

Костин ИС-23

Вариант 12.

Практическое занятие № 17

Тема: Составление программ с регулярными выражениями в IDE PyCharm Community.

Цели практического занятия: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с классами IDE PyCharm Community.

№ 17.1

Постановка задачи:

Создайте класс "Животное" с атрибутами "имя" и "вид". Напишите метод, который выводит информацию о животном в формате "Имя: имя, Вид: вид".

Тип алгоритма: ООП.

Текст программы:

```
# Вариант 12
# Создайте класс "Животное" с атрибутами "имя" и "вид". Напишите метод, который
# выводит информацию о животном в формате "Имя: имя, Вид: вид".
class Animal:
    def __init__(self, name, vid):
        self.name = name
        self.vid = vid

    def display_info(self):
        print(f'Имя: {self.name}, Вид: {self.vid}')

cat = Animal('кот', 'кошачьи')
dog = Animal('Собакакка', 'собачьи')
cat.display_info()
dog.display_info()
```

Протокол работы программы:

Имя: кот, Вид: кошачьи

Имя: Собакакка, Вид: собачьи

Process finished with exit code 0

№ 17.2

Постановка задачи:

Создайте базовый класс "Транспорт" со свойствами "марка", "модель" и "год выпуска". От этого класса унаследуйте класс "Автомобиль" и добавьте в него свойство "тип кузова".

Тип алгоритма: ООП.

Текст программы:

```
# Вариант 12
# Создайте базовый класс "Транспорт" со свойствами "марка", "модель" и "год
# выпуска". От этого класса унаследуйте класс "Автомобиль" и добавьте в него
# свойство "тип кузова".
class Transport:
    def __init__(self, brand, model, release_date):
        self.brand = brand
        self.model = model
        self.release_date = release_date

    def __str__(self):
        return f'марка: {self.brand}; модель: {self.model}; год выпуска:
{self.release_date}'

class Automobile(Transport):
    def __init__(self, brand, model, release_date, k_type):
        super().__init__(brand, model, release_date)
        self.k_type = k_type

    def __str__(self):
        return f'марка: {self.brand}; модель: {self.model}; год выпуска:
{self.release_date}; тип кузова: {self.k_type}'

trans = Transport('BMX', 'Ultra', 2018)
car = Automobile('Man', 'Super', 2021, 'КРУТОЙ!!!')
print(trans)
print(car)
```

Протокол работы программы:

марка: BMX; модель: Ultra; год выпуска: 2018

марка: Man; модель: Super; год выпуска: 2021; тип кузова: КРУТОЙ!!!

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ с классами в IDE PyCharm Community. Были использованы языковые конструкции `class` и др.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.