

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №4.1
«Элементы объектно-ориентированного программирования в языке Python»

Выполнила:
Кузнецова Алена Валерьевна
3 курс, группа ИВТ-б-о-21-1,
09.03.01 «Информатика
и вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем»,
очная форма обучения

(подпись)

Ставрополь, 2023 г.

Цель работы: приобретение навыков по работе с классами и объектами при написании программ с помощью языка программирования Python.

Выполнение работы:

1. Создали общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.

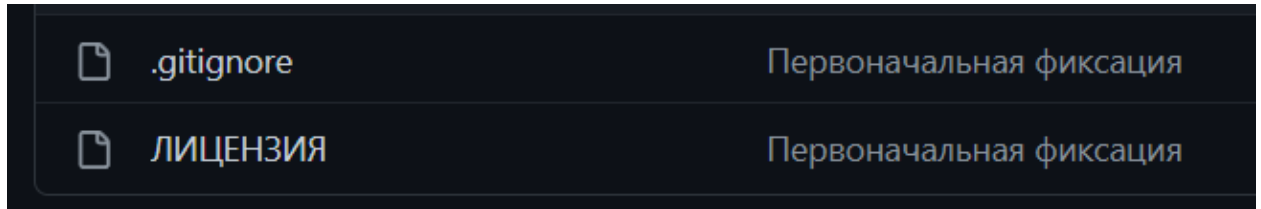


Рисунок 1 – Создание репозитория

2. Выполнили клонирование созданного репозитория.

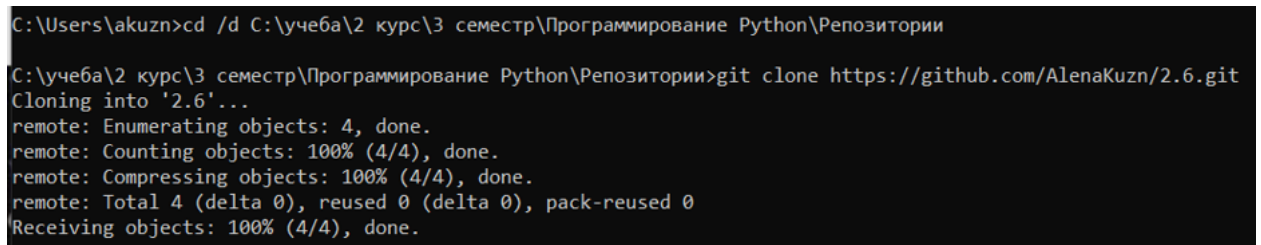


Рисунок 2 – Клонирование репозитория

3. Организовали свой репозиторий в соответствии с моделью git-flow.

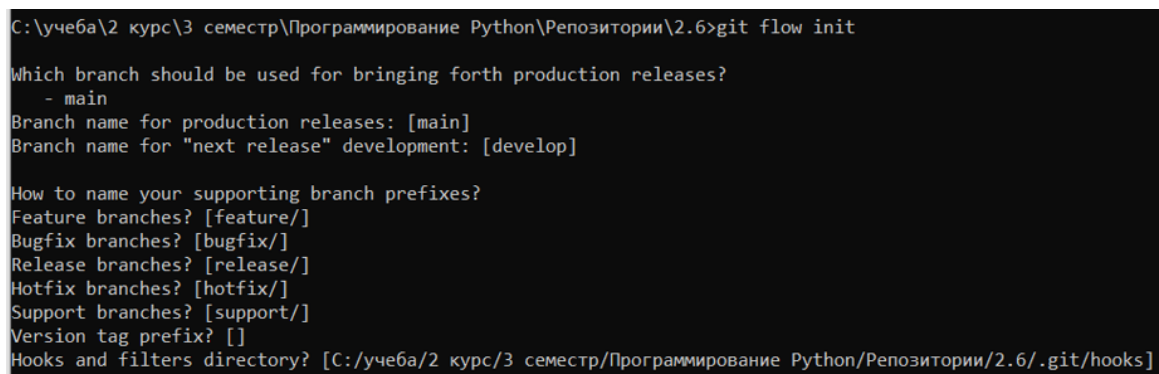
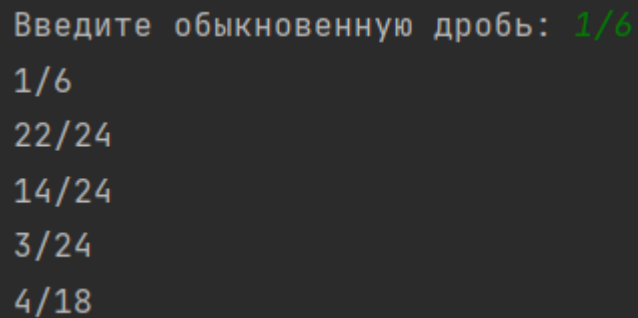


Рисунок 3 – Моделью ветвления git-flow

4. Проработали пример лабораторной работы:



```
Введите обыкновенную дробь: 1/6
1/6
22/24
14/24
3/24
4/18
```

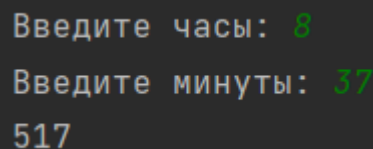
Рисуно 4 – Результат

5. Выполнили индивидуальное задание:

Вариант 9

Задание 1

Поле `first` — целое положительное число, часы; поле `second` — целое положительное число, минуты. Реализовать метод `minutes()` — приведение времени в минуты.



```
Введите часы: 8
Введите минуты: 37
517
```

Рисунок 5 – Результат задание 1

Задание 2

Реализовать класс `Account`, представляющий собой банковский счет. В классе должны быть четыре поля: фамилия владельца, номер счета, процент начисления и сумма в рублях. Открытие нового счета выполняется операцией инициализации. Необходимо выполнять следующие операции: сменить владельца счета, снять некоторую сумму денег со счета, положить деньги на счет, начислить проценты, перевести сумму в доллары, перевести сумму в евро, получить сумму прописью (преобразовать в числительное).

```
Введите фамилию владельца счета: Kuznetsova
Введите номер счета: 739475
Введите процент: 3
Введите баланс в рублях: 10500
Введите сумму для пополнения: 300
Баланс на счете: 10800 руб.
Баланс после начисления процентов: 11124
Введите сумму, которую хотите списать: 2000
Баланс на счете: 9124 руб.
Баланс в долларах: 92.16 $
Баланс в евро: 86.08 €
Баланс прописью: девять тысяч сто двадцать четыре
```

Рисунок 6 – Результат задание 2

Вывод: мы приобрели навыки по работе с классами и объектами при написании программ с помощью языка программирования Python.

Контрольные вопросы:

1. Как осуществляется объявление класса в языке Python?

Классы объявляются с помощью ключевого слова `class` и имени класса. Как правило, имя класса начинается с заглавной буквы и обычно является существительным или словосочетанием.

2. Чем атрибуты класса отличаются от атрибутов экземпляра?

Атрибуты класса являются общими для всех объектов этого класса, тогда как атрибуты экземпляра уникальны для каждого объекта.

3. Каково назначение методов класса?

Методы класса предназначены для выполнения определенных действий с данными класса. Они могут изменять значения атрибутов, выполнять вычисления, взаимодействовать с другими объектами и т.д.

4. Для чего предназначен метод `__init__()` класса?

Метод `__init__()` класса в Python используется для инициализации объектов класса, то есть задания начальных значений атрибутов объекта. Он вызывается автоматически при создании нового объекта класса. В методе

`__init__()` можно определить атрибуты объекта и задать им начальные значения.

5. Каково назначение `self`?

Аргумент `self` представляет конкретный экземпляр класса и позволяет нам получить доступ к его атрибутам и методам. В примере с `__init__` мы создаем атрибуты для конкретного экземпляра и присваиваем им значения аргументов метода. Важно использовать параметр `self` внутри метода, если мы хотим сохранить значения экземпляра для последующего использования.

6. Как добавить атрибуты в класс?

Для добавления атрибутов в класс нужно использовать ключевое слово `"self"` и указать имя атрибута внутри метода класса. Например:

```
class MyClass:
    def __init__(self, name):
        self.name = name
```

```
my_object = MyClass("John")
print(my_object.name)
```

7. Как осуществляется управление доступом к методам и атрибутам в языке Python?

Управление доступом к методам и атрибутам осуществляется с помощью модификаторов доступа: `public` (общедоступный), `private` (закрытый) и `protected` (защищенный).

`Public` - методы и атрибуты, к которым можно обратиться из любого места программы.

`Private` - методы и атрибуты, доступ к которым имеют только методы класса. Обозначаются двумя подчеркиваниями перед именем метода или атрибута.

`Protected` - методы и атрибуты, доступ к которым имеют только методы класса и его потомки. Обозначаются одним подчеркиванием перед именем метода или атрибута.

8. Каково назначение функции `isinstance`?

Функция `isinstance()` используется для проверки принадлежности объекта к определенному классу. Она принимает два аргумента: объект и тип (класс), и возвращает `True`, если объект является экземпляром этого класса, и `False` в противном случае. Например:

```
class MyClass:
```

```
    pass
```

```
my_object = MyClass()
```

```
print(isinstance(my_object, MyClass))
```