

**Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»**

Кафедра кибернетики в системах организационного управления

Язык программирования Python

**Методические указания и задания
к выполнению курсового проекта для направления подготовки
09.03.03 «Прикладная информатика»**

Воронеж 2024

Составитель: В.Е. Белоусов

УДК 519.876

ББК

Язык программирования Python [Текст]: метод. указания и задания к выполнению курсового проекта для студентов направлений подготовки 09.03.03 «Прикладная информатика» / Воронеж. гос. техн. ун-т; сост.: В.Е. Белоусов. – Воронеж, 2024. – 35 с.

Приведены требования к содержанию курсового проекта, исходные данные и пример выполнения по дисциплине «Язык программирования Python».

Предназначены для студентов направлений 09.03.03 «Прикладная информатика» всех форм обучения.

Ил. 9. Табл. 15. Библиогр.: 6 назв.

Печатается по решению редакционно-издательского совета Воронежского государственного технического университета.

Рецензенты: кафедра проблем управления Московского физико-технического института (государственного университета)
А.К. Погодаев, д-р техн. наук, профессор, ректор
Липецкого государственного технического университета

Введение

В последние годы компьютеры проникли практически во все сферы человеческой деятельности и стали движущей силой прогресса. Для функционирования компьютеров требуется программное обеспечение, которое разрабатывается коллективами программистов.

Программирование – это научная и практическая деятельность, основным объектом изучения которой являются компьютерные программы. Программы рассматриваются как совокупность команд и данных, представленных в объектной форме и предназначенных для функционирования на вычислительной машине или управляющем устройстве с целью получения определенного конечного результата.

Объектно-ориентированное программирование – это методология разработки программ как совокупности объектов, относящихся к определенным классам, и взаимодействующих через свойства и методы друг друга для выполнения операций и преобразования данных, определяя поведение своих экземпляров. Свойства и методы образуют интерфейс объекта, а сущности с конкретным сочетанием свойств являются экземплярами объекта.

Эта методология полагает, что объекты окружающего мира обладают тремя составляющими: они имеют свойства, описывающие конкретные сущности, позволяют изменять эти свойства разными методами, реализующие интерфейс взаимодействия, и способны реагировать на события, возникающие как вне объекта, так и внутри самого объекта, определяя его поведение. Именно в таком смысле реализуется понятие объекта – как совокупность:

- свойств, т.е. набора простых и составных данных;
- методов, т.е. набора функций, реализующих алгоритмы обработки данных;
- событий, т.е. набора условий, на которые объект должен реагировать. Объектно-ориентированный подход вводит следующие понятия:

Класс – это составной тип данных, подразумевающий некоторый способ представления и структуризацию данных внутри себя, а также поведение, реализуемое через алгоритмы, оперирующие этими данными в вычислительном процессе. Таким образом, класс объединяет данные и алгоритмы в одно целое, и управляет доступом к ним со стороны других программных объектов. Класс является шаблоном, по которому создаются программные объекты. Обычно классы разрабатывают таким образом, чтобы они соответствовали объектам предметной области.

Свойство – это данные, находящиеся внутри класса. Это могут быть любые простые или составные данные, а также любое их сочетание, организованное в некоторую логически связанную структуру, и доступное для манипулирования в вычислительном процессе.

Метод – это алгоритм, реализующий обработку данных внутри класса. Это может быть функция или подпрограмма, которая каким-либо образом манипулирует данными внутри класса в ходе вычислительного процесса.

Экземпляр – это сущность в адресном пространстве вычислительного

процесса, появляющаяся при создании объекта с конкретным сочетанием свойств и размещением в памяти.

Интерфейс – это совокупность свойств и методов класса, которые доступны другим программным объектам для взаимодействия с его экземплярами в вычислительном процессе.

Объектно-ориентированный подход дает следующие преимущества:

- возможность управлять большими проектами, разбивая большие задачи на небольшие и легко разрабатываемые части;
- возможность многократного использования одного и того же кода при порождении множества объектов;
- возможность переопределения поведения объектов при использовании уже разработанных объектов;
- по-настоящему модульное программирование, сводящее к минимуму повторения при кодировании.

Методические указания предназначены для освоения студентами методологии объектно-ориентированного программирования посредством решения задач на разработку программ.

При выполнении работы студенты должны приобрести навыки составления алгоритмов, структурирования данных, написания собственных программ, тестирования программ, и их документального оформления.

Содержат цель, задание, требования к выполнению, комментарии, список пунктов оформления, варианты заданий, и пример выполнения с комментариями.

Указания к выполнению и оформлению курсового проекта

Общие требования:

- Работа выполняется индивидуально, номер варианта соответствует цифре в списке учебной ведомости для очных студентов.
- При выполнении работы помимо задания должны быть учтены требования комментарии к работе.
- Выполненная работа демонстрируется преподавателю. В процессе демонстрации студент должен понимать и уметь объяснить все этапы работы, а также использовать «правильную» терминологию.
- В разделе «Оформление» к работе указано содержание отчёта по работе.

Требования к оформлению отчетов:

- Отчёт оформляется в отдельном файле и загружается в Moodle.
- Дополнительно загружается файл java программы.
- Скриншоты экранов не допускаются.
- Выполнение работы осуществляется по модулю и включает 4 части.
-

Требования к оформлению листингов программ:

- Листинги программ оформляются шрифтом Courier 12 одиночным интервалом.
- Перед каждой разработанной функцией должен быть комментарий, соответствующий требованиям разработанного алгоритма.
- В теле функции должны быть поясняющие комментарии, если части алгоритма не являются очевидными.

При выполнении заданий допускается использование Giga IDE и Google Colab для языка программирования Python.

Цель работы – получить навыки программирования для решения сложной прикладной задачи с неявными параметрами.

Варианты заданий:

№ варианта	Группа (0-погибшие, 1-выжившие)	Условия
1	0	Женщины от 20 до 30 лет, севших в порту Квинстаун, имеющих братьев
2	0	Мужчины от 20 до 30 лет, севших в порту Шербур, имеющих сестер
3	0	Дети до 7 лет, севших в порту Саутгемптон, имеющих отца и мать
4	0	Женщины от 31 до 40 лет, севших в порту Квинстаун, имеющих дочерей
5	0	Мужчины от 31 до 40 лет, севших в порту Шербур, имеющих сыновей
6	1	Дети от 8 до 10 лет, севших в порту Шербур, имеющих отца и мать
7	1	Женщины от 41 до 50 лет, севших в порту Саутгемптон, имеющих сыновей
8	1	Мужчины от 41 до 50 лет, севших в порту Квинстаун, имеющих братьев
9	1	Дети от 11 до 13 лет, севших в порту Саутгемптон, имеющих сестер
10	1	Пассажиры 1 класса, имевшие более одной каюты, имеющие братьев
11	0	Пассажиры 2 класса, ехавшие в одной одной каюте, имеющие сестер
12	0	Пассажиры 3 класса, севшие в Квинстаун, имевшие либо брата либо сестру
13	0	Дети, имеющие 1 брата или сестру, севшие в порту Шербур, из полных семей
14	0	Дети, имеющие 2 братьев и сестер, севшие в порту Саутгемптон, из неполных семей
15	0	Дети, имеющие 1 родителя, севшие в порту Квинстаун, имеющие братьев
16	1	Дети, имеющие двух родителей, пассажиры 2 класса, имеющие братьев или сестер, севшие в порту Саутгемптон
17	1	Дети (мальчики), пассажиры 1 класса, севшие в порту Саутгемптон, Шербур, из полных семей
18	1	Дети (девочки), пассажиры 2 класса, ехавшие в одной одной каюте, имеющие сестер

Гибель Титаника является одним из самых печально известных кораблекрушений в истории. 15 апреля 1912 года во время своего первого плавания, Титаник затонул после столкновения с айсбергом, при этом погибло 1502 из 2224 пассажиров и членов экипажа.

Некоторые группы людей имели больше шансов выжить, по сравнению с другими. Например, женщины, дети, пассажиры высшего класса.

Для обучения применяются данные, размещенные на ресурсе Kaggle: [Titanic: Machine Learning from Disaster | Kaggle](#).

Для каждого из методов считается оценка методом кросс-валидации. Лучший из них применяется для прогнозирования результата.

Описание dataset Titanic

Набор данных включает в себя два CSV-файла.

Файл [train.csv](#) представляет собой обучающий набор, файл [test.csv](#) — тестовый набор.

Обучающий набор содержит признак [Survived](#) для каждого пассажира, обозначающий, выжил данный пассажир или нет (0 для умерших, 1 для выживших).

Каждая строка наборов данных содержит следующие поля:

- [Pclass](#) — класс пассажира (1 — высший, 2 — средний, 3 — низший);
- [Name](#) — имя;
- [Sex](#) — пол;
- [Age](#) — возраст;
- [SibSp](#) — количество братьев, сестер, сводных братьев, сводных сестер, супругов на борту титаника;
- [Parch](#) — количество родителей, детей (в том числе приемных) на борту титаника;
- [Ticket](#) — номер билета;
- [Fare](#) — плата за проезд;
- [Cabin](#) — каюта;
- [Embarked](#) — порт посадки (C — Шербур; Q — Квинстаун; S — Саутгемптон).

В поле [Age](#) приводится количество полных лет. Для детей меньше 1 года — дробное.

Если возраст не известен точно, то указано примерное значение в формате [xx.5](#).

Приведение данных

Многие методы машинного анализа работают только с числовыми данными. Поэтому необходимо принять некоторые соглашения по приведению нечисловых параметров к числовым.

Поля **Pclass**, **Age**, **Sibsp**, **Parch**, **Fare** являются числовыми и не требуют преобразования.

Визуальное изучение данных показало, что поля **Age** и **Fare** могут принимать пустые значения. Для пассажиров, значение которых не указано, предполагается использование медианы среди ненулевых значений этого параметра.

В поле **Sex** предполагается значения **male** заменить на 0, значения **female** — на 1.

Поле **Embarked** также может принимать пустое значение. Для таких записей поле **Embarked** будет заполнено значением **S**.

Преобразование к числу значений поля **Embarked** происходит следующим образом: **S** заменяется на 0, **C** — на 1, **Q** — на 2.

Для обучения используются следующие поля: **Pclass**, **Sex**, **Age**, **SibSp**, **Parch**, **Fare**, **Embarked**.

Список литературы

1. Никульчев Е. В., Алексеенко А. С., Ильин Д. Ю. Системы сбора и предобработки данных. Методы статистического анализа с использованием Google Colab, 2023 - 121 с.

2. Широбокова С. Н., Кацупеев А. А., Сулыз А. В. Программирование на языке Python, 2020 - 104 с. Южно-Российский государственный политехнический университет (НПИ) имени М.И. Платова/

ПРИЛОЖЕНИЕ

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Воронежский государственный технический университет»

Кафедра кибернетики в системах организационного управления



КУРСОВОЙ ПРОЕКТ

по дисциплине

Язык программирования Python

Тема: Методы продвинутой работы с датасетом Titanic.csv

Выполнил:

Проверил:

Воронеж 2024

ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ

по дисциплине Язык программирования Python
(название дисциплины)

Студенту _____
(фамилия, имя, отчество)

Тема: _____

Руководитель _____
(должность, ученая степень, ученое звание,

фамилия, имя, отчество)

ЦЕЛЕВАЯ УСТАНОВКА. Ознакомление с концепциями инкапсуляции и модульности. Изучение приемов работы с классами, конструкторами и деструкторами, разработка интерфейса методов класса, создание и работа с экземплярами класса. Освоение принципа «класс-элемент — класс-набор».

График выполнения курсового проекта

по курсу 2 для группы _____

ЗАДАНИЕ	СРОКИ ВЫПОЛНЕНИЯ
Библиография, план, введение	
Анализ предметной области.	
Формирование алгоритма решения задачи	
Формирование кода программы	
Выводы по работе.	
Сдача курсового проекта на проверку	
Защита курсового проекта	

Дата выдачи задания «__» _____ 202_ г.

Срок сдачи студентом законченного курсового проекта «__» _____ 202_ г.

Руководитель _____
(уч.степень, подпись, фамилия)

Задание принял к исполнению «__» _____ 202_ г. _____
(подпись, фамилия)

Зав. базовой кафедрой «КСОУ» _____
(уч.степень, подпись, фамилия)

Рассмотрено на заседании кафедры «Кафедра кибернетики в системах
организационного управления»

Протокол №__ от _____ 202_ г.