера варианта (табл. 5.4). В отчет нужно вставить скриншот, текст составленного запроса и таблицу с результатами его работы.

Таблица 5.4 - Варианты для запроса № 1

Номер варианта	Город	Радиус поиска, км
1	Санкт-Петербург	100
2	Владимир	200
3	Рига	85
4	Прага	90
5	Мюнхен	95
6	Милан	70
7	Ташкент	90
8	Красноярск	150
9	Анкара	150
10	Марракеш	120
11	Бахрейн	150
12	Сан-Паулу	105
13	Буэнос-Айрес	115
14	Лима	300
15	Богота	145
16	Порту	105
17	Барселона	80
18	Бейрут	150
19	Верона	85
20	Триест	120
21	Сараево	100
22	Варна	90
23	Самарканд	80
24	Гуанчжоу	50
25	Бангкок	145
26	Ченнай	120
27	Бангалор	90
28	Карачи	135
29	Катар	135
30	Сан-Марино	75

Представление № 1. Составить запрос и создать на его основе представление, которое будет содержать следующие данные:

- полигон, созданный с помощью объединения всех стран (их полигонов), относящихся к указанному региону;
- среднее значение указанного показателя для этого региона за определенный год.

Показатель, год и регион взять на основе номера варианта (табл. 6.8). В отчет нужно вставить скриншот и текст запроса, формирующего представление, таблицу с результатами его работы, скриншот из QGIS с визуализацией данных созданного представления.

Варианты для представления № 1

Номер варианта	Показатель	Регион мира	Год
1	Занятость в сельском хозяйстве (смоделированная оценка ILO), % от общей занятости	Northern America	2015
2	Продолжительность обязательного образования, лет	C.W. of Ind. States	2017
3	Использование альтернативной и ядерной энергии, % от общего потребления	Northern Africa	2014
4	Количество абонентов фиксированной телефонной связи (на 100 человек)	Northern America	2013
5	Процент подростков, не посещающих школу	Near East	2014
6	Сумма сборов за использование интеллектуальной собственности (платежный баланс в долларах США)	Northern Africa	2015
7	Годовой рост добавленной стоимости сельского, лесного и рыбного хозяйства, %	Latin Amer. & Carib	2012
8	Уровень грамотности всего взрослого населения для людей в возрасте 15 лет и старше, %	Baltics	2015
9	Количество видов птиц, находящихся под угрозой исчезновения	Northern America	2012
10	Производство электроэнергии из гидроэнергетических источников, % от общего объема	Baltics	2014
11	Доступ к электричеству для городской территории	Eastern Europe	2013
12	Доступ к чистому топливу и технологиям для приготовления пищи	Asia (ex. Near East)	2012
13	Производство зерновых, т	Northern Africa	2014
14	Площадь лесов, % от площади всех земель	Latin Amer. & Carib	2014
15	Пашни, га/чел.	C.W. of Ind. States	2017

Практическая работа 6. Анализ и визуализация данных в базе пространственных данных геоинформационных систем

Цель работы: усвоить понятие агрегирующих функций как одного из базовых понятий СУБД, способы визуализации данных в геоинформационных системах, принципы выполнения SQL-запросов, в том числе пространственных, к нескольким таблицам.

Задачи работы

1. Изучить:

- аналитические функции геоинформационных систем;
- способы визуализации статистических данных в геоинформацион- ных системах.

2. Научиться:

- выполнять агрегирующие операции к данным в геоинформационных системах;
- строить тематические карты в геоинформационных системах;
- выполнять пространственные запросы к нескольким таблицам.

Перечень обеспечивающих средств

Для выполнения работы необходимо иметь методические указания по выполнению работы, а также программное обеспечение QGIS (версия 3.10 и выше).

Составление запросов с агрегирующей функцией

В QGIS можно выполнять запросы к данным (в том числе и из файлов), составляя текстовые запросы на языке SQL либо используя графический интерфейс. Первый вариант находится в меню *Базы данных - Менеджер*

БД, далее кнопка *Окно SQL*. Для запуска второго варианта нужно проделать те же действия, что и для первого варианта, и потом нажать кнопку *Конструктор SQL-запросов.*

Агрегирующие функции расположены в категории *Обобщение* и позволяют проводить базовые статистические вычисления над данными числовых типов. Для примера оценим средний уровень грамотности по регионам на основе данных из таблицы *Countries_of_the_world*.

Для выполнения этой задачи используется агрегирующая функция *Avg* и группировка (поле *Group by*) по колонке *Region*. Обратите внимание на то, что для корректного выполнения запроса требуется использовать одинарные кавычки в названиях таблиц и колонок вместо двойных (рис. 6.1).

Текст SQL-запроса:

SELECT 'Countries_of_the_world'.'Region',
AVG('Countries_of_the_world'.'Literacy (%)')
FROM 'Countries_of_the_world'
GROUP BY 'Countries_of_the_world'.'Region';.

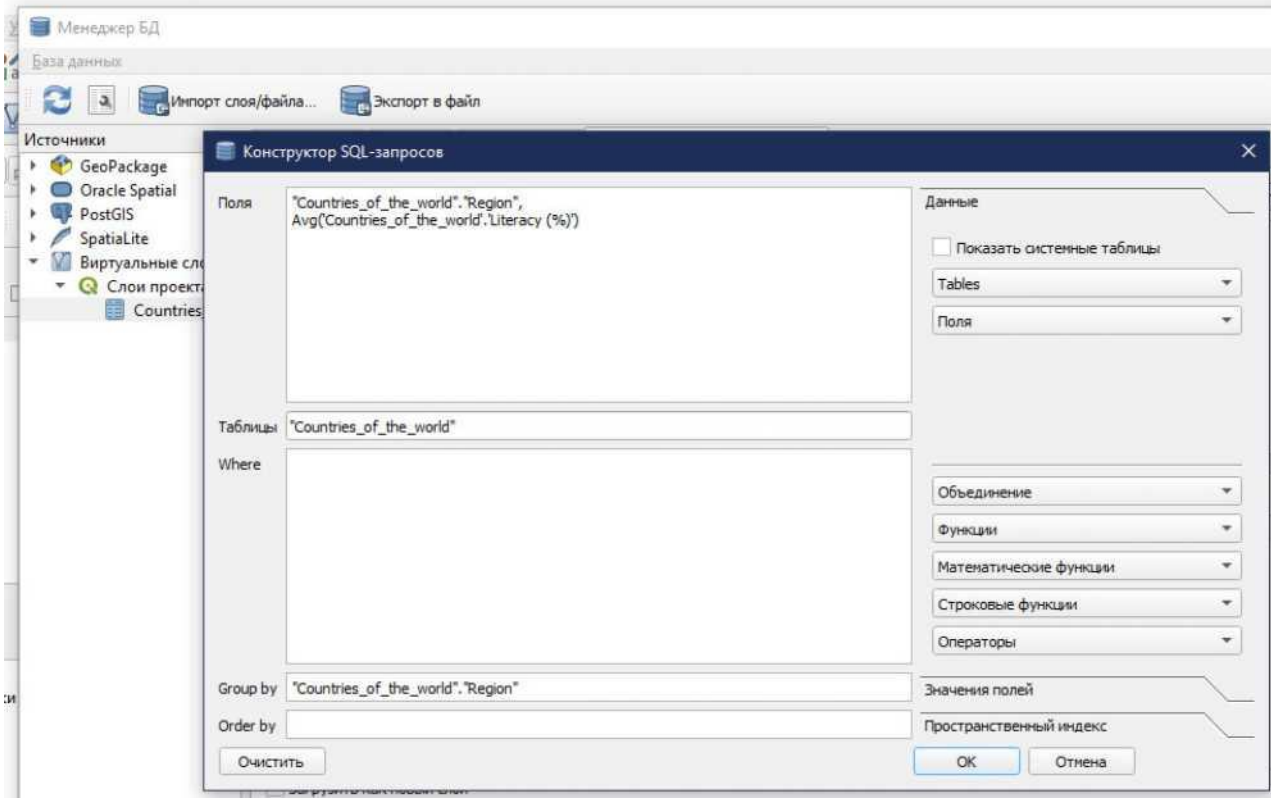


Рис. 6.1. SQL-запрос с агрегирующей функцией

В результате выполнения запроса получаем таблицу с названиями регионов и соответствующим им значениям показателей. Названия регионов и соответствующие им значения нужно внести в табл. 6.1.

Таблица 6.1

Результаты выполнения запроса расчета среднего уровня грамотности

Наименование региона	Средний уровень грамотности (%)

Далее составьте еще один запрос с использованием агрегирующей функции на примере подсчета количества стран, относящихся к определенному региону и отсортируйте полученный список по названию региона. Для вычисления количества используется функция *Count*, а для сортировки нужно указать необходимую колонку в поле *Order by*. В отчет нужно вставить текстовое выражение полученного запроса и заполнить табл. 6.2 в соответствие с полученными результатами.

Таблица 6.2

Результаты выполнения запроса вычисления количества стран по регионам

Наименование региона	Количество стран в регионе

Построение и выполнение запросов с расчетом количественных характеристик

Синтаксис SQL запросов допускает использование стандартных математических операций для расчета значений. В качестве примера рассчитаем плотность жителей на квадратный километр по странам путем деления количество жителей на вычисленную площадь полигона; данные используются из слоя *Countries*. Для расчета площади на основе метрики объектов используется стандартная функция *ST_Area* (список *Функции*).

Аргументом этой функции должна быть геометрия объектов (в QGIS это отдельное поле ' *geometry*').

Текст SQL-запроса:

SELECT 'Countries'.'NAME', 'Countries'.'POP2005' /
ST_Area('Countries'.geometry') FROM 'Countries'

В результате деления получается дробное число, что противоречит логике показателя, и для отображения результата в виде целого числа нужно дополнить запрос функцией *Round* с указанием нулевого значения в качестве параметра, показывающего количество знаков после запятой (рис. 6.2).

Текст SQL-запроса:

SELECT 'Countries'.'NAME', Round('Countries'.'POP2005' /
ST_Area('Countries'.geometry'), 0) FROM 'Countries'

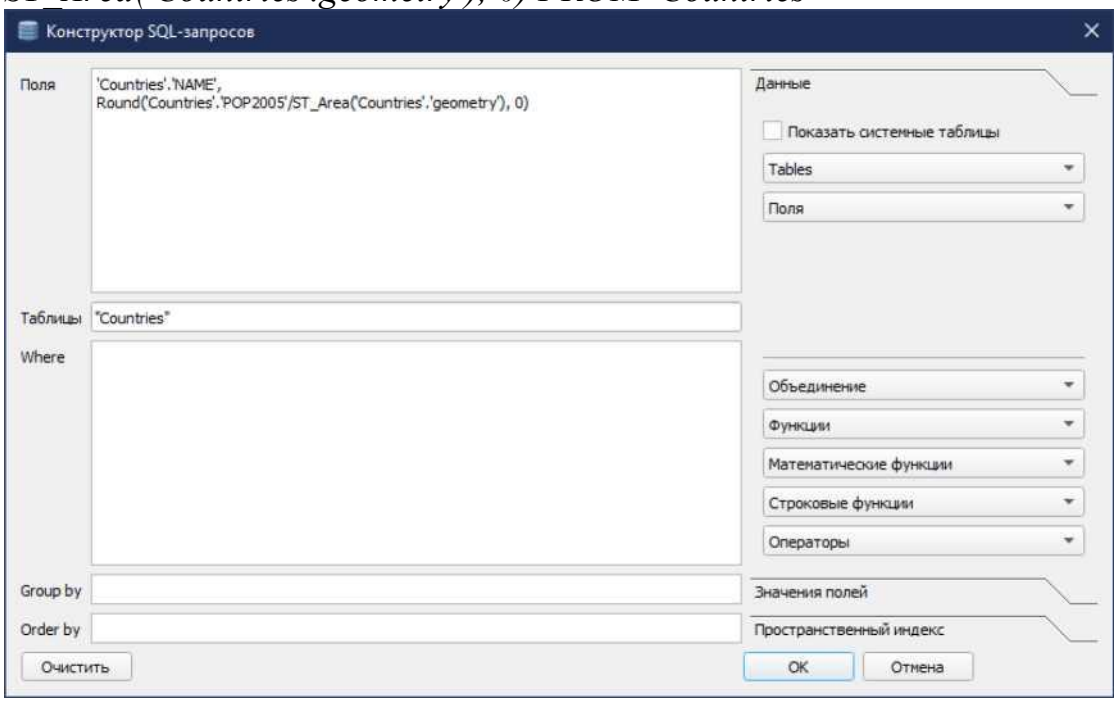


Рис. 6.2. Запрос с расчетом количественных пространственных характеристик и округлением результатов

Еще одним недостатком текущего варианта запроса является то, что в выводе, в заголовке столбца, указан полный текст запроса. Для задания произвольного заголовка колонки в результатах запроса в QGIS используются псевдонимы (aliases) с тем же синтаксисом, что и в стандарте SQL, то есть после колонки или вычисляемого значения в блоке *SELECT* нужно добавить

ключевое слово AS и в одинарных кавычках указать необходимое значение псевдонима. Для корректного представления результатов запроса с точки зрения человека нужно установить псевдоним для колонки с результатами расчета в виде значения *Density*.

Текст SQL-запроса:

SELECT 'Countries'.'NAME', Round('Countries'.'POP2005' / ST_Area('Countries'.geometry'), 0) As 'Density' FROM 'Countries'

В результате выполнения запроса должен получиться следующий результат (рис. 6.3).

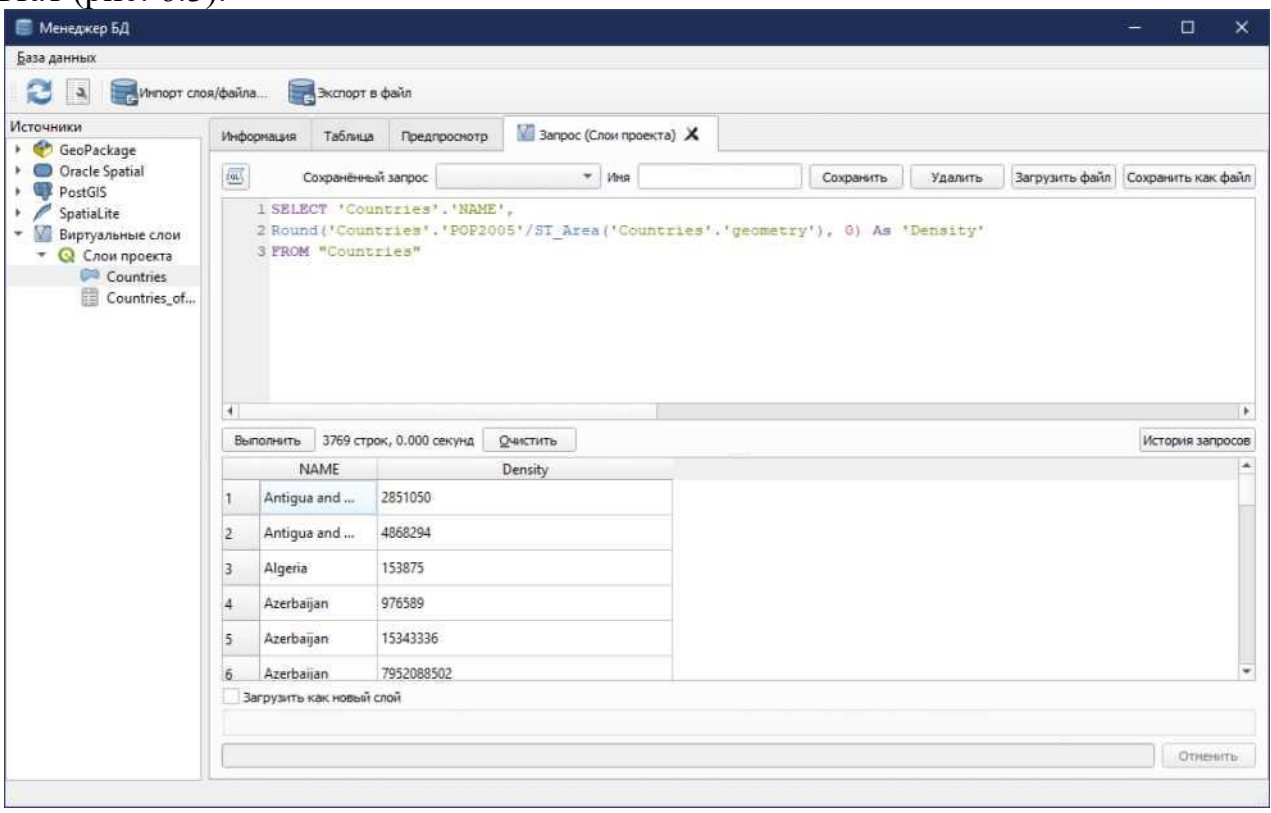


Рис. 6.3. Результаты запроса с расчетом количественных характеристик и группировкой результатов

Построение и выполнение пространственных запросов

Одно из основных отличий геоинформационных систем - наличие функций пространственного анализа и возможность выполнять пространственные запросы.

Например, выполним поиск всех городов, расположенных в 50 км от Москвы. В QGIS, при использовании стандартных инструментов, он выполняется в три шага.

Поскольку для геометрических вычислений QGIS использует единицы измерения слоя, то перед выполнением запросов слой с городами, полученный в первой работе, нужно сохранить с проекционной системой координат (например, *World Mercator*, *EPSG*: *3395*). Для этого используется пункт контекстного меню *Экспорт - Сохранить объекты как* (рис. 6.4). Для последующих запросов нужно будет использовать новый слой.

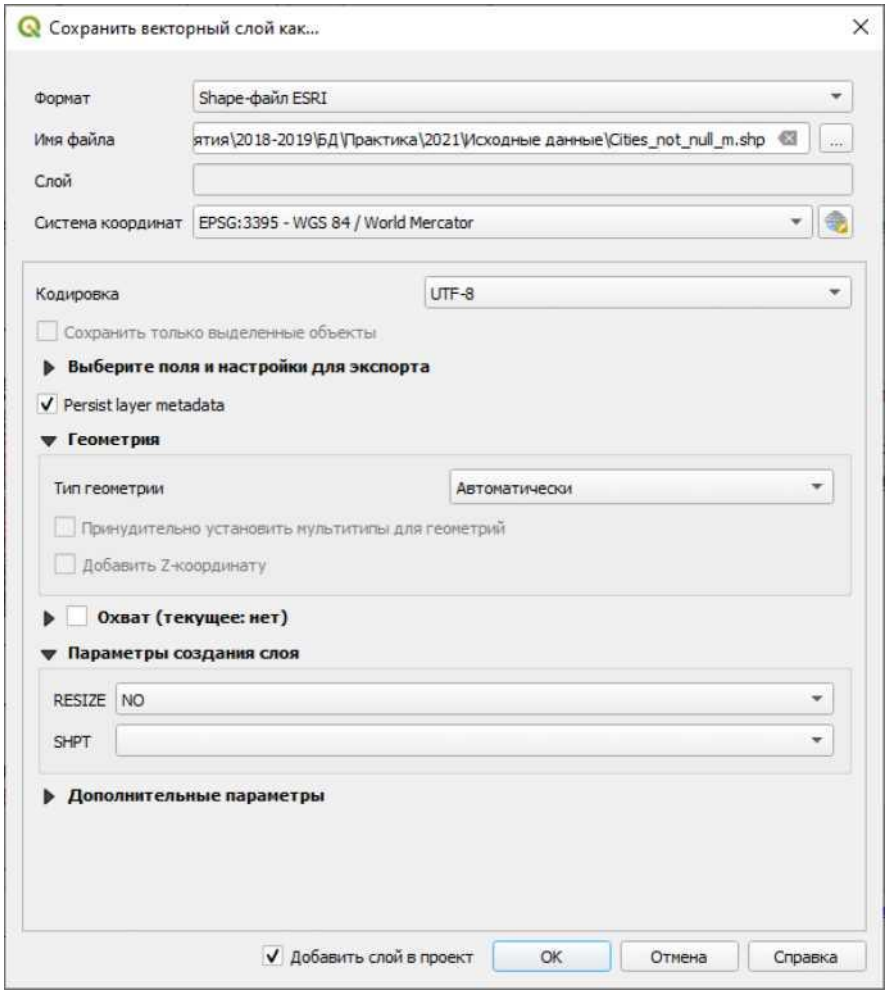


Рис. 5.4. Настройки изменения проекции слоя

Первым шагом будет поиск нужного города по названию. Для этого нужно выполнить команду *Выбрать объекты по выражению* или при открытой таблице атрибутов слоя нажать на кнопку *Выделить объекты, удовлетворяющие условию* (рис. 6.5).

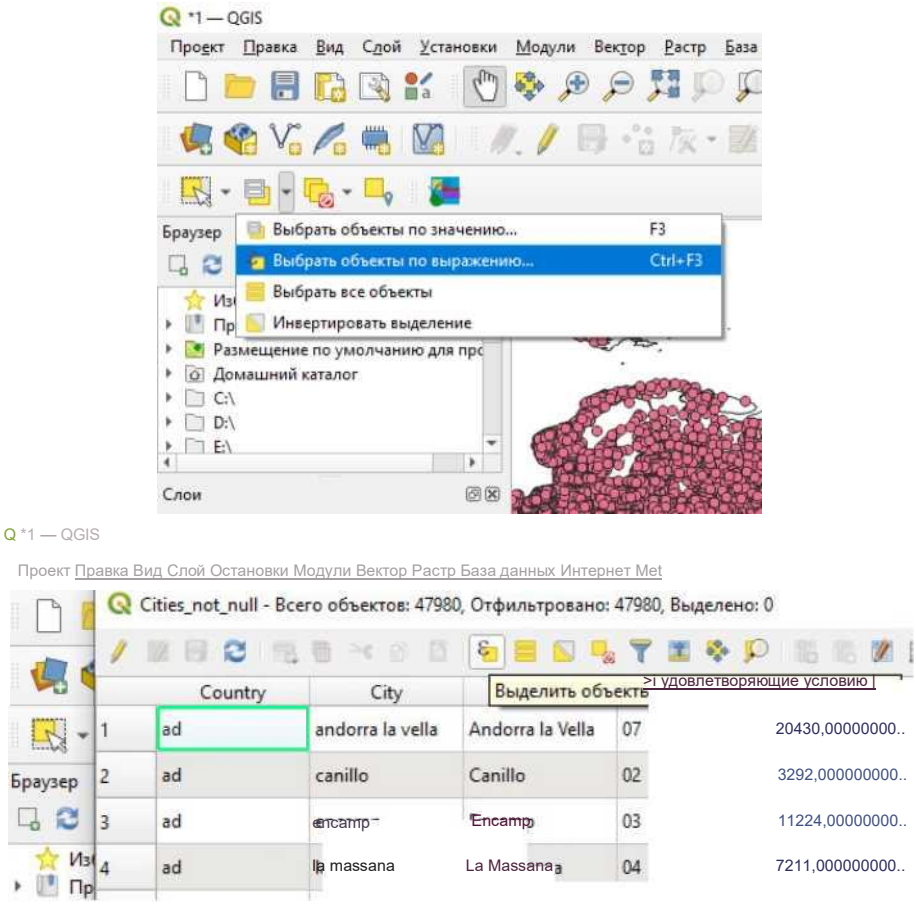


Рис. 5.5. Инструменты выбора объектов на основе значений атрибутов

Городов, имеющих такое название, два, поэтому, кроме указания названия, также нужно добавить условие принадлежности к определенному региону (рис. 6.6).

После того как визуально проверена корректность результатов поиска (для этого можно воспользоваться кнопкой *Увеличить до результатов поиска* в окне составления запроса), можно переходить к следующему шагу.

Далее, используя результаты предыдущего запроса, построим буферную зону с радиусом в 50 км относительно выбранного объекта. Для этого используется инструмент *Буфер,* находящийся в меню *Вектор - Геообработка* (рис. 6.7).

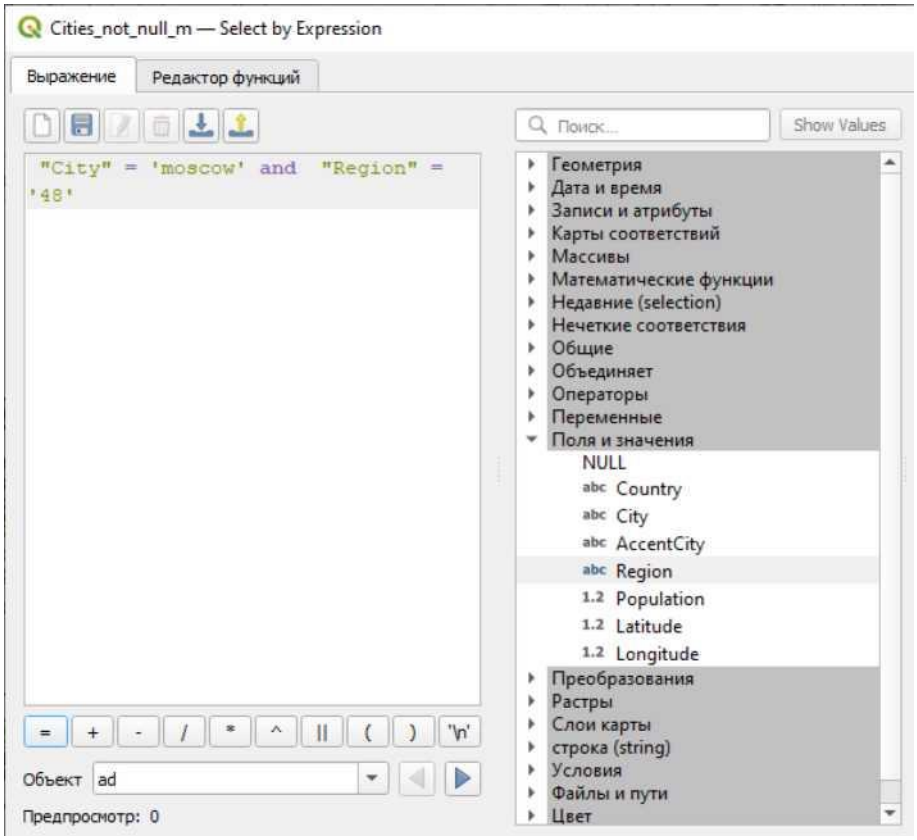


Рис. 6.6. SQL-запрос на выборку опорного объекта

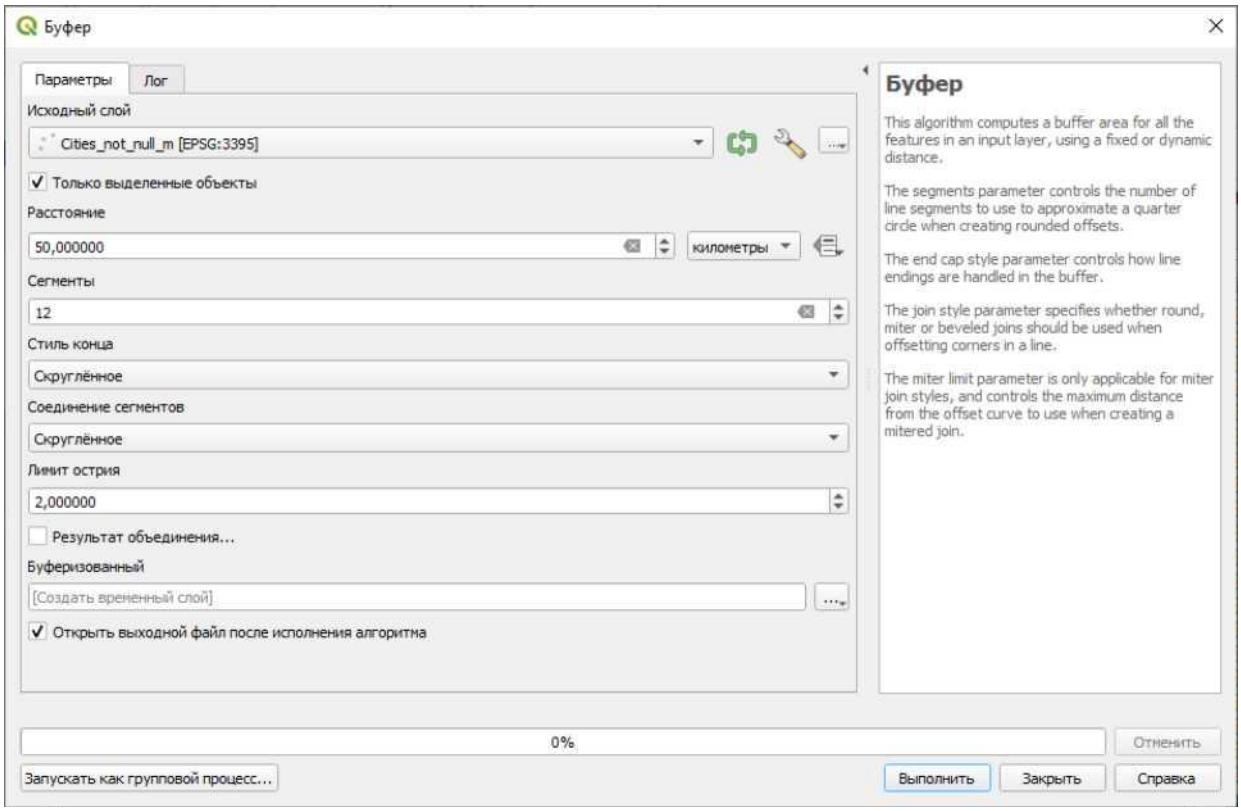


Рис. 6.7. Параметры инструмента построения буферной зоны

Теперь можно выполнить выбор городов, попавших в построенную буферную зону. Для этого нужно запустить инструмент *Выбрать по пространственному отношению,* находящийся в меню *Вектор - Выбор,* и

использовать в настройках тип пространственного отношения - *в пределах* (*are within*) (рис. 6.8). В результате будут выделены города, находящиеся внутри 50-километровой буферной зоны (рис. 6.9).

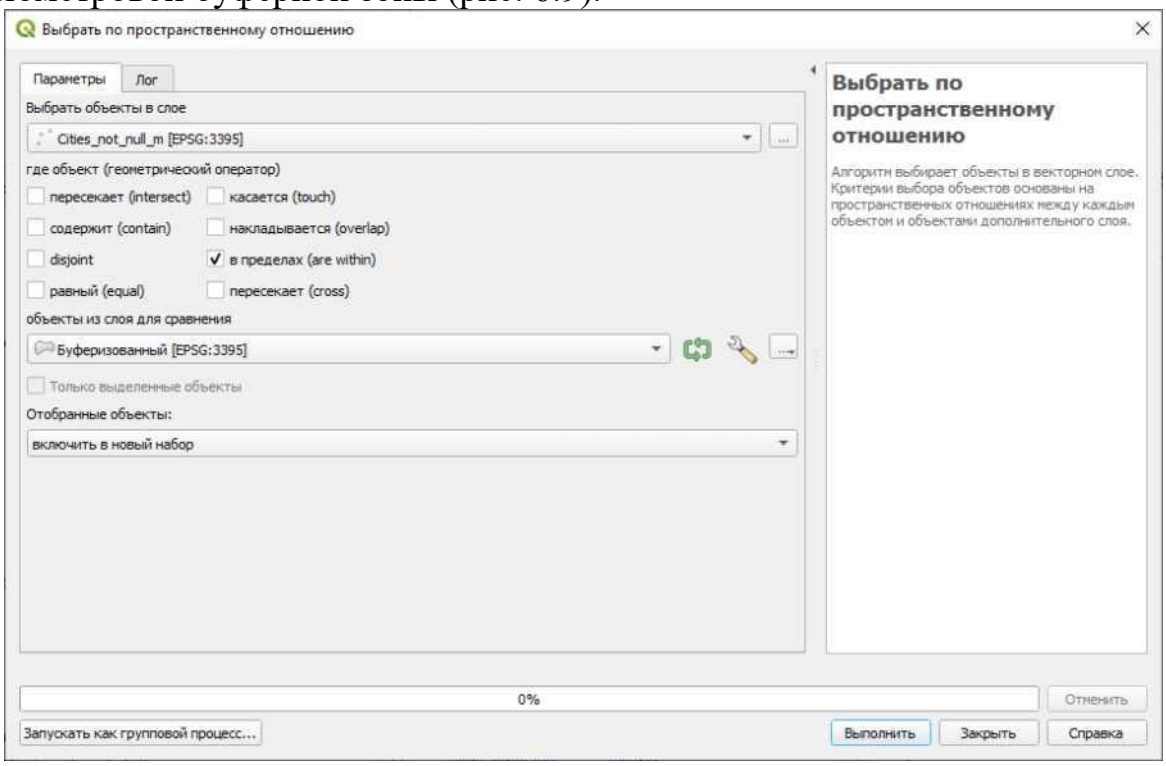


Рис. 6.8. Параметры выбора объектов

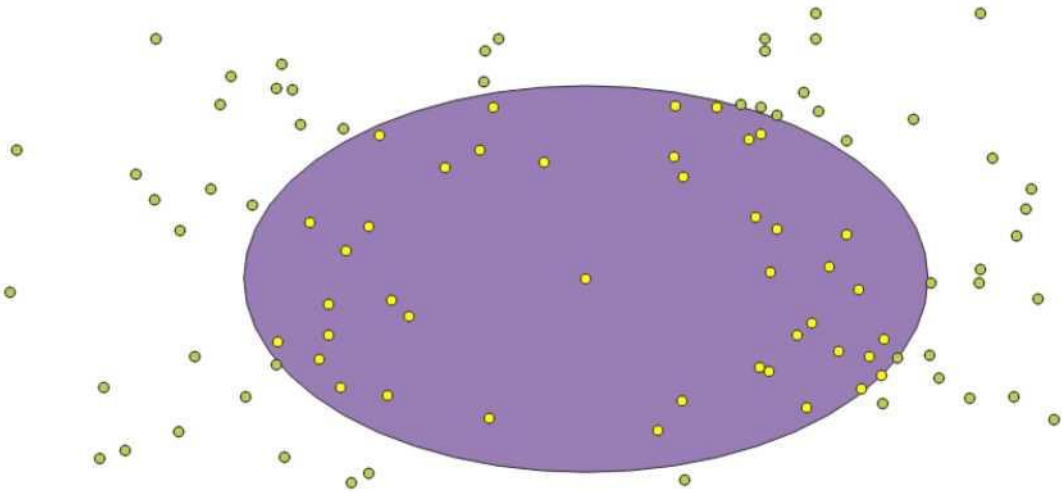


Рис. 5.9. Результаты запроса

Альтернативным вариантом решить эту задачу поиска объектов на указанном расстоянии можно с помощью команды *ST_Distance* (выпадающий список *Функции*), которая используется совместно с вложенным запросом (рис. 6.10).

Текст SQL-запроса:

SELECT 'Cities_not_null_m'.'City',

ST_Distance((SELECT 'Cities_not_null_m'.'geometry'

FROM 'Cities_not_null_m'

WHERE 'Cities_not_null_m'.'City' = 'Moscow' and 'Cities_not_null_m'.'Region' = '48'), 'Cities_not_null_m'.'geometry', 50000) AS 'Distance' FROM 'Cities_not_null_m'

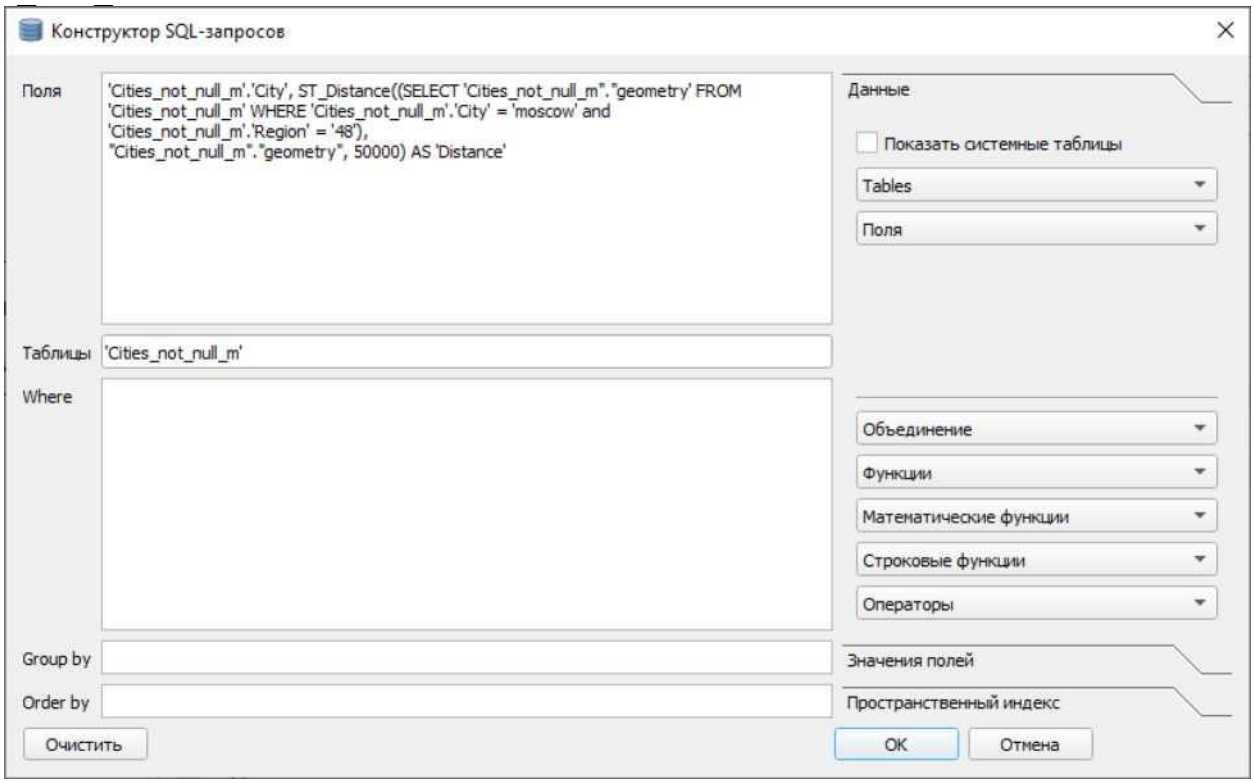


Рис. 6.10. Использование функции *STDistance* в SQL-запросе

Построение картограммы

Осуществим построение картограммы на основе данных о регионах, к которым принадлежат страны.

Поскольку в исходной таблице с контурами стран представлена не вся необходимая информация, то ее необходимо связать с информацией из таблицы *Countries_of_the_world*. Ключевыми полями для связи будут названия стран. Для того чтобы настроить параметры связи между таблицами, нужно открыть *Свойства основного слоя (Countries)* и перейти в раздел *Связи*. В этом разделе нужно нажать на кнопку *Добавить новую связь* и указать присоединяемую таблицу (*Countries_of_the_world*) и ключевые поля *Country* и *NAME* (рис. 6.11).

Теперь у стран мира добавилось поле с названиями регионов мира, и по этому полю можно построить тематическую карту регионов.

Для настройки тематической карты нужно перейти на вкладку *Оформление* в свойствах слоя. Выбрать *Символизацию по уникальным значениям* и указать, что значения будут использованы из присоединенного поля *Countries_of_the_world_Region* (рис. 6.12).

Результат создания тематической карты по регионам мира представлен на рис. 6.13 ниже.

Q Layer Properties — Countries — Связи

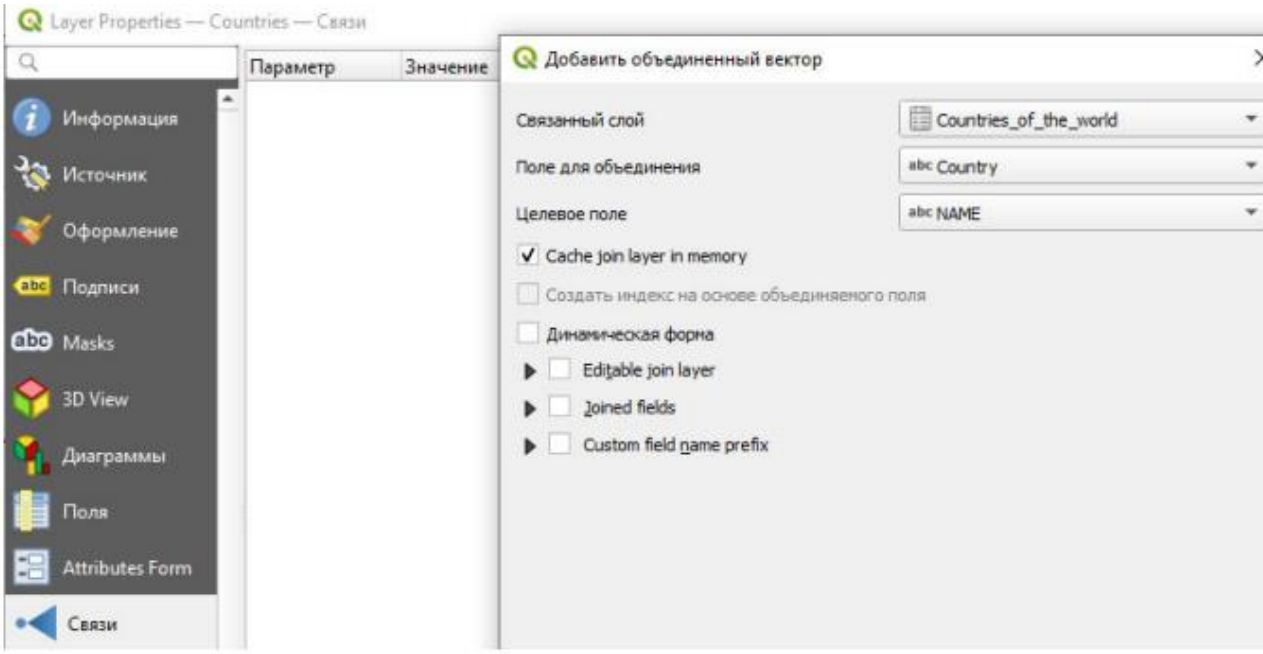


Рис. 6.11. Меню настройки связей таблиц

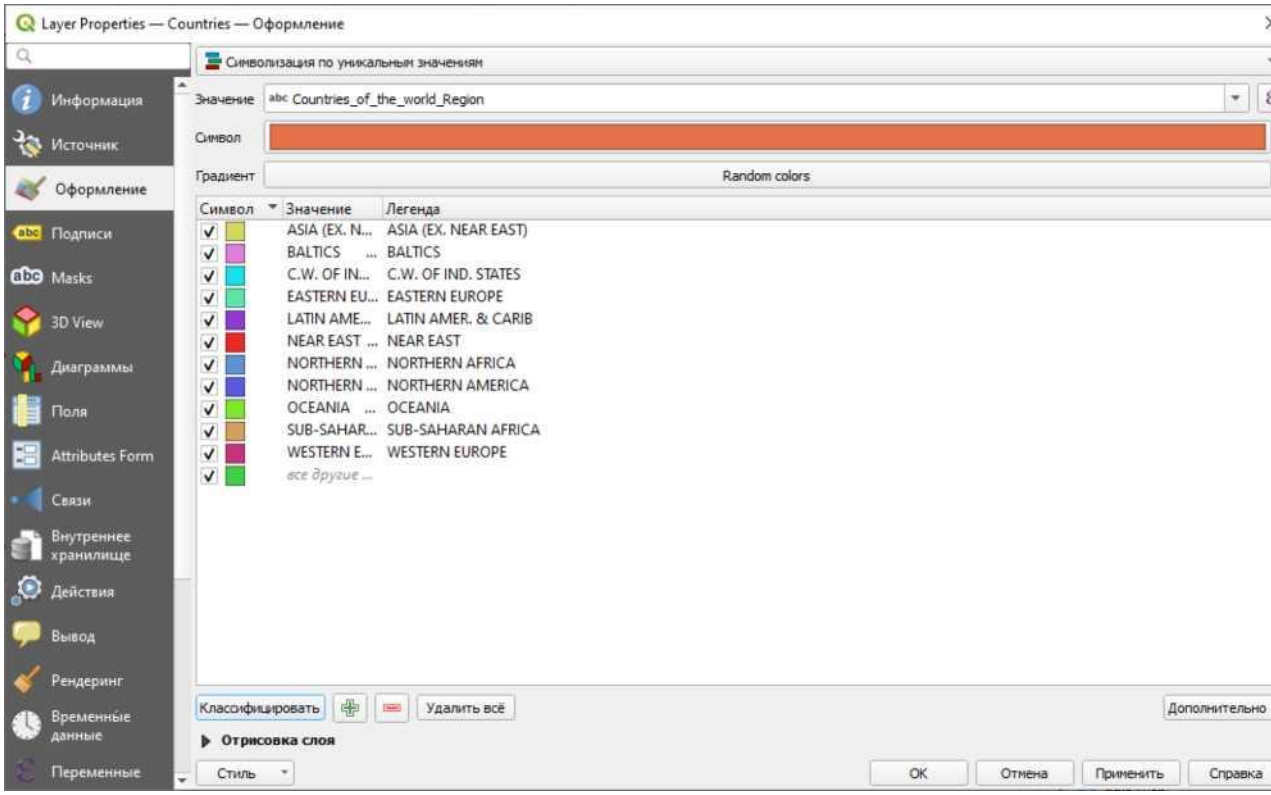


Рис. 6.12. Настройки внешнего вида тематической карты

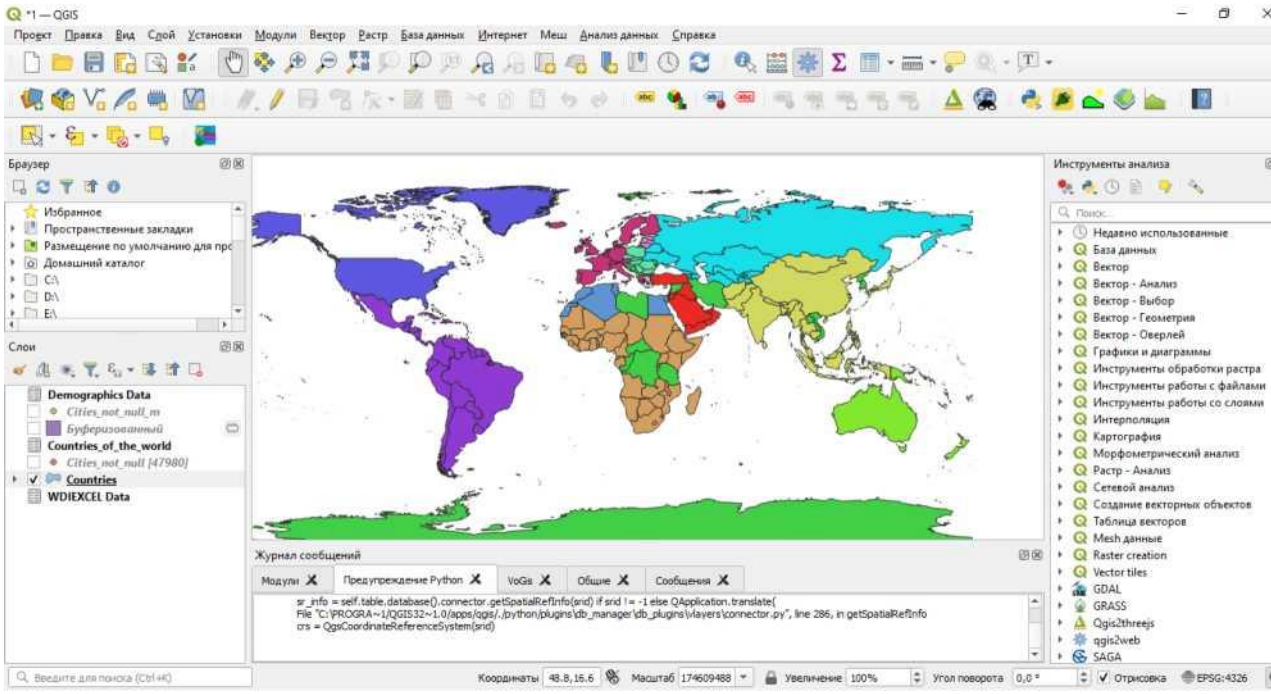


Рис. 6.13. Результат построения тематической карты

Построение картодиаграммы

Для примера построения картодиаграмм используются данные за 5 лет о численности населения по странам мира из таблицы *Demographics*.

Так же, как и в предыдущем примере, выполняется объединение таблиц с данными, только в этом случае не по названию страны, а по кодам стран (*ISO3* и *Country_Code*). Данный вариант позволит минимизировать возможные ошибки при связывании, в отличие от предыдущего варианта, где вероятность различного написания названий стран гораздо выше (рис. 6.14).

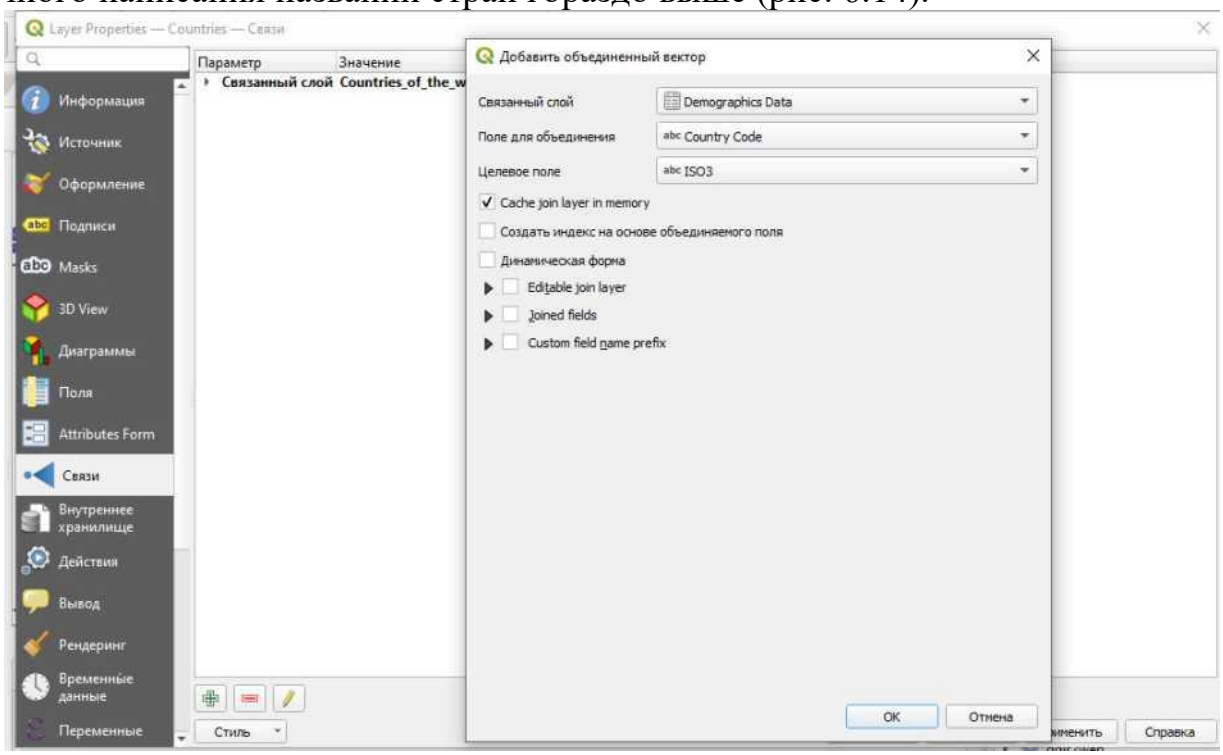


Рис. 6.14. Диалог связывания таблиц

Для построения столбчатой диаграммы (раздел *Диаграммы* в свойствах слоя) используются данные с 2009 по 2013 гг. В настройках выбирается тип диаграммы - *Гистограмма,* указываются 5 атрибутов с данными, на вкладке *Рендеринг* нужно обнулить отступы (*Bar Spacing*) между столбцами, на вкладке *Размер* подобрать оптимальное соотношение максимального значения и высоты столбца (рис. 6.15).

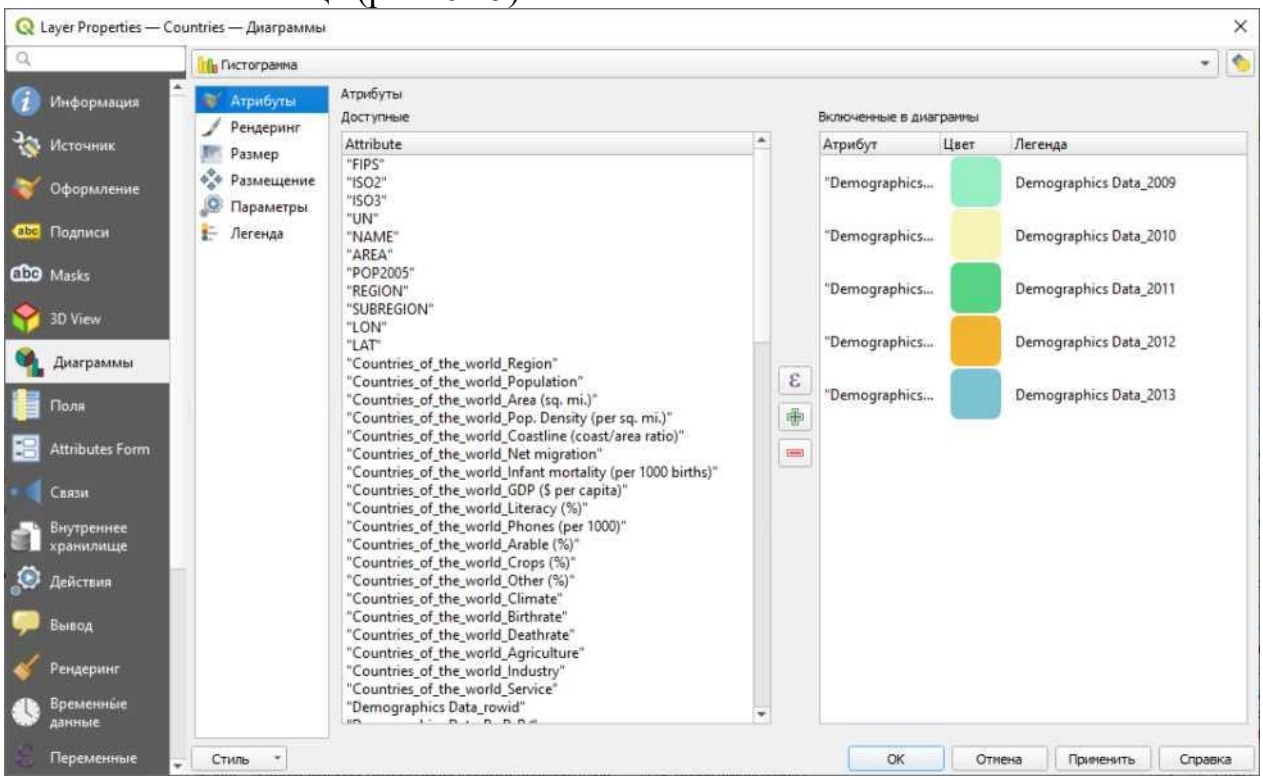


Рис. 6.15. Диалог настройки параметров диаграммы

Результат построения картодиаграмм представлен на рис. 6.16.

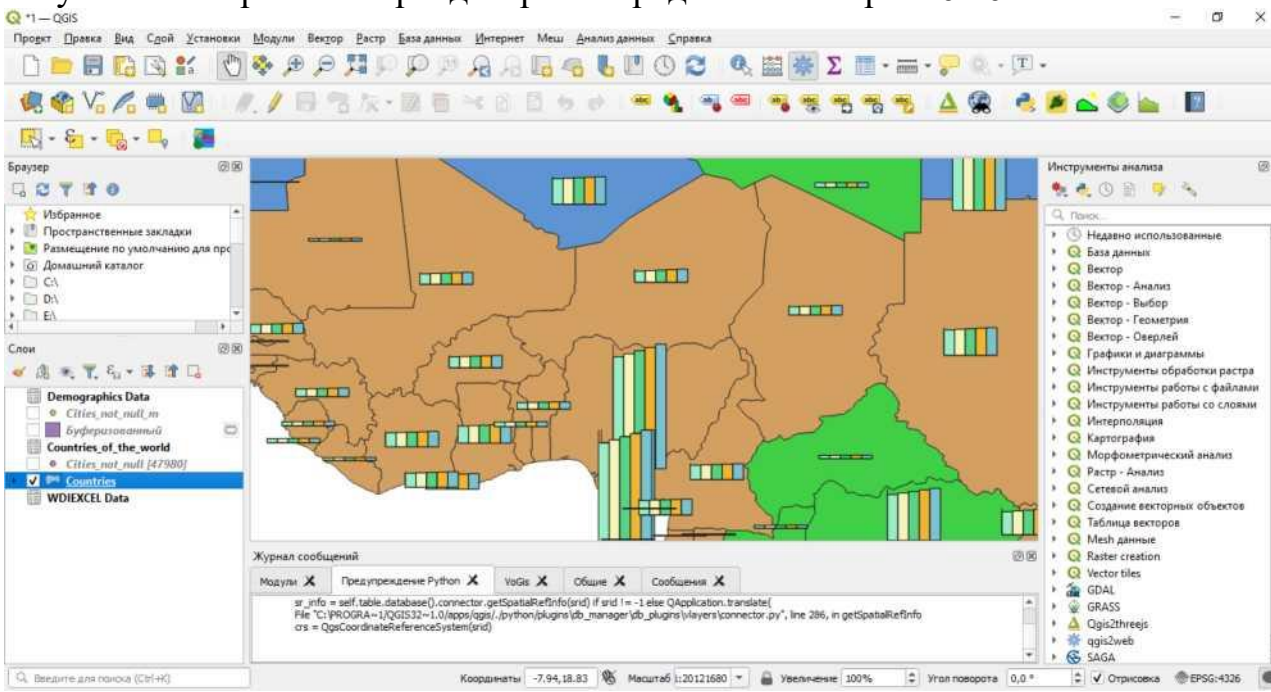


Рис. 6.16. Результат построения тематической карты

Для более наглядной визуализации можно включить подписи названий

стран. Для этого в разделе *Подписи* свойств слоя нужно выбрать пункт *Простая подпись* и указать атрибут с данными - *Name.* Параметры шрифта и размещения текста подбираются самостоятельно для обеспечения наилучшей читаемости. Результат построения тематической карты с картодиаграммами и подписями представлен на рис. 6.17.

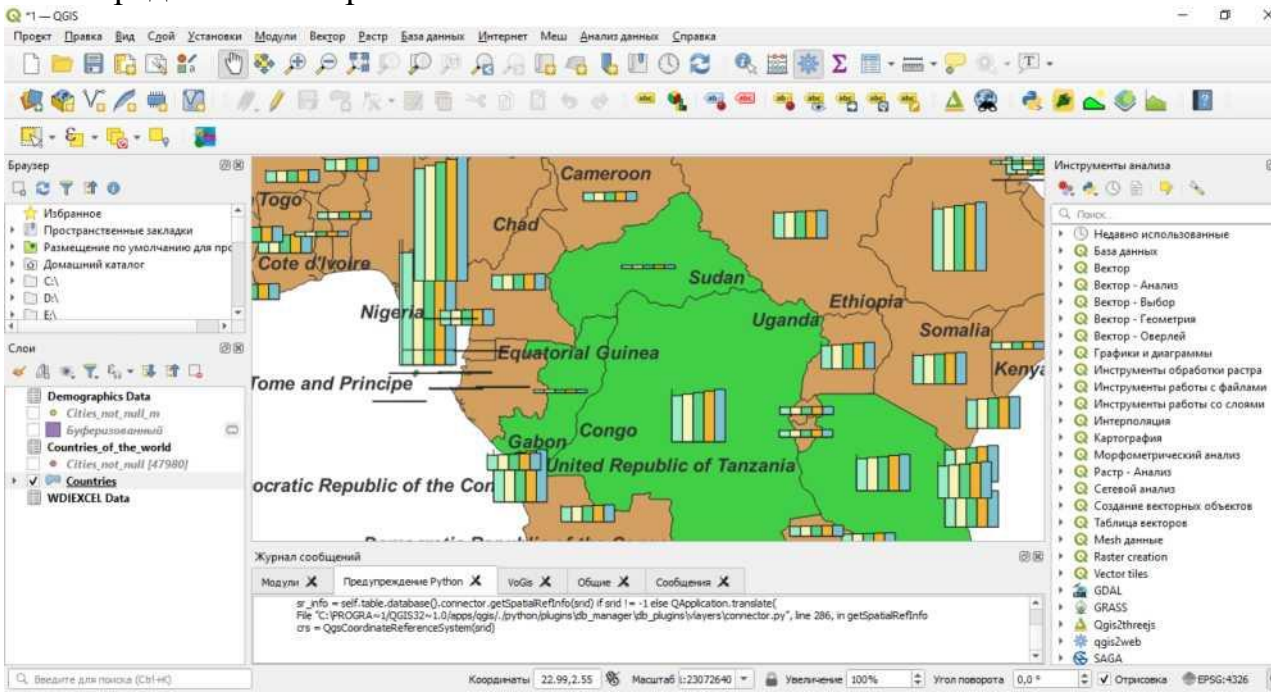


Рис. 6.17. Результат настройки внешнего вида слоев карты

Индивидуальные задания

Запрос № 1. Составить список стран региона с указанием количества населения за указанный год, упорядоченный по возрастанию. Регион и год взять на основе номера варианта (табл. 5.3). В отчет нужно вставить скриншот (один или более при необходимости) с составленным запросом и таблицу с результатами его работы.

Запрос № 2. Найти максимальную численность населения для определенной страны на указанном интервале лет (обе границы интервала включаются). В отчет нужно вставить скриншот (один или более при необходимости) с составленным запросом и итоговое значение. Варианты указаны в табл. 5.4.Варианты для запроса № 1

Номер варианта	Регион	Год
1	Asia (ex. Near East)	2002
2	Baltics	2010
3	C.W. Of Ind. States	2009
4	Eastern Europe	2008
5	Latin Amer. & Carib	2002
6	Near East	2005

7	Northern Africa	2004
8	Northern America	2003
9	Oceania	2008
10	Sub-Saharan Africa	2006
11	Western Europe	2007
12	Asia (ex. Near East)	2007
13	Baltics	2010
14	C.W. of Ind. States	2012
15	Eastern Europe	2011

Варианты для запроса № 2

Номер варианта	Страна	Интервал лет
1	Albania	2002-2004
2	Algeria	2008-2010
3	Argentina	2007-2009
4	Armenia	2006-2008
5	Austria	2002-2004
6	Bahrain	2005-2007
7	Belgium	2004-2006
8	Bolivia	2003-2005
9	Brazil	2008-2010
10	Bulgaria	2006-2008
11	Canada	2007-2009
12	Colombia	2007-2009
13	Croatia	2010-2012
14	Denmark	2010-2012
15	Estonia	2009-2011

Практическая работа 7. Анализ топологических отношений и пространственных данных

Цель задания — знакомство с редактированием векторных пространственных данных, элементами базовых технологий ГИС (оверлей, пространственный запрос, атрибутивный запрос).

Необходимая теоретическая подготовка: Системы координат и проекции карт, привязка геоизображений, трансформирование геоизображений. Пространственные запросы, атрибутивные запросы, оверлей.

Необходимая практическая подготовка: Знание основных компонент интерфейса QGIS (менеджер источников данных, панель слоёв, фрейм карты). Добавление источников пространственных данных в проект. Настройка символики и подписей объектов. Привязка растра. Создание макета, добавление карты и зарамочного оформления, экспорт макета.

Исходные данные: Схема районов города с сайта НЕАГЕНТ по состоянию на 2021 г.; набор пространственных данных о зданиях и сооружениях, созданный на основе базы OpenStreetMap

Результат: Набор пространственных данных с вернакулярными районами г. Новосибирск и статистикой по застройке в пределах районов. Картодиаграммы по количеству домов и степени застроенности. Картографическое изображение.

- Привязать растровую схему районов города
- Создать набор пространственных данных о районах путем цифрования растровой карты
- Добавить семантическую информацию о районах
- Выполнить серию пространственных и атрибутивных запросов для определения структуры застройки в пределах районов
- Построить картодиаграммы по полученным значениям
- Подготовить проект карты с компоновкой

Подготовка исходных данных

1. Загрузите архив с исходными данными и распакуйте его в вашу рабочую директорию.
2. Создайте новый проект QGIS и загрузите любую картографическую основу с веб-сервиса. Сохраните проект в ту же директорию, где находятся исходные данные.
3. Привяжите схему районов. Если вы не знаете, как привязать схему, сверьтесь с упражнением 5

Создание набора пространственных данных

На этом этапе мы создадим новый векторный набор пространственных данных, в котором будут храниться контура районов. QGIS поддерживает множество форматов пространственных данных, выбор конкретного формата в каждом случае определяется задачами, требованиями и ограничениями проекта. В качестве формата по умолчанию в QGIS 3 используется GeoPackage.

Файл GeoPackage представляет собой базу данных SQLite с дополнительными функциями для хранения наборов пространственных данных. Вся необходимая информация хранится в единственном файле *.gpkg; в процессе работы создаются временные файлы. В одном файле может быть сохранено несколько наборов пространственных данных (иногда их также называют «таблицами», *tables*, или «слоями», *layers*)

1. Запустите процедуру создания нового набора данных GeoPackage. Для этого нажмите кнопку на панели менеджера источников данных или сочетание клавиш Ctrl+Shift+N.

2. В открывшемся окне введите следующие параметры:

- **База данных:** сохраните файл *.gpkg в директорию вашего проекта. Назовите его по шаблону Districts_%Фамилия%.gpkg;
- **Имя таблицы:** примите значение, предлагаемое по умолчанию после указания пути к базе;
- **Тип геометрии:** полигон (площадная)
- Чекбоксы «Include Z dimension» и «Include M values» оставьте выключенными;
- **Система координат**: такая же, как система координат проекта.

3. Помимо геометрии объектов, нам нужно будет хранить семантическую информацию — номера районов. Создайте новое поле целочисленного типа с названием district_id, как показано на рисунке ниже.

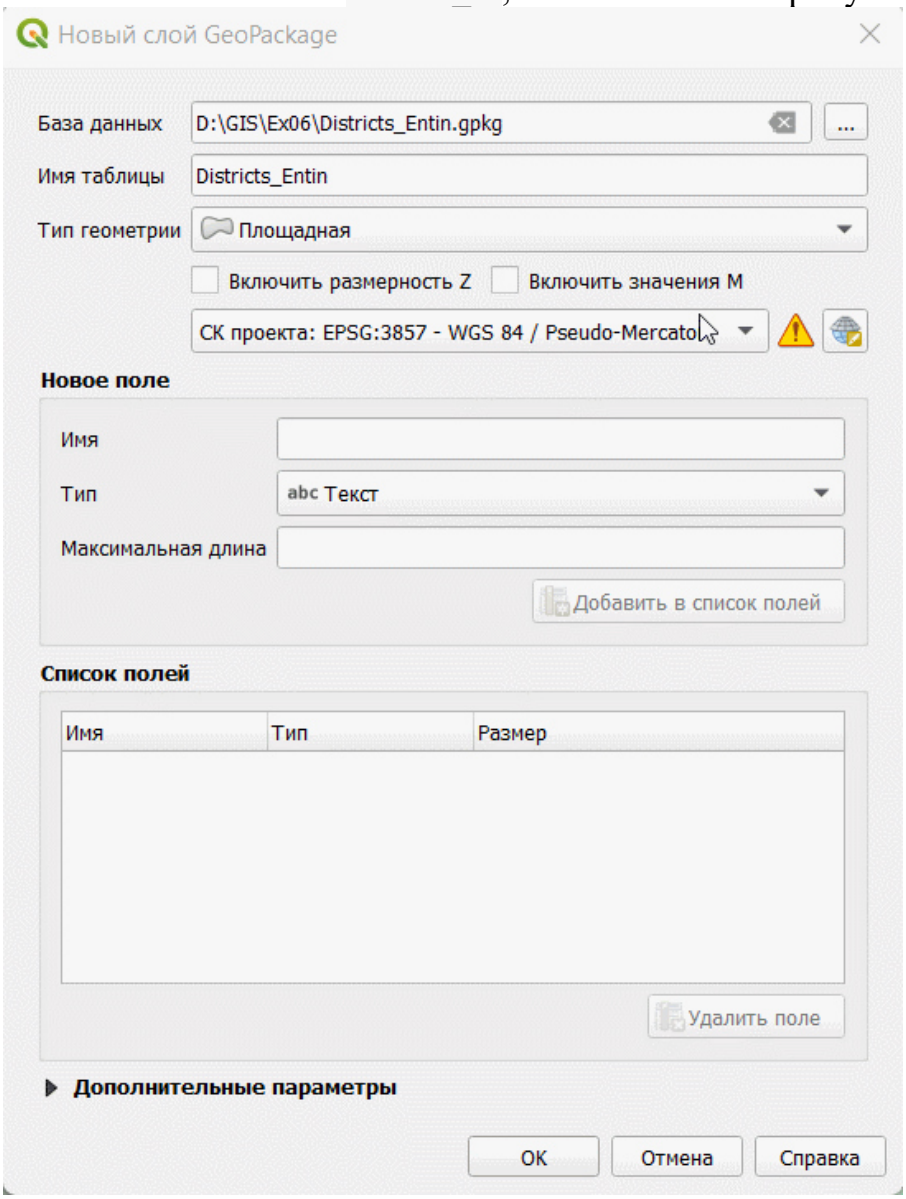


Рис.7.1. Создание слоя в QGIS

4. Проверьте правильность заполнения параметров. Если всё заполнено правильно, нажмите OK. Новый слой будет добавлен в проект.

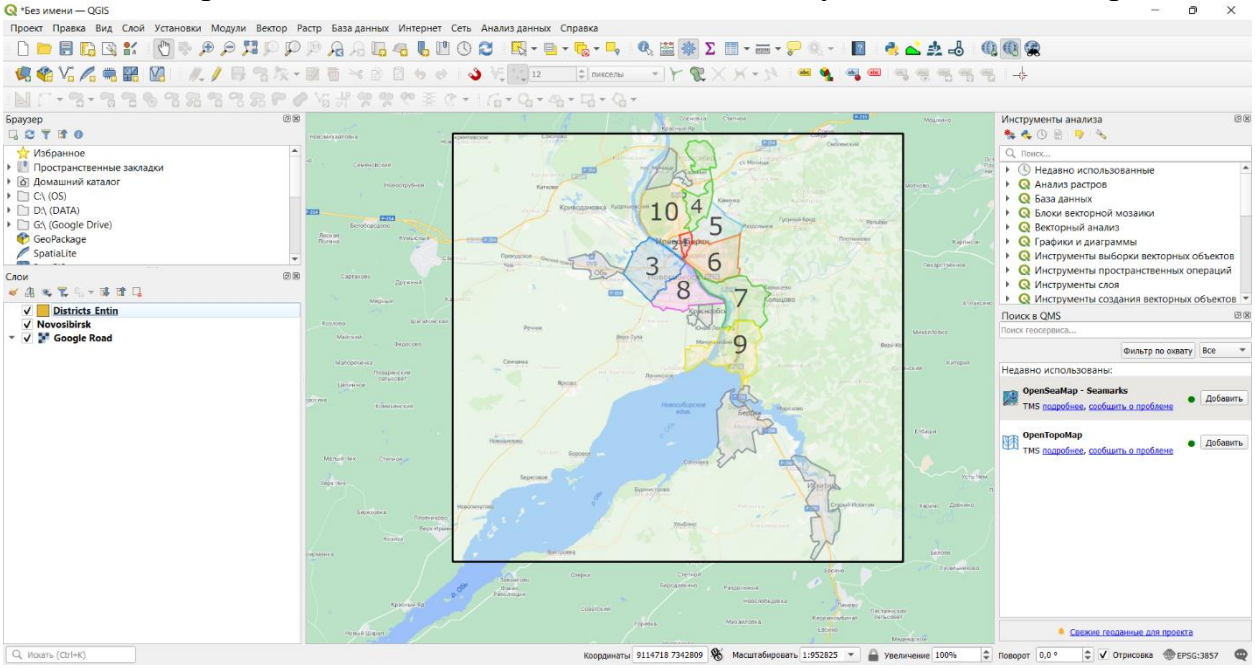


Рис.7.2 Созданный слой

Векторизация районов

1. Измените стиль слоя на pattern dot blue. Эта символика будет удобна для цифрования, поскольку позволит не «скрывать» растровый слой под создаваемыми объектами и в то же время будет контрастно выделяться на их фоне.

2. Включите режим редактирования при помощи кнопки или из контекстного меню слоя. Запустите режим редактирования.

3. Когда режим редактирования запущен, нажмите кнопку , чтобы начать добавление нового объекта

4. Увеличьте изображение до района, обозначенного цифрой «1». Вы начнёте векторизацию с него. По нажатию левой кнопки мыши устанавливается положение первого узла создаваемого контура. «Обходите» контур по часовой стрелке или против часовой стрелки, устанавливая новые узлы нажатием левой кнопки мыши, как показано на изображении ниже

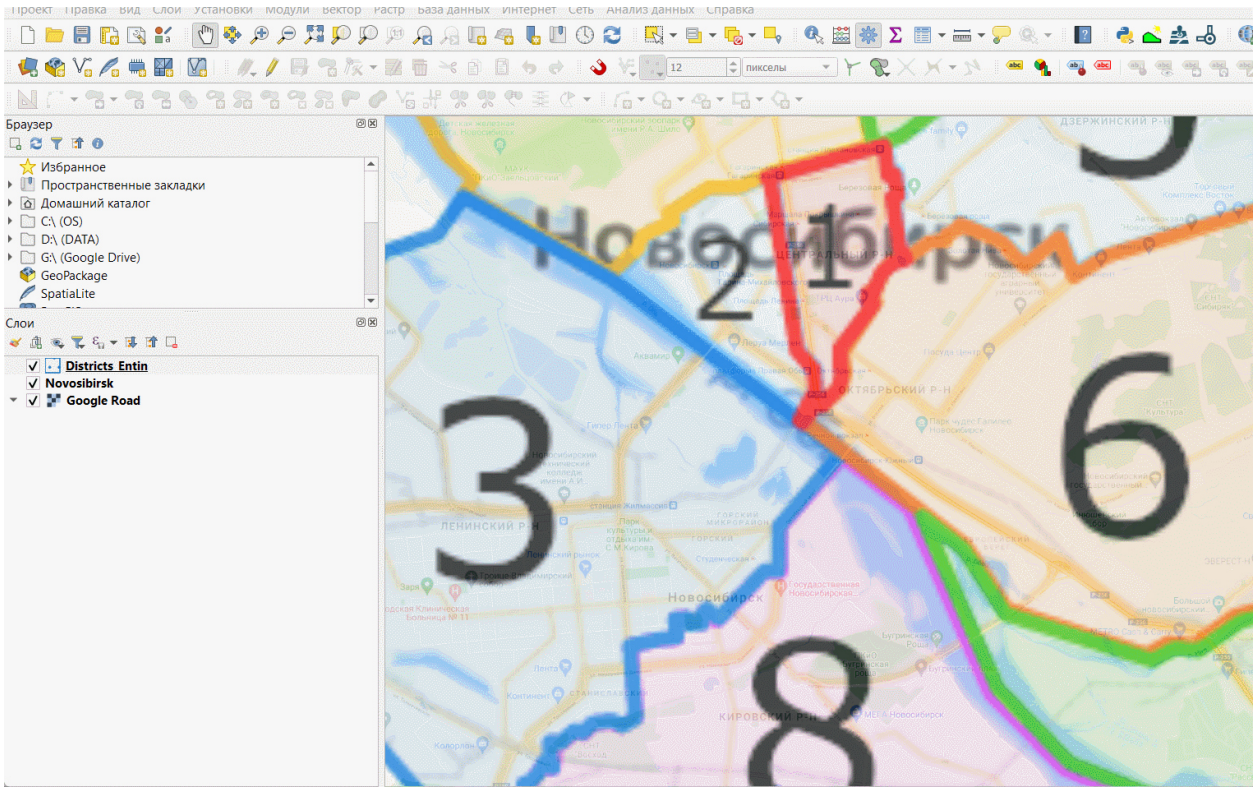


Рис.7.3. Векторизация слоя

5. Чтобы завершить редактирование объекта, нажмите правую кнопку мыши.

6. Когда вы завершаете создание объекта в QGIS, появляется окно ввода значений атрибутов. Введите номер района, как показано ниже

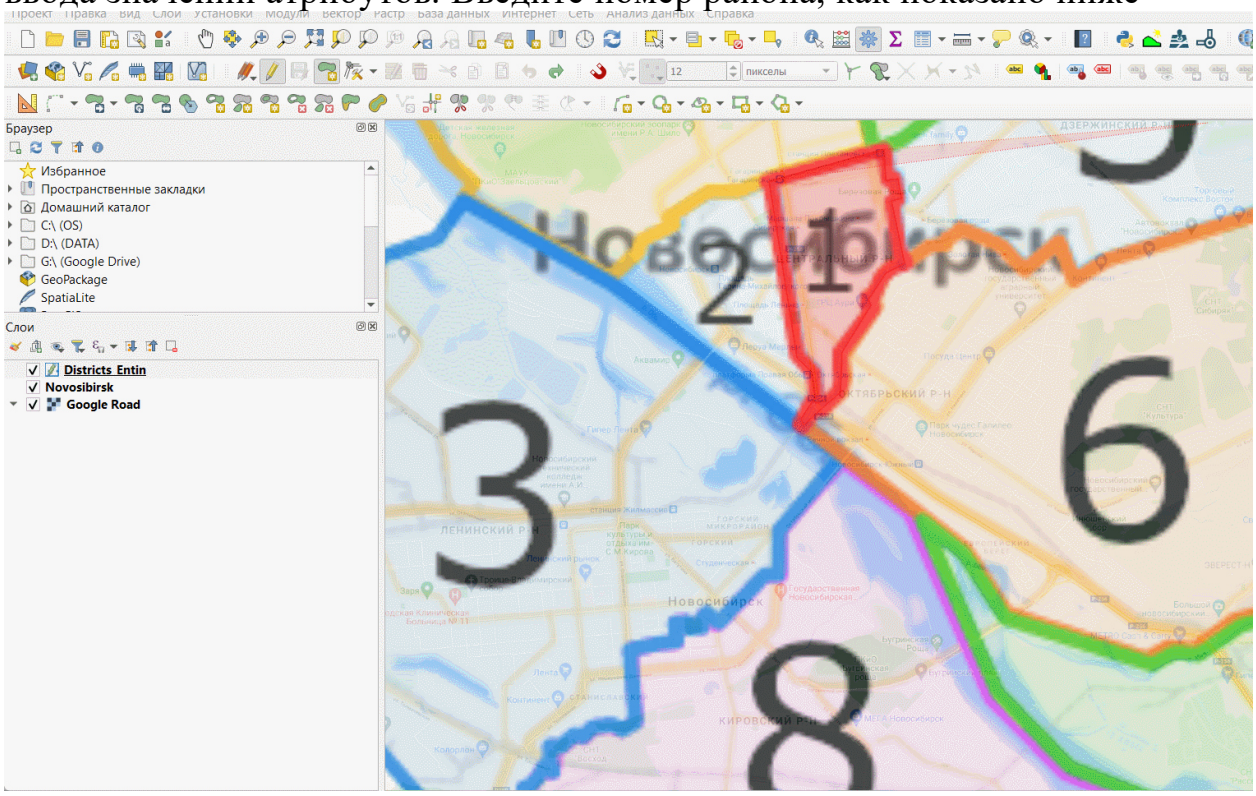
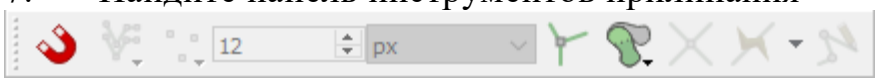


Рис.7.4. Заполнение атрибутов

Когда вы создаёте пространственные объекты в ГИС, важно соблюдать топологию объектов. Координаты узлов соседних объектов должны

совпадать, между объектами не должно быть пробелов и перекрытий. Чтобы добиться такого соответствия, вам необходимо активировать опцию прилипания (снеппинга).

7. Найдите панель инструментов прилипания



8. Включите режим прилипания, нажав на иконку с изображением магнита

9. Нажимая на соседние иконки в панели, включите прилипание к объектам во всех слоях, но только к вершинам объектов

Теперь, если вы будете подводить курсор к существующим объектам, его положение будет автоматически «притягиваться» к ближайшим узлам.

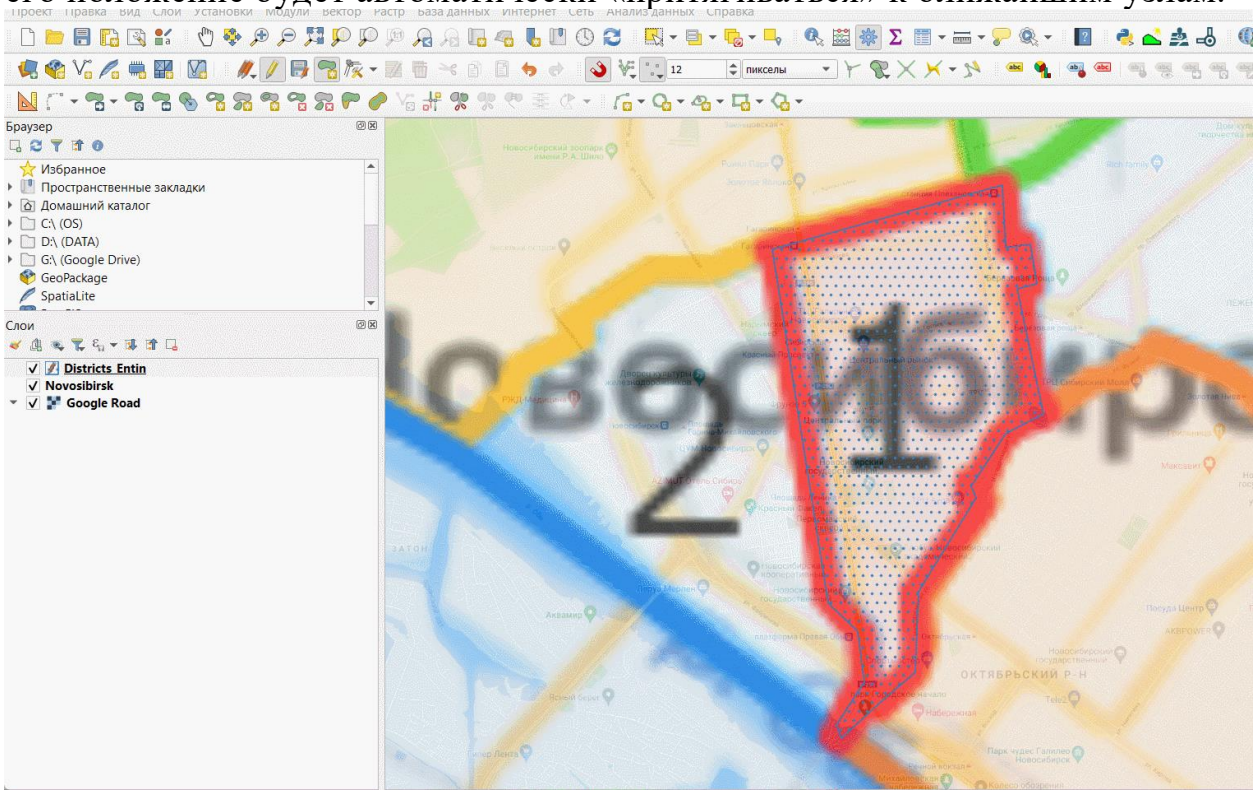


Рис.7.5. Инструменты прилипания

10. Прилипание позволяет реализовать создание новых объектов путём трассировки по существующим объектам. Включите режим трассировки, нажав на кнопку

11. Оцифруйте соседний объект с помощью трассировки. Для этого подведите курсор к углу существующего объекта и установите первый узел. Затем переместите курсор вдоль общей границы и установите следующий узел. Промежуточные узлы будут добавлены автоматически. После оцифровки общей границы продолжайте векторизацию обычным способом. После добавления нового объекта не забудьте внести его атрибуты во всплывающем окне.

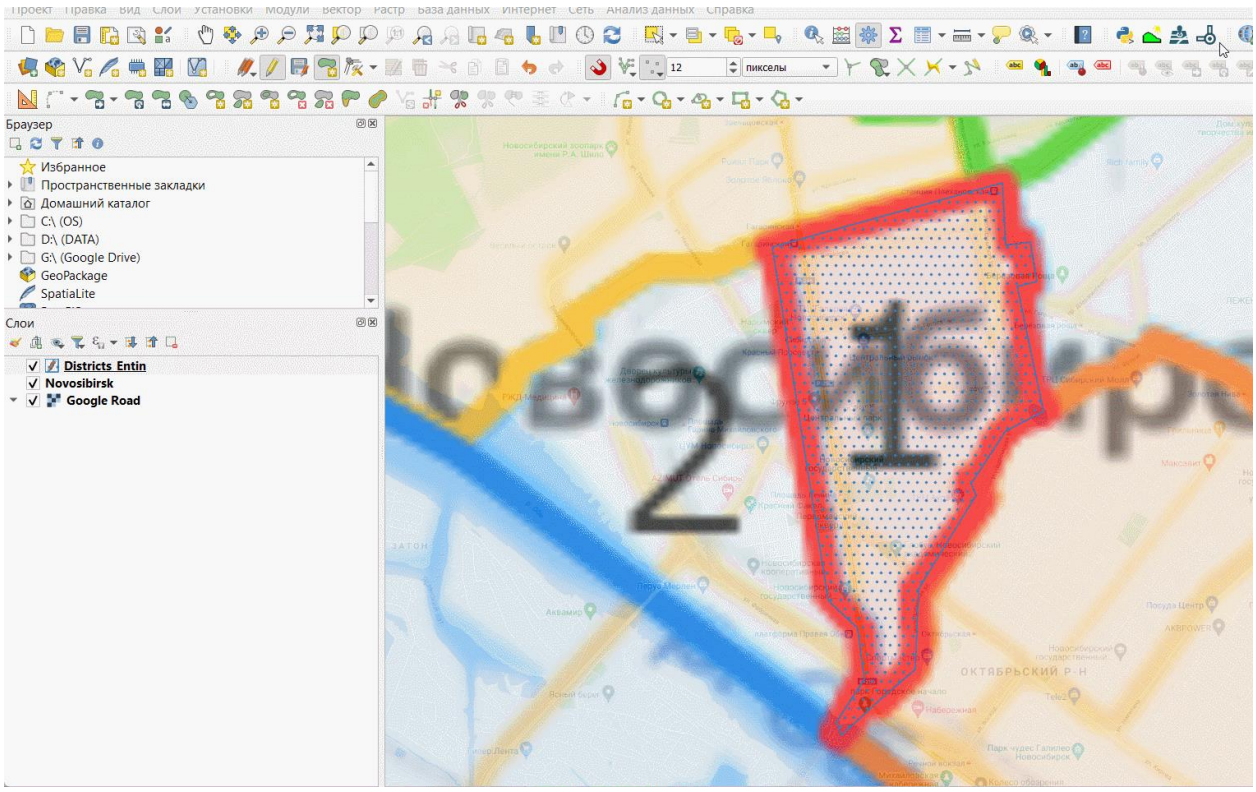


Рис.7.6. добавления нового объекта

14. Используя полученные навыки, оцифруйте все районы города.

Важно! Время от времени сохраняйте ваши правки при помощи кнопки на панели редактирования. В QGIS, как и во всех ГИС-пакетах, сохранение правок в данных выполняется отдельно от сохранения проекта.

Несколько советов по редактированию данных:

Чтобы отменить последний установленный узел, нажмите Backspace;

Чтобы полностью удалить последнюю оцифрованную фигуру, нажмите Ctrl+Z.

Чтобы изменить положение отдельного узла уже созданного объекта, можно воспользоваться инструментом редактирования узлов .

Трассировка в QGIS работает только тогда, когда весь путь трассировки отображается на экране. Если часть сегмента вылезла за пределы фрейма, трассировка не сработает.

Изучите панель «Дополнительные инструменты оцифровки» — она может вам пригодиться

Плавные кривые линии (берега, реки и т.п.) при цифровании обычно аппроксимируются ломаными линиями с относительно большим числом узлов — так, чтобы отличия не были визуально заметны. Хотя многие форматы данных позволяют создавать криволинейные объекты, этот подход не являются общепринятым.

18. По окончании цифрования объектов отключите растровый слой и отобразите данные в полном охвате. Сделайте снимок экрана.

Пространственные и атрибутивные запросы

В этой части работы вы оцените число жилых домов, оцифрованных пользователями ресурса OpenStreetMap, в пределах вычерченных вами районов. Волонтёрская географическая информация, к которой относится и OpenStreetMap, является важным источником данных для географических исследований благодаря своей детальности и относительной оперативности обновления, однако имеет ряд недостатков: неравномерное покрытие территории данными, низкая согласованность, структурированность и надёжность данных.

Для подсчёта числа зданий вы воспользуетесь следующим алгоритмом:

- Выбрать район на карте
- Выбрать здания, попадающие в его пределы (*пространственный запрос*).
- Из полученной выборки оставить только здания, принадлежащие к определённому типу (*атрибутивный запрос*).
- Записать число отобранных зданий в соответствующий атрибут текущего района.

Операции повторяются для каждого района.

1. Добавьте на карту набор данных о зданиях, который вы загрузили в начале занятия.
2. Откройте **таблицу атрибутов** слоя зданий. Для этого нажмите правой кнопкой мыши на слой в панели слоёв и выберите опцию «Открыть таблицу атрибутов» в контекстном меню.

Таблица атрибутов — это представление базы данных в виде таблицы. В большинстве случаев работа с семантической информацией в ГИС выполняется через таблицу атрибутов.

Определите, какие атрибуты есть у набора пространственных данных о зданиях. Установите, какие значения атрибутов могут соответствовать жилым домам.

2. Закройте таблицу атрибутов слоя зданий и откройте таблицу атрибутов слоя районов.
3. Убедитесь, что для слоя районов включён режим редактирования.
4. В окне таблицы атрибутов слоя районов нажмите кнопку или Ctrl+W, чтобы добавить новое поле, в которое будет записано число жилых домов для каждого района.
5. Добавьте поле с именем number_of_houses целочисленного типа.

На этом подготовительные операции закончены, мы переходим к запросам.

3. **Выберите** любой из районов: выделите строку в таблице атрибутов или воспользуйтесь инструментом выборки в окне карты .

Выделенный объект будет подсвечен жёлтым цветом на карте и синим цветом в таблице атрибутов.

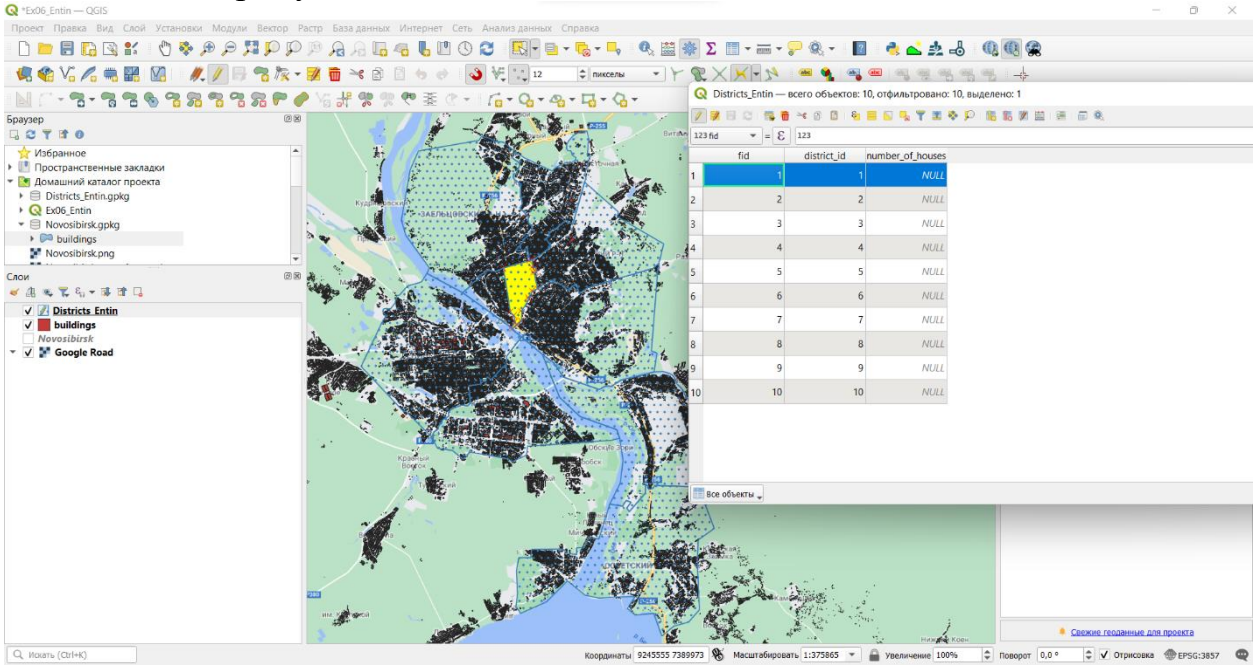


Рис.7.7. Таблица атрибутов

5. Теперь выберите те объекты из слоя зданий, которые находятся внутри выбранного района. Для этого запустите инструмент «Вектор» — «Выбор» — «Выбрать по расположению…». Используйте условие «находятся внутри» (*are within*). Включите опцию «Использовать только выделенные объекты» для выбирающего слоя.

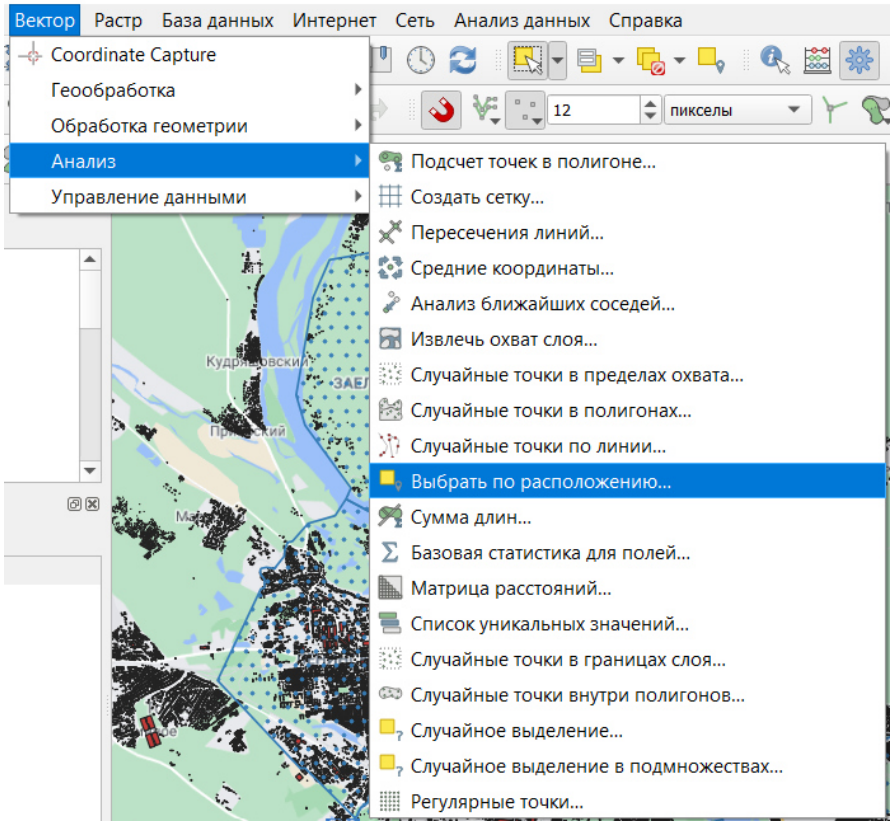


Рис.7.8. Анализ векторных данных по расположению

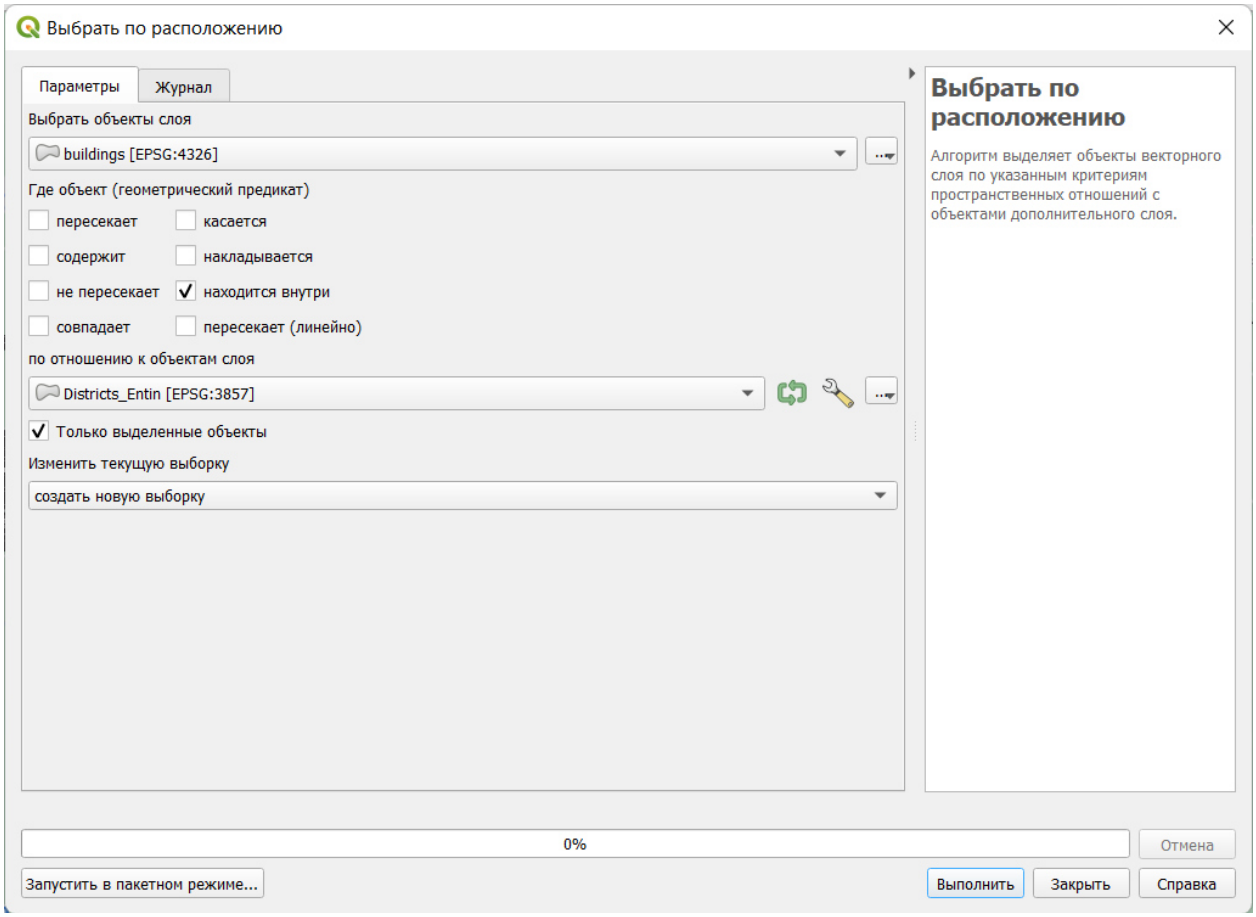


Рис.7.9. Свойства атрибутов по расположению

Нажмите «Выполнить», чтобы выбрать объекты в слое зданий.

6. Имея выборку в слое зданий и не закрывая окно пространственного запроса («Выбрать по расположению…»), осуществите выборку по атрибутам. Для этого выберите в таблице слоёв слой зданий и нажмите кнопку «Выделить объекты, удовлетворяющие условию» (или нажмите Ctrl+F3. Откроется форма ввода атрибутивного запроса.

Атрибутивные запросы в ГИС, как правило, создаются с использованием диалектов языка SQL. Само выражение представляет собой только условие (*where clause*) и часто использует значения атрибутов. В QGIS можно выполнить запрос и без составления выражения на языке SQL, но поскольку SQL более функционален, мы призываем вас сразу начать пользоваться им.

Форма ввода атрибутивного запроса состоит из трёх частей. В левой части конструируется собственно запрос, средняя содержит список доступных переменных и функций, в правой отображается справочная и служебная информация. Сейчас вам нужно будет составить атрибутивный запрос, который позволит выбрать здания, являющиеся (с некоторой вероятностью) жилыми домами.

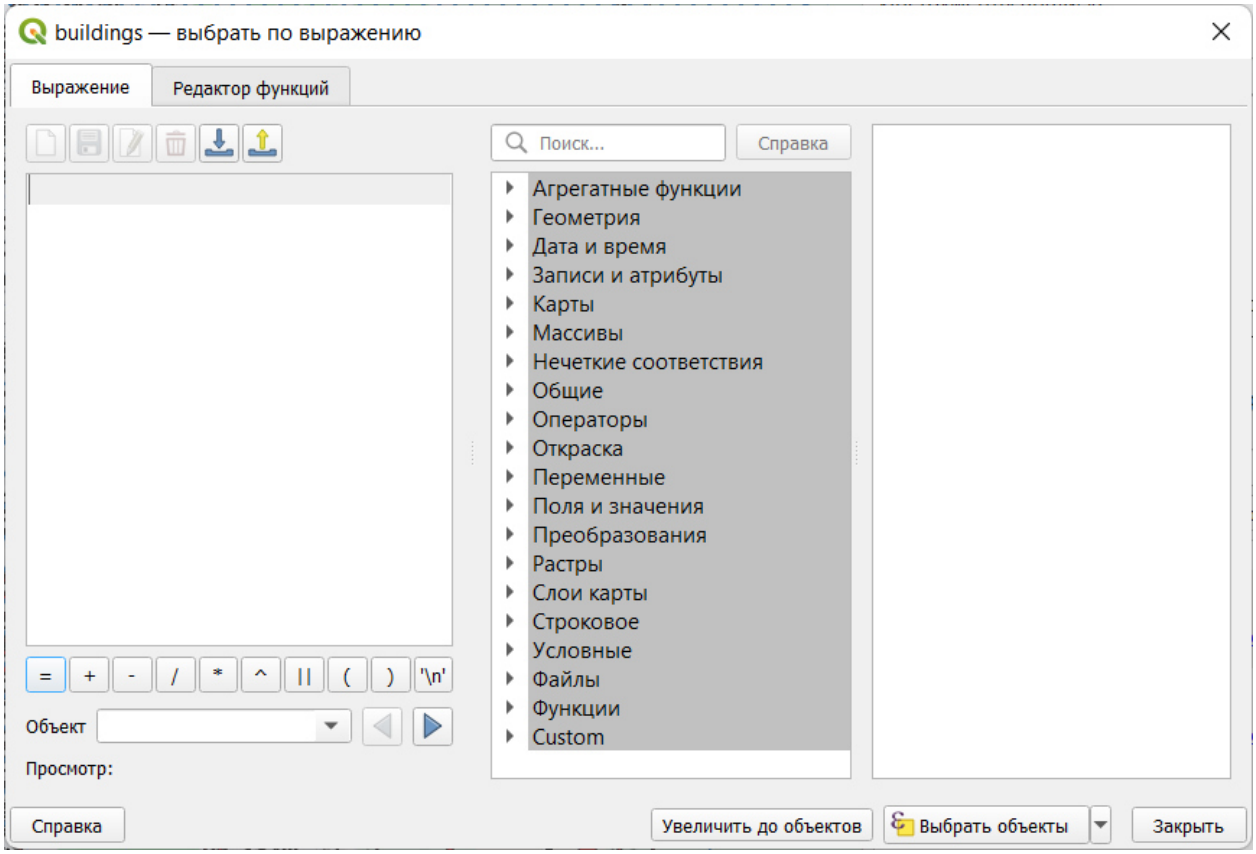


Рис.7.10. выборка в слое зданий

Атрибутивный запрос представляет собой логическое выражение, которое применяется к кортежу базы данных (строке таблицы). Результатом вычисления логического выражения является либо логическая 1 (ИСТИНА), либо логический 0 (ЛОЖЬ). В выборку попадают объекты, для которых при вычислении выражения получено значение 1.

В простейшем случае выражение состоит из проверки равенства. Например, выражение "type" = 'house' будет истинным для всех объектов, у которых в поле type записано значение house. Можно конструировать более сложные логические выражения с использованием операторов AND (И), OR (ИЛИ) и других. Например, выражение "type" = 'apartments' AND "name" IS NOT NULL будет истинным для объектов, у которых в поле type записано значение apartments (т.е. многоэтажные жилые дома) и одновременно в поле name записано какое-либо значение, отличное от пустого (NULL) (т.е. имеющие название). А запрос "type" = 'garage' OR "type" = 'garages' выберет из слоя зданий объекты, у которых в поле type записано либо значение garage, либо значение garages.

Примечание 1: в интерфейсе атрибутивного запроса и во всех остальных инструментах QGIS, использующих тот же интерфейс, названия полей обрамляются двойными кавычками (“), а строковые переменные — одинарными кавычками (’) Примечание 2: в QGIS и в большинстве систем управления базами данных (СУБД) для работы с пустыми значениями (NULL) используются отдельные операторы. Например, выражение "type" = NULL будет воспринято как некорректное, вместо него следует писать "type" IS NULL

7. Составьте выражение, реализующее следующее условие:

Выбрать объекты, у которых атрибуту type присвоено значение apartments или house

Вводить выражение можно, пользуясь списком функций и полей в правой части окна. Двойной клик по полю или функции добавляет соответствующую запись в окно выражения.

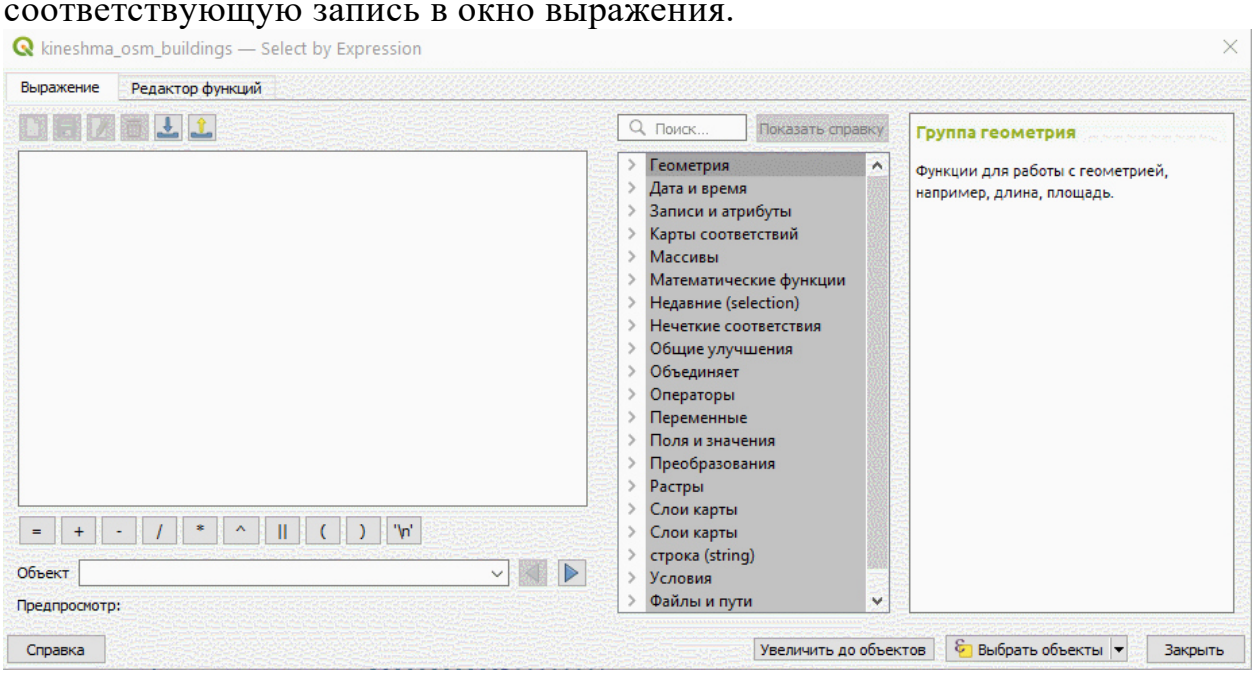
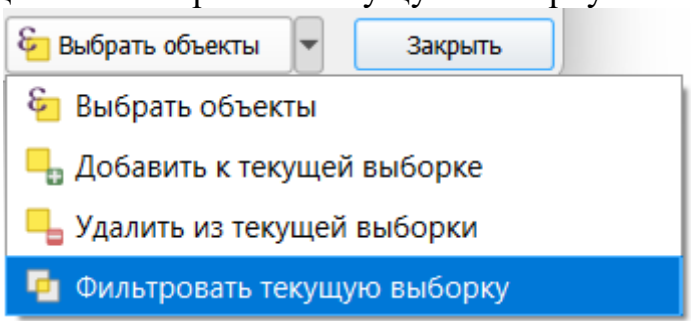


Рис.7.11. ввод выражения

После окончания ввода выражения и перед тем, как перейти к следующему шагу, скопируйте получившееся выражение в отчётный файл.

11. Мы сформулировали условие для выборки, однако нам нужно выбрать объекты не просто из слоя, а из уже существующей выборки. Для этого откройте выпадающий список кнопки «Выбрать объекты» и выберите функцию «Фильтровать текущую выборку»:



12. После применения фильтра появится всплывающее сообщение с числом выбранных объектов. Эта информация будет продублирована внизу окна QGIS:

139 объектов выделено в слое buildings.

13. Введите полученную цифру в таблицу атрибутов слоя районов в строке, соответствующей выбранному объекту (подсвечена синим).

14. Теперь, не закрывая окна пространственного и атрибутивного запроса, выбирайте по очереди каждый следующий объект в слое границ районов и повторяйте для них шаги 3-13. Таким образом вы заполните весь

столбец number_of_houses в таблице атрибутов. В процессе выполнения не забывайте периодически сохранять правки.

15. Скопируйте таблицу атрибутов в любой табличный процессор (Microsoft Excel, Google Sheets, LibreOffice Calc). Для этого при помощи сочетания клавиш Ctrl+A выделите все записи в таблице, скопируйте при помощи Ctrl+C и вставьте записи без форматирования в табличный процессор при помощи Ctrl+Shift+V.

16. Удалите столбец wkt_geometry, если он есть в вашей таблице. Скопируйте остальные столбцы и вставьте их в отчётный файл.

18. Завершите редактирование слоя районов.

19. Отключите отображение слоя зданий.

Визуализация слоя при помощи картодиаграмм

Как вы уже знаете из курса картографии, не существует прямого взаимно-однозначного соответствия между способами изображения, принятыми в картографии, и способами визуализации данных в ГИС. Чтобы отобразить абсолютную величину в пределах единиц статистического учёта (или аналогичных контуров), в картографии применяют способ картодиаграмм. Для создания условных знаков, соответствующих способу картодиаграмм, в QGIS применяется вкладка «Диаграммы» в свойствах слоя.

1. Откройте свойства слоя районов и перейдите на вкладку «Диаграммы». По умолчанию отображение диаграмм отключено. Включите отображение круговых диаграмм, выбрав опцию Pie Chart из выпадающего списка вверху.

2. В настройках диаграмм перейдите на вкладку «Атрибуты». Добавьте атрибут number_of_houses к диаграмме. Задайте для него мягкий оттенок оранжевого или жёлтого цвета.

Примечание: вкладка «Атрибуты» нужна для настройки секторов круговых диаграмм. Если вы добавляете несколько атрибутов к диаграмме, программа автоматически рассчитывает размеры секторов для каждого объекта. Сейчас перед нами не стоит такая задача, так что, фактически, можно было не добавлять атрибуты или добавить любой другой.

3. На вкладке «Отрисовка» («Рендеринг») установите для диаграмм настройку прозрачности (70 %)

4. На вкладке «Размер» измените способ задания размера с фиксированного на изменяющийся («масштабируемый»).

Изменение размера диаграмм в QGIS работает следующим образом. Пользователь задаёт максимальное значение показателя, который будет управлять размером диаграммы, и соответствующий ему максимальный диаметр диаграммы. Размер круга масштабируется пропорционально величине показателя. Из картографических соображений следует всегда выбирать масштабирование площади (в новых версиях QGIS некорректно переведено как «к поверхности»), а не линейного размера (диаметра).

5. В выпадающем списке «Атрибут» выберите нужный атрибут и нажмите кнопку «Найти, чтобы отобразить его максимальное значение. Оно будет отражено во вкладке «Макс. значение».

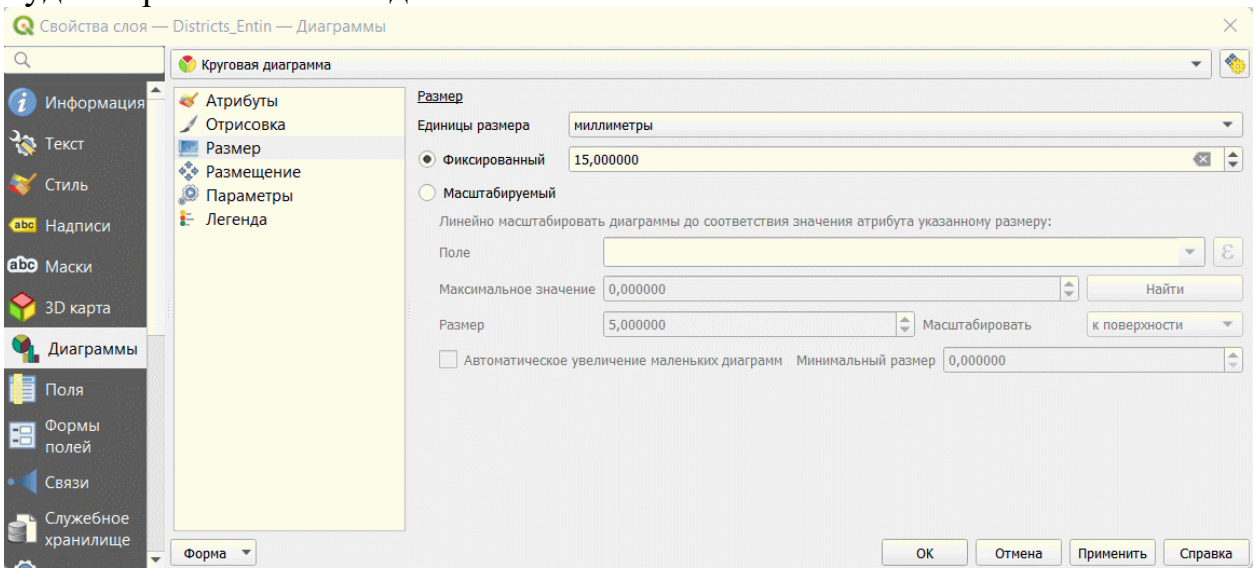


Рис.7.12. Свойства слоя диаграммы

6. Округлите полученное максимальное значение в большую сторону до величины, кратной 100.

7. Установите максимальный размер диаграмм равным 14 мм. Диаграмма такого диаметра будет соответствовать объекту с указанным максимальным значением атрибута.

В QGIS есть ещё одна полезная опция — увеличение размера диаграмм. Она применяется, если при масштабировании некоторые диаграммы становятся слишком малы. В таком случае их размер увеличивается до минимального (задаваемого пользователем) порогового значения. В этом упражнении она нам не требуется

8. Перейдите на вкладку «Размещение» и проверьте, что для ваших диаграмм указана опция размещения над центроидом.

9. Перейдите на вкладку «Легенда». Здесь вы настроите комбинацию условных знаков для отображения в легенде. Нажмите кнопку «Условные обозначения для размеров диаграмм» (*Show Legend Entries for Diagram Size*).

10. Настройте отображение значков следующим образом:

- «Коллапсируйте» значки легенды;
- В качестве символа используйте белый маркер круглой формы с тёмно-серой обводкой;
- Задайте заголовок («Число домов»);

Интерфейс настройки диаграммы должен принять приблизительно следующий вид:

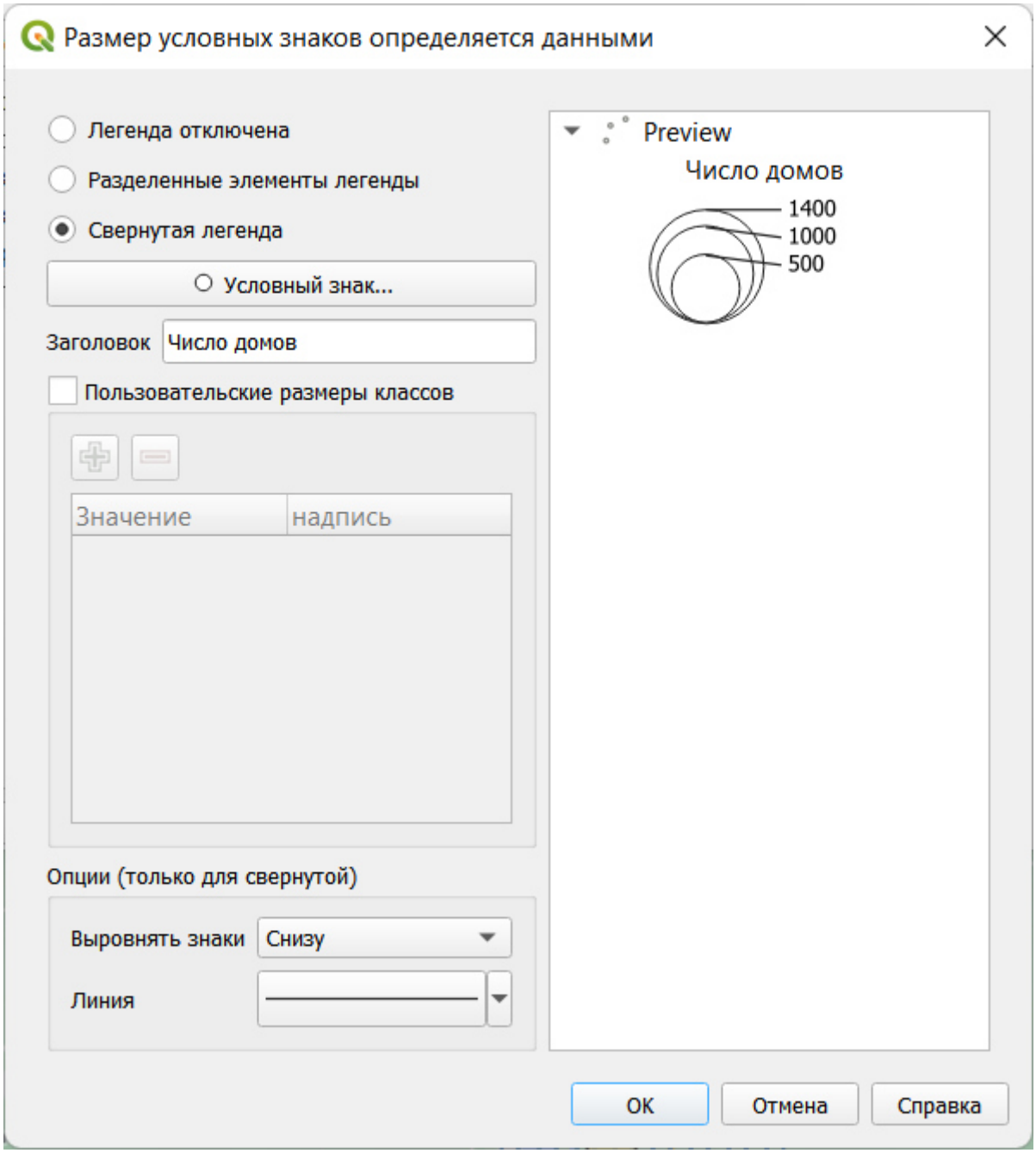


Рис.7.13. Размер условных знаков

Важное замечание: такой набор значков не является картографически корректным для легенды к абсолютной непрерывной шкале значков. По состоянию на февраль 2022 г. ни один широко используемый ГИС-пакет не может сделать легенду к размерам кругов картодиаграммы согласно принятым картографическим правилам. Легенды к таким картам следует составлять или исправлять вручную.

По окончании настройки символов окно QGIS будет выглядеть следующим образом:

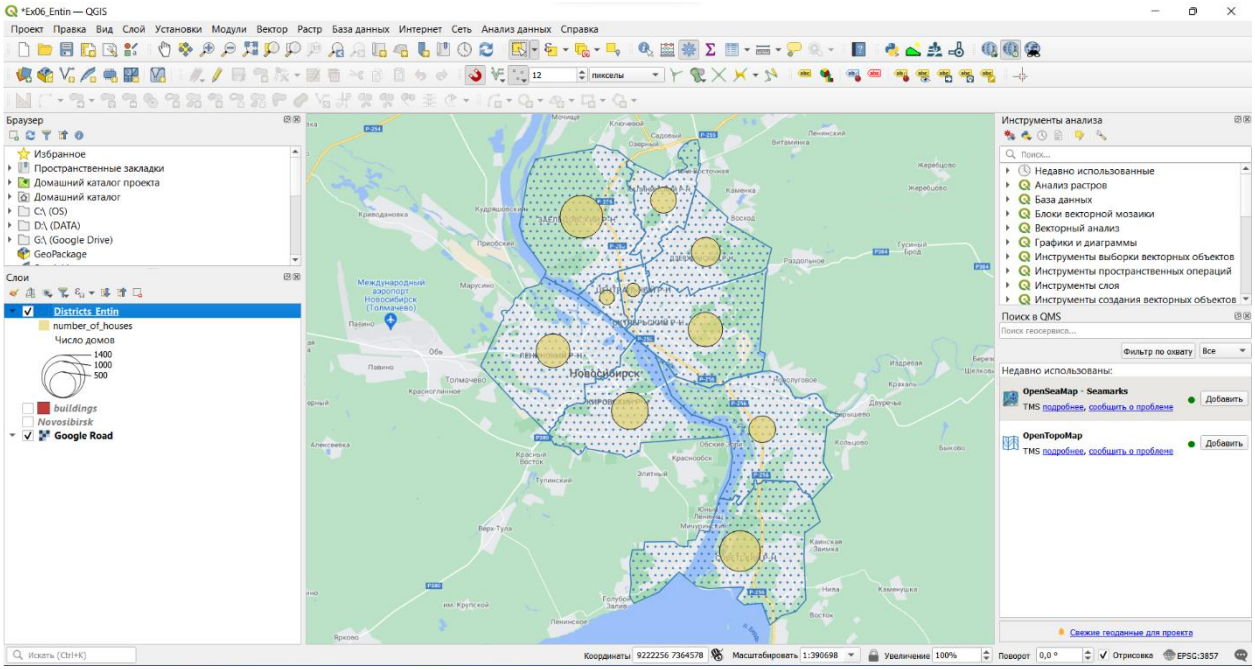


Рис.7.14. Отображение условных знаков в проекте

11. Самостоятельно измените настройки отображения слоёв так, чтобы базовая карта не «доминировала» над тематическим содержанием, и сделайте снимок экрана.

Практическаяработа 8. Создание картографической продукции на основе базы данных

Цель задания — знакомство с моделями пространственных объектов и пространственных данных, форматами хранения растровых и векторных данных. Визуализация данных на карте. Создание макета компоновки карты.

Необходимая теоретическая подготовка: модели пространственных данных, модели пространственных объектов, базы пространственных данных, картографические проекции, системы координат.

Исходные данные: база пространственных данных на территорию Кавказских гор, собранная из нескольких источников.

Изображение, которое вы видите во фрейме данных, можно экспортировать в отдельный графический файл «как есть» (с помощью опции **Проекты — Импорт/экспорт — Экспортировать карту как изображение…**). Однако для картографических целей, как правило, формируется **компоновка карты**. На листе заданного формата размещается картографическое изображение, добавляется название, легенда, масштабная линейка и элементы зарамочного оформления.

Сейчас мы создадим макет компоновки с расчётом на то, что итоговая карта будет вставлена в отчёт.

1. Создайте новый макет компоновки (**Проект — Создать Макет…**) или Ctrl+P.
2. Введите название макета: Ex01_%Фамилия%, где %Фамилия% — ваша фамилия на русском языке.

После ввода названия откроется окно компоновки (*Layout*)

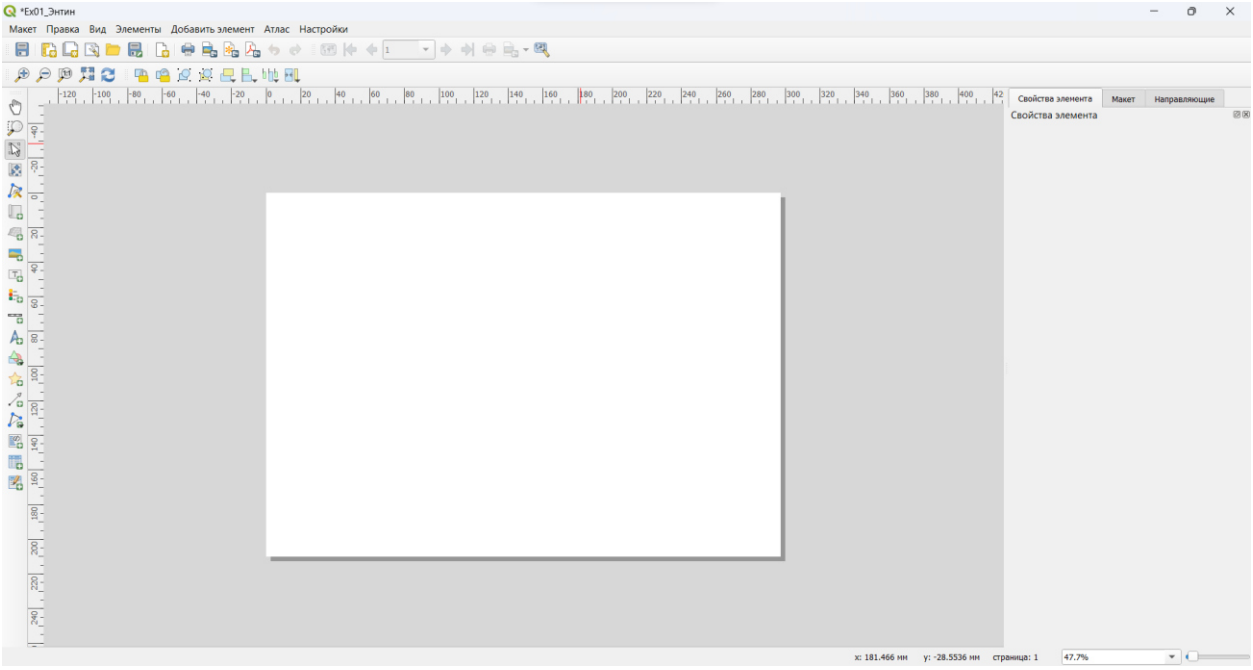
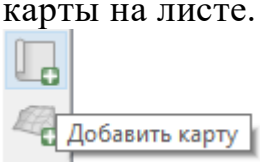


Рис. 8.1. Пространство макета компоновки

3. Добавьте на лист картографическое изображение. Для этого используется инструмент **Добавить карту** из панели инструментов. Выберите инструмент и «растяните» прямоугольник карты на листе.



Инструмент добавления карты

4. После добавления элемента откроется панель его свойств. Изучите настройки, доступные в этой панели, а затем установите для карты знаменатель масштаба *4 000 000* и размеры 237×130237×130 мм. В том же разделе, где устанавливаются размеры элемента, задайте для элемента карты положение по X=30X=30 мм и положение по Y=30Y=30 мм.

Положение элемента на листе отсчитывается от верхнего левого угла листа до точки привязки элемента.

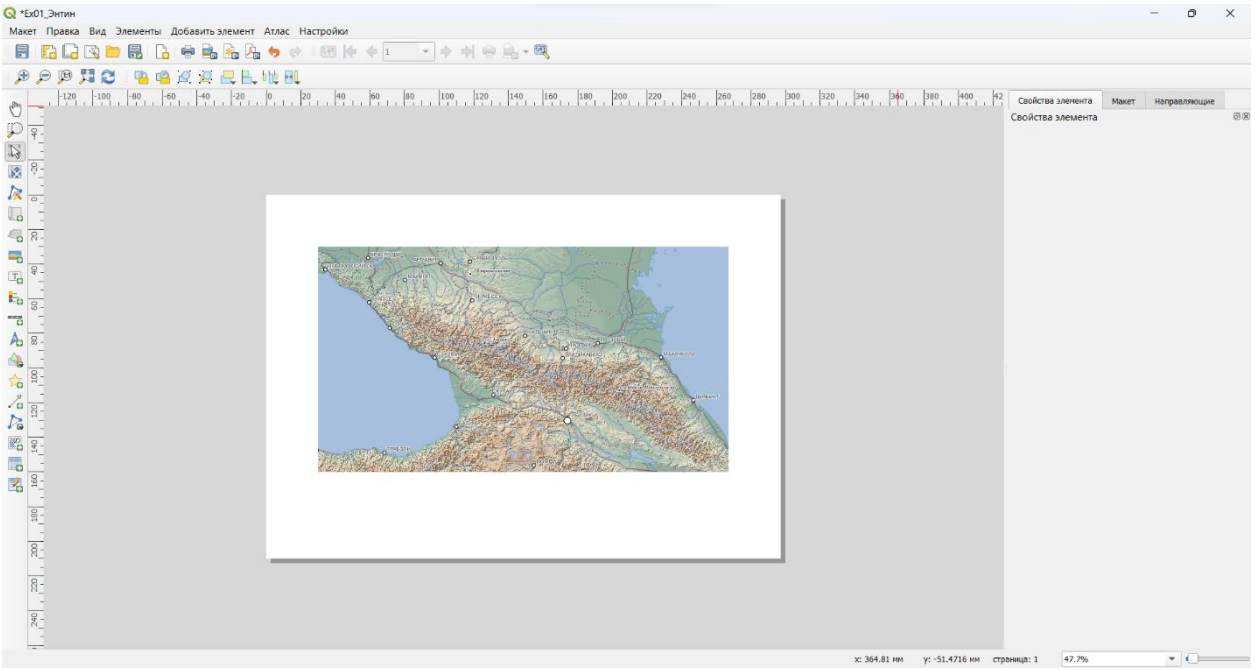


Рис. 8.2. Картографическое изображение в макете компоновки

5. Добавьте к карте градусную сетку. Для этого в свойствах элемента найдите раздел **Сетки**, нажмите на кнопку **Добавить новую сетку**, а затем **Настроить сетку…** (*Modify Grid*). Откроется меню настройки сетки. Задайте для сетки проекцию WGS84, интервал по долготе — 4°4°, интервал по широте — 2°2°. Также уменьшите толщину линий сетки до 0.10.1 мм. Для этого щёлкните левой кнокой мыши по элементу *Стиль линии* и выполните необходимые настройки в уже привычном для вас интерфейсе. Вернуться обратно к настройкам сетки можно, нажав на кнопку *Назад* в левом верхнем углу интерфейса.

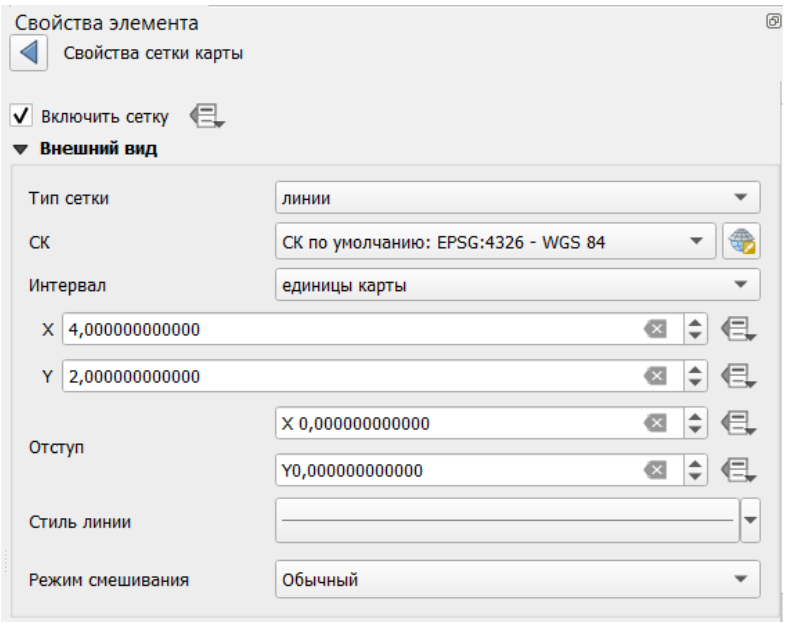


Рис. 8.3. Панель настроек сетки

6. Добавьте рамку сетки в виде простой линии. Для этого в свойствах элемента в разделе **Рамка** установите настройку **Стиль области**: **Линия границы**

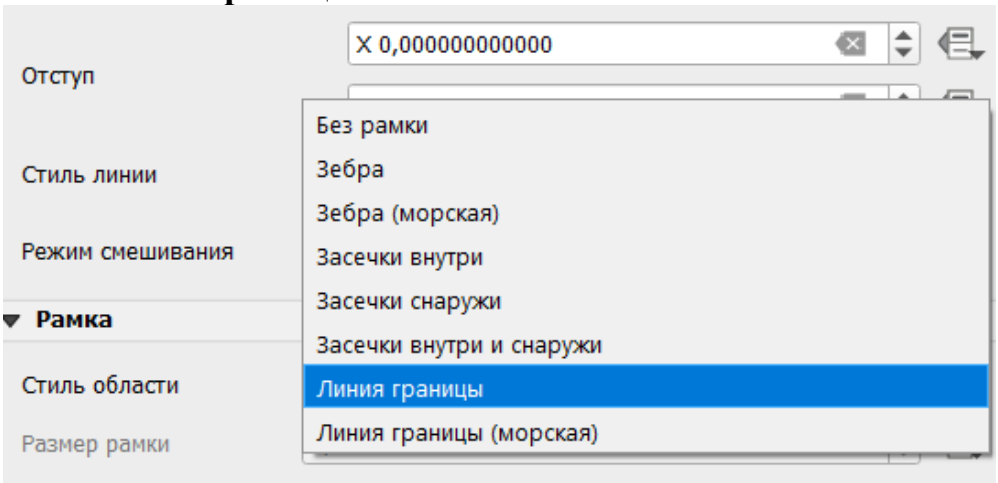


Рис. 8.4. Настройка рамки сетки

7. Включите отображение подписей координатной сетки. Для этого в настройках координатной сетки отметьте флажок **Отображение координат**. Настройте отображение подписей так, чтобы широта подписывалась только вдоль западной и восточной рамки, а долгота — только вдоль северной и южной. Используйте формат координат **Десятичные с окончанием** и нулевое число знаков после запятой (этот параметр в QGIS называется **Точность координат**).

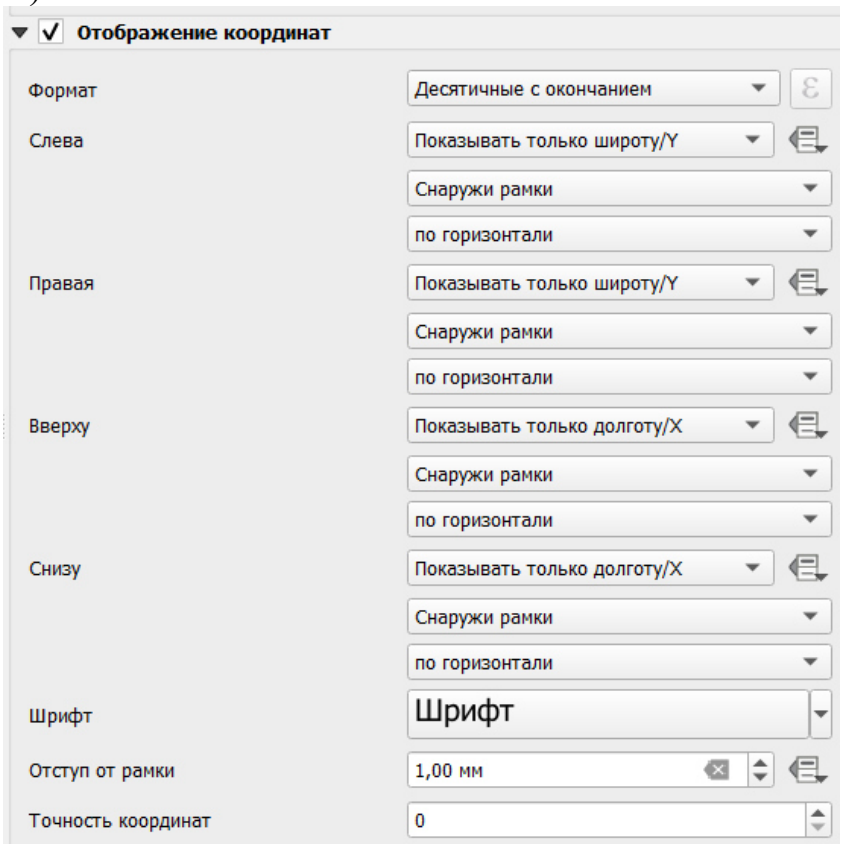


Рис. 8.5. Панель настроек координат сетки

8. Вернитесь к макету и передвиньте картографическое изображение внутри элемента таким образом, чтобы вместилась вся основная часть Главного Кавказского хребта. Можно ориентироваться на города: в северо-западном углу карты должен отображаться Краснодар, в юго-восточном — Баку. При необходимости можно уменьшить масштаб карты.

Для перемещения карты внутри фрейма используется инструмент **Перемещение содержимого элемента** .

9. Добавьте на лист название карты. Для этого вставьте новую надпись (с помощью инструмента **Добавить надпись**) и разместите её над элементом карты. Введите название карты «Кавказские горы», используйте выключку (горизонтальное выравнивание) по центру, настройте параметры шрифта на своё усмотрение (заголовки обычно набираются прописными буквами с разреженным кернингом).

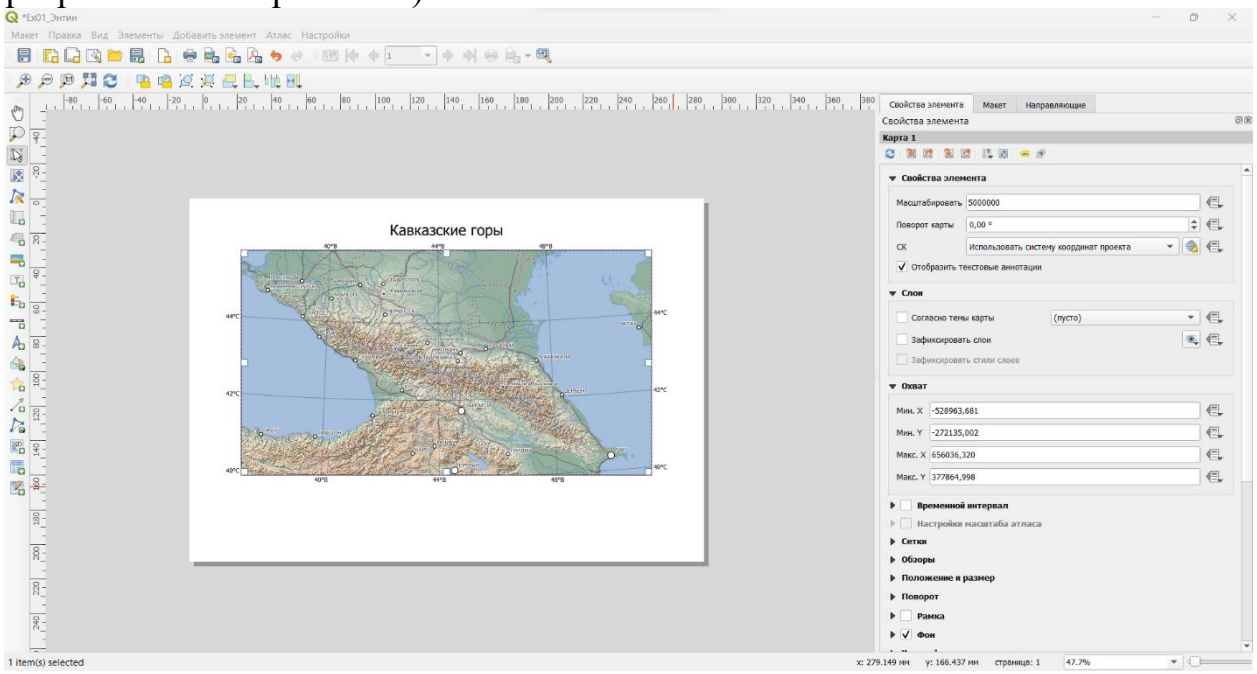


Рис. 8.6. Макет компоновки после добавления названия

10. Добавьте на лист масштабную линейку с помощью кнопки **Добавить масштабную линейку** . Переместите линейку в юго-западный угол карты, установите для неё отображение фона и границы. Если необходимо, уменьшите высоту линейки, кегль шрифта и отступы подписей, чтобы линейка смотрелась более компактно.

11. Добавьте на лист легенду с помощью кнопки **Добавить легенду** . Легенда будет собрана автоматически на основе тех настроек визуализации, которые применены для слоёв карты.

12. Отредактируйте легенду. Для этого сначала выключите автообновление (*Auto update*) элементов легенды, чтобы сделать список элементов доступным для редактирования. Сохраните в

легенде только условные знаки населённых пунктов и границ, а также переименуйте неинформативные или пустые подписи.

13. Добавьте обводку для элемента легенды и разместите легенду в северо-восточном углу карты.

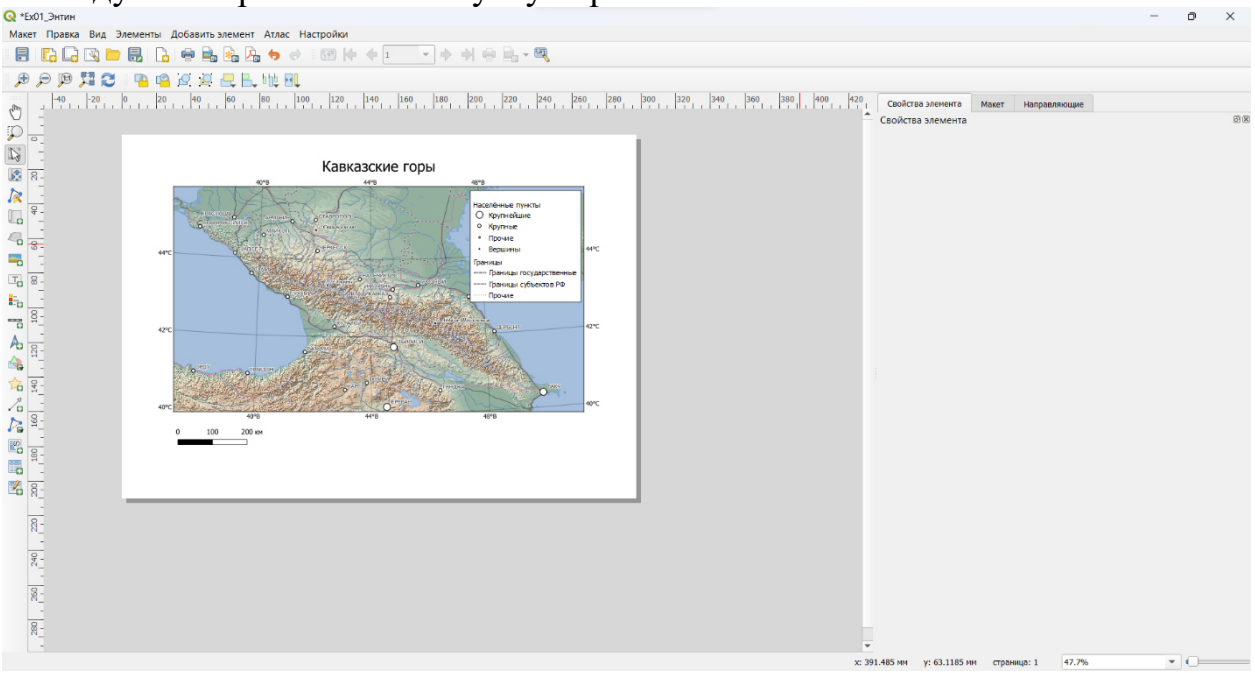


Рис. 8.7. Макет компоновки после добавления масштабной линейки и легенды

14. Добавьте ещё один текстовый элемент и впишите в него имя автора карты.

15. Экспортируйте получившуюся карту в изображение формата PNG («Макет» — «Экспорт в изображение…» или специальная кнопка на главной панели инструментов макета). При экспорте включите настройку **Обрезать по содержимому**, чтобы не экспортировать поля страницы.

16. Вставьте экспортированное изображение в отчётный файл.

Список литературы

1. Матушкин, А. С. Картографирование и анализ пространственных данных с использованием геоинформационной системы QGIS : учебное пособие / А. С. Матушкин. – Киров: ВятГУ, 2018. – 100 с. https://iweb.vyatsu.ru/document/Documents/OOP_Document/3-07.03.04.01/%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D0%BE%D0%B5%20%D0%BF%D0%BE%D1%81%D0%BE%D0%B1%D0%B8%D0%B5%20%D0%9A%D0%B0%D1%80%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%B8%20%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7%20%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D1%8B%D1%85%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85.pdf

2. Колесников, А. А. Базы пространственных данных : учебно-методическое пособие / А. А. Колесников, П. Ю. Бугаков. — Новосибирск : СГУГиТ, 2022. — 87 с. — ISBN 978-5-907513-38-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/317426

3. Основы геоинформатики: практикум в QGIS Андрей Энтин, Тимофей Самсонов, Андрей Карпачевский https://aentin.github.io/qgis-course/

БАЗЫ ГЕОПРОСТРАНСТВЕННЫХ ДАННЫХ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к проведению практических работ
для студентов, обучающихся по направлению подготовки:
05.04.03 Картография и геоинформатика
(программа: Геоинформационное моделирование)
всех форм обучения

Составители:
Васильчикова Екатерина Владимировна

Уч.- изд. л. 6

ФГБОУ ВО «Воронежский государственный технический университет»
394006 Воронеж, ул. 20-летия Октября, 84