

## Отчет по практической

### Практическое занятие № 17

Тема: составление программ с использованием ООП.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с ООП в IDE PyCharm Community.

Постановка задачи:

1. Создайте класс «Студент», который имеет атрибуты имя, фамилия и оценки. Добавьте методы для вычисления среднего балла и определения, является ли студент отличником.
2. Создайте класс "Автомобиль", который содержит информацию о марке, модели и годе выпуска. Создайте класс "Грузовик", который наследуется от класса "Автомобиль" и содержит информацию о грузоподъемности. Создайте класс "Легковой автомобиль", который наследуется от класса "Автомобиль" и содержит информацию о количестве пассажиров.

Тип алгоритма 1 программы: цикла, ветвления

Тип алгоритма 2 программы: линейный

Текст 1 программы:

```
# Создайте класс «Студент», который имеет атрибуты имя, фамилия и оценки.  
# Добавьте методы для вычисления среднего балла и определения, является ли студент  
# отличником
```

```
class Student:
```

```
    def __init__(self, name, last_name, marks1, marks2, marks3):  
        self.name = name  
        self.last_name = last_name  
        self.marks1 = marks1  
        self.marks2 = marks2  
        self.marks3 = marks3
```

```
    def gpa(self):
```

```
        self.result = (self.marks1 + self.marks2 + self.marks3) / 3  
        self.result_text = "Средний балл у Студента", self.name, self.last_name, ':', self.result  
        self.string = "  
        for i in self.result_text:  
            self.string += str(i)  
            self.string += '  
        return self.string
```

```
    def the_best(self):
```

```
        self.string_1 = "  
        if self.result >= 4.5:  
            self.result_text_victory = "Студент ", self.name, self.last_name, "отличник!"  
            for i in self.result_text_victory:  
                self.string_1 += str(i)  
                self.string_1 += '  
            return self.string_1  
        else:  
            self.result_text_fail = "Студент ", self.name, self.last_name, ", к сожалению, не  
отличник!"  
            for i in self.result_text_fail:  
                self.string_1 += str(i)  
                self.string_1 += '  
        return self.string_1
```

```
StudentOne = Student("Игорь", "Самойлов", 5, 4, 2)
print(StudentOne.gpa())
print(StudentOne.the_best())
```

Протокол работы 1 программы:

Средний балл у Студента Игорь Самойлов : 3.6666666666666665

Студент Игорь Самойлов , к сожалению, не отличник!

Process finished with exit code 0

Текст 2 программы:

```
# Создайте класс "Автомобиль", который содержит информацию о марке, модели и
# годе выпуска. Создайте класс "Грузовик", который наследуется от класса
# "Автомобиль" и содержит информацию о грузоподъемности. Создайте класс
# "Легковой автомобиль", который наследуется от класса "Автомобиль" и содержит
# информацию о количестве пассажиров.
```

```
class Automobile:
```

```
    def __init__(self, mark, model, year):
        self.mark = mark
        self.model = model
        self.year = year
```

```
    def get_auto(self):
        return self.mark, self.model, self.year
```

```
class Truck(Automobile):
```

```
    def __init__(self, mark, model, year, capacity):
        Automobile.__init__(self, mark, model, year)
        self.capacity = capacity
```

```
    def get_truck(self):
        return self.mark, self.model, self.year, self.capacity
```

```
class Car(Automobile):
```

```
    def __init__(self, mark, model, year, capacity_people):
        Automobile.__init__(self, mark, model, year)
        self.capacity_people = capacity_people
    def get_car(self):
        return self.mark, self.model, self.year, self.capacity_people
```

```
AutomobileOne = Automobile("Mercedes", "Benz C-W205", 2019)
print(AutomobileOne.get_auto())
AutomobileTwo = Truck("Volvo", "VNL", 2020, 3000)
print(AutomobileTwo.get_truck())
AutomobileThree = Car("Toyota", "Camry", 2021, 5)
print(AutomobileThree.get_car())
```

Протокол работы 2 программы:

('Mercedes', 'Benz C-W205', 2019)

('Volvo', 'VNL', 2020, 3000)

('Toyota', 'Camry', 2021, 5)

Process finished with exit code 0

Вывод: : Я закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ с использованием ООП. Выполнены разработка кода, отладка, тестирование и оптимизация программного кода. Готовый код выложен на GitHub