**МИНИCTEPCTBO НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«Северо-Кавказский федеральный университет» Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №11**

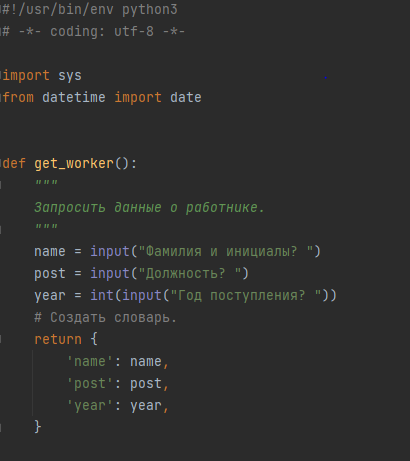
**по дисциплине «Основы программной инженерии»**

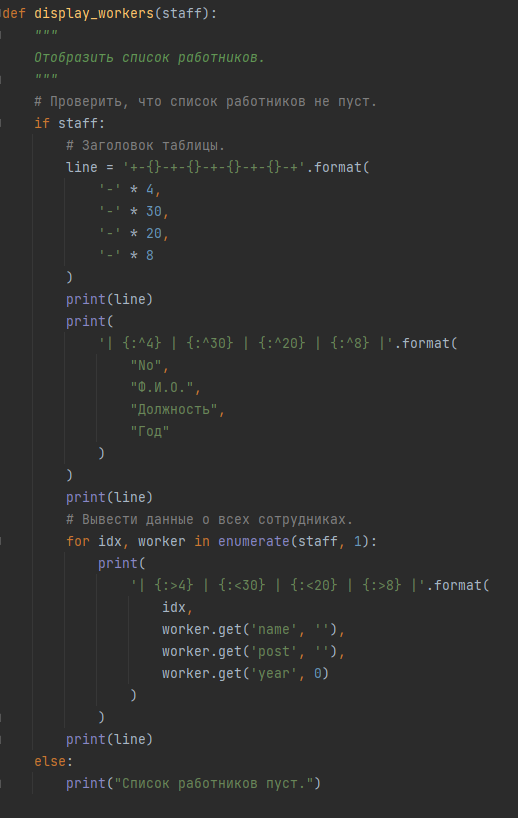
|  |
| --- |
| Выполнил студент группы ПИЖ-б-о-20-1 |
| Ваньянц И.М. « » 20 г. |
| Подпись студента |
| Работа защищена « » \_20 г. |
| Проверил Воронкин Р.А.  (подпись) |

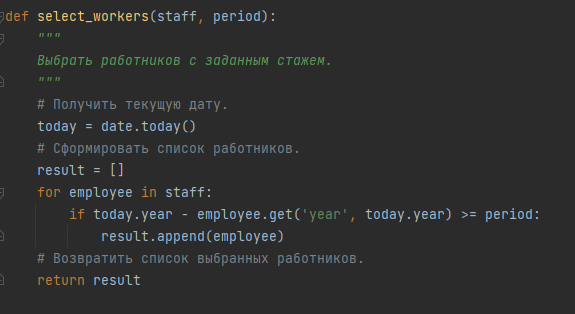
Ставрополь 2021

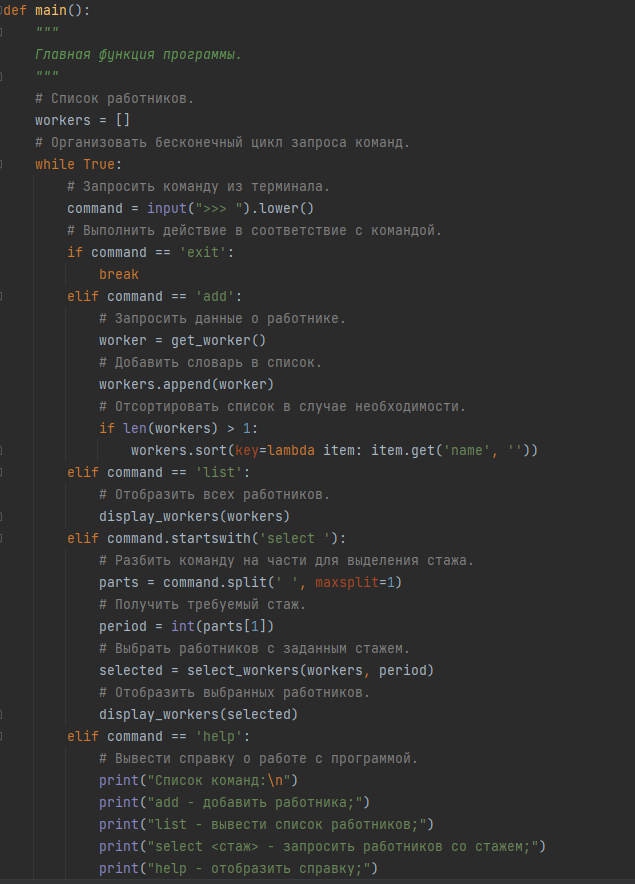
**ХОД РАБОТЫ**

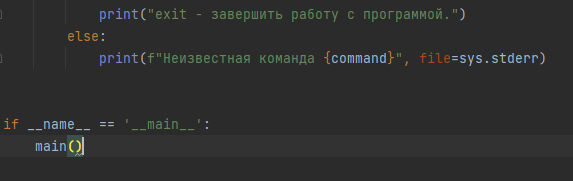
1. Пример 1











Рисунки 1, 2, 3, 4, 5 – код программы

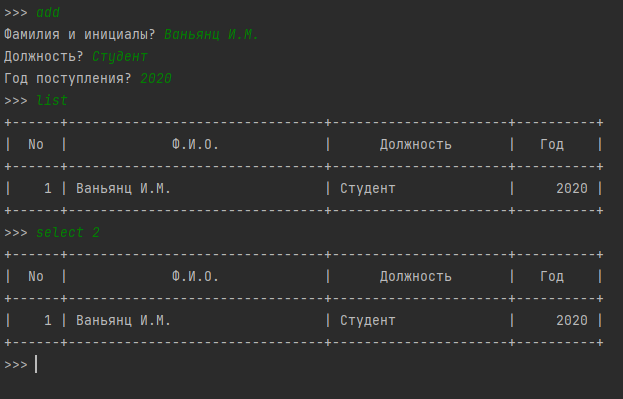


Рисунок 6 – результат выполнения программы

2. Пример 2

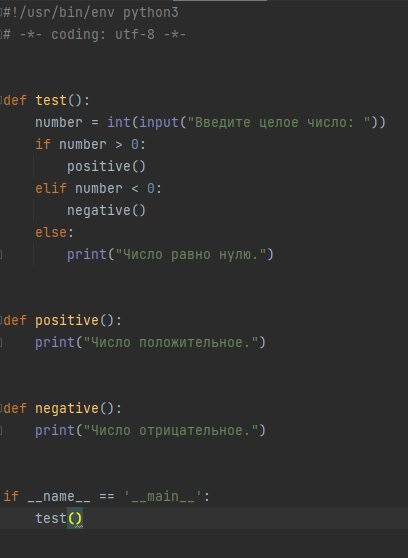


Рисунок 7 – код программы



Рисунок 8 – результат выполнения программы при 10

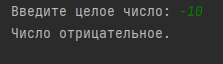


Рисунок 9 – результат выполнения программы при -10



Рисунок 10 – результат выполнения программы при 0

1. Пример 3

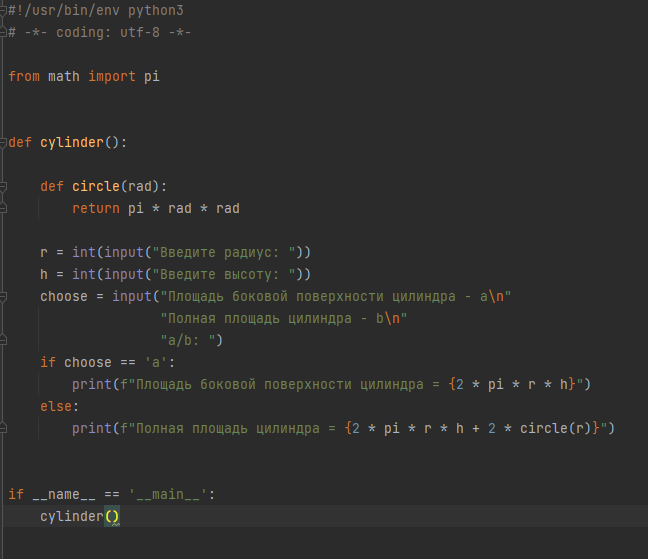


Рисунок 11 – код программы

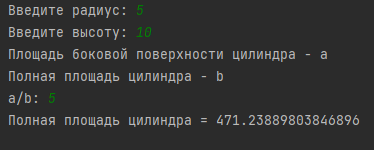


Рисунок 12 – результат выполнения программы

1. Пример 4

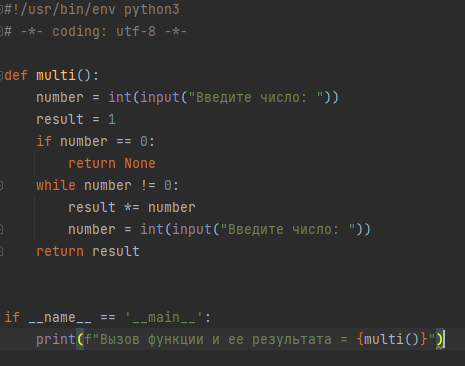


Рисунок 13 – код программы

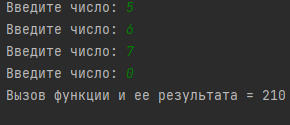


Рисунок 14 – результат выполнения программы

1. Пример 5

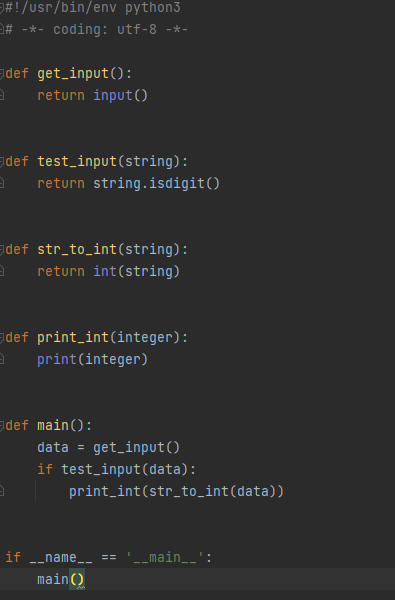


Рисунок 15 – код программы

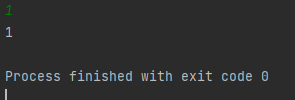
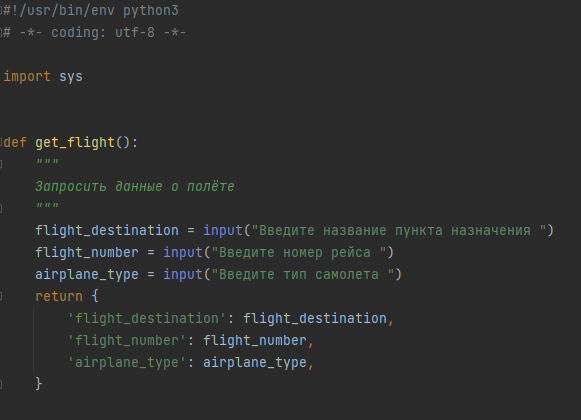
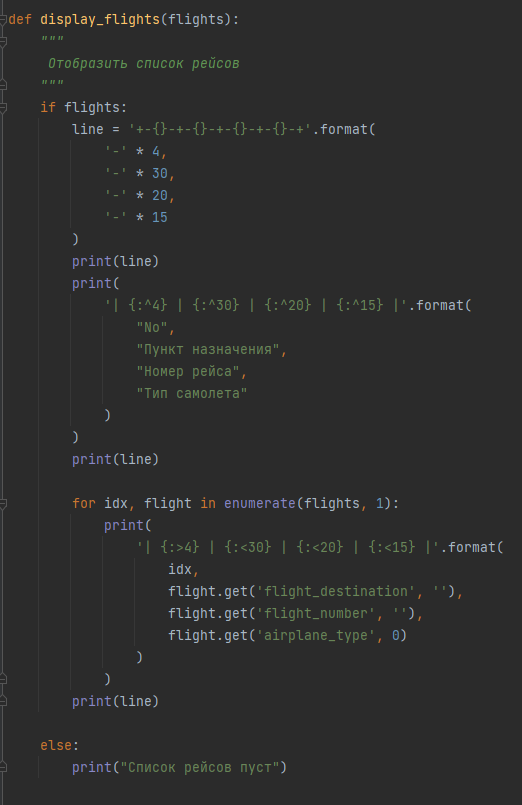
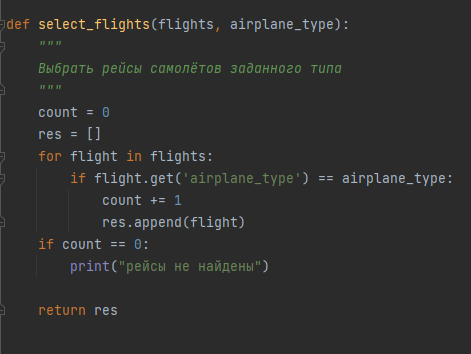


Рисунок 16 – результат выполнения программы

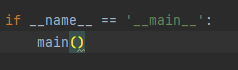
1. Индивидуальное задание











Рисунки 17, 18, 19, 20, 21 – код программы

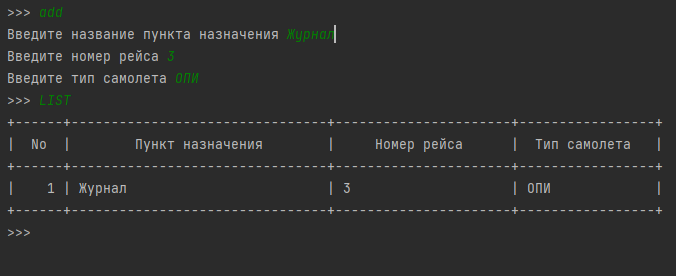


Рисунок 22 – результат выполнения программы

**ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Каково назначение функций в языке программирования Python?

Функция представляет собой обособленный участок кода, который можно вызывать, обратившись к нему по имени, которым он был назван. При вызове происходит выполнение команд тела функции.

1. Каково назначение операторов def и return?

В языке программирования Python функции определяются с помощью оператора def. Выход из функции и передача данных в то место, откуда она была вызвана, выполняется оператором return.

1. Каково назначение локальных и глобальных переменных при написании функций в Python?

Локальные переменные видны только в локальной области видимости, которой может выступать отдельно взятая функция. Глобальные переменные видны во всей программе. "Видны" – значит, известны, доступны. К ним можно обратиться по имени и получить связанное с ними значение. К глобальной переменной можно обратиться из локальной области видимости. К локальной переменной нельзя обратиться из глобальной области видимости, потому что локальная переменная существует только в момент выполнения тела функции.

1. Как вернуть несколько значений из функции Python?

В Питоне позволительно возвращать из функции несколько объектов, перечислив их через запятую после команды return.

1. Какие существуют способы передачи значений в функцию?

C помощью так называемых параметров, которые указываются в скобках в заголовке функции. Количество параметров может быть любым. Однако в Python у функций бывают параметры, которым уже присвоено значение по умолчанию. В таком случае, при вызове можно не передавать соответствующие этим параметрам аргументы. Хотя можно и передать.

1. Как задать значение аргументов функции по умолчанию?

def do\_smth(a, b=2) # Значение по умолчанию b = 2

1. Каково назначение lambda-выражений в языке Python?

Интересный синтаксис, позволяющий определять небольшие однострочные функции на лету. lambda – это выражение, а не инструкция. По этой причине ключевое слово lambda может появляться там, где синтаксис языка Python не позволяет использовать инструкцию def, – внутри литералов или в вызовах функций, например.

1. Как осуществляется документирование кода согласно PEP257?

• Тройные кавычки используются даже если строка помещается на одной линии. Это облегчает последующее расширение документации.

• Закрывающие кавычки находятся на той же строке, что и открывающие. Для однострочных docstring это выглядит лучше.

• Ни до, ни после документации не пропускаются строки. Код пишется сразу же на следующей линии

• Документационная строка — это «фраза», заканчивающаяся точкой. Она описывает эффект функции или метода в командном тоне.

• Однострочная документация НЕ должна быть простой «подписью», повторяющей параметры функции/метода Многострочные:

• Многострочные документации состоят из сводной строки (summary line) имеющей такую же структуру, как и однострочный docstring, после которой следует пустая линия, а затем более сложное описание.

• Оставляйте пустую строку после всех документаций (однострочных или многострочных), которые используются в классе;

• Документация скрипта (автономной программы) представляет из себя сообщение «о правильном использовании» и возможно будет напечатано, когда скрипт вызовется с неверными или отсутствующими аргументами.

• Документация модуля должна обычно содержать список классов, исключений и функций (и любых других важных объектов), которые экспортируются при помощи библиотеки, а также однострочное пояснение для каждого из них.

• Документация функции или метода должна описывать их поведение, аргументы, возвращаемые значения, побочные эффекты, возникающие исключения и ограничения на то, когда они могут быть вызваны.

• Документация класса должна обобщать его поведение и перечислять открытые методы, а также переменные экземпляра.

• Если класс является потомком и его поведение в основном наследуется от основного класса, в его документации необходимо упомянуть об этом и описать возможные различия.

9) В чем особенность однострочных и многострочных форм строк документации? Одиночные строки документации предназначены для действительно очевидных случаев. Они должны умещаться на одной строке. Многострочные строки документации состоят из однострочной строки документации с последующей пустой строкой, а затем более подробным описанием.