

## Отчет по практической

### Практическое занятие № 17

**Тема:** составление программ с использованием ООП.

**Цель:**

**Постановка задачи:**

**1.**

Создайте класс «Банк», который имеет атрибуты суммы денег и процентной ставки. Добавьте методы для вычисления процентных начислений и снятия денег.

**2.**

Создайте класс "Животное", который содержит информацию о виде и возрасте животного.

Создайте классы "Собака" и "Кошка", которые наследуются от класса "Животное" и содержат информацию о породе.

**Тип алгоритма 1:** Линейный

**Тип алгоритма 2:** Линейный

**Текст программы:**

**1.**

*# Создайте класс «Банк», который имеет атрибуты суммы денег и процентной ставки.*

*# Добавьте методы для вычисления процентных начислений и снятия денег.*

```
class Bank:
    def __init__(self, sum, prots):
        self.sum = sum
        self.prots = prots
    def prots_nachisl(self):
        self.res = (self.sum * self.prots)
        print('Процентные начисления составляют')
        return self.res
    def snyatie(self):
        self.res2 = self.res + self.sum
        print('Вы сняли со счета: ')
        return self.res2
```

```
BankOne = Bank(123, 12/100)
```

```
print(BankOne.prots_nachisl())
```

```
print(BankOne.snyatie())
```

**2.**

*# Создайте класс "Животное", который содержит информацию о виде и возрасте*

*# животного. Создайте классы "Собака" и "Кошка", которые наследуются от класса*

*# "Животное" и содержат информацию о породе.*

```
class Animal:
    def __init__(self, vid, age):
        self.vid = vid
        self.age = age

    def get_animal(self):
        return self.vid, self.age
```

```
class Dog(Animal):
    def __init__(self, vid, age, poroda_dog):
        Animal.__init__(self, vid, age)
        self.poroda_dog = poroda_dog
```

```
def get_dog(self):  
    return self.vid, self.age, self.poroda_dog
```

```
class Cat(Animal):  
    def __init__(self, vid, age, poroda_cat):  
        Animal.__init__(self, vid, age)  
        self.poroda_cat = poroda_cat  
  
    def get_cat(self):  
        return self.vid, self.age, self.poroda_cat
```

```
AnimalOne = Animal('Мышь', 1)  
DogOne = Dog('Собака', 3, "Бульдог Вик")  
CatOne = Cat('Кошка', 2, "Вислоухая Гато")  
print(AnimalOne.get_animal())  
print(DogOne.get_dog())  
print(CatOne.get_cat())
```

### **Протокол работы программы:**

**1.**

**Процентные начисления составляют  
14.76**

**Вы сняли со счета:  
137.7**

**2.**

**('Мышь', 1)  
( 'Собака', 3, 'Бульдог Вик')  
( 'Кошка', 2, 'Вислоухая Гато')**