

## Практическое занятие №16

**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 «Счетчик», который имеет атрибут текущего значения и методы для инкремента и декремента значения.
  
- 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.

### **1) Текст программы:**

```
1 import pickle
2
3
3 usages  👤 Maksim Beloglazov
4 class Counter:
5     👤 Maksim Beloglazov
6     def __init__(self):
7         self.counter_num = 0
8
9     3 usages  👤 Maksim Beloglazov
10    def increment(self):
11        self.counter_num += 1
12
13    1 usage  👤 Maksim Beloglazov
14    def decrement(self):
15        self.counter_num -= 1
16
17    1 usage  👤 Maksim Beloglazov
18    def save_def(filename, instance):
19        with open(filename, 'wb') as f:
20            pickle.dump(instance, f)
21
22    1 usage  👤 Maksim Beloglazov
23    def load_def(filename):
24        with open(filename, 'rb') as f:
25            return pickle.load(f)
```

```
23     # Сохранение экземпляров класса
24     counter1 = Counter()
25     counter1.increment()
26
27     counter2 = Counter()
28     counter2.increment()
29     counter2.increment()
30
31     counter3 = Counter()
32     counter3.decrement()
33
34     save_def( filename: 'counters.data', instance: [counter1, counter2, counter3])
35
36     # Загрузка экземпляров класса из файла
37     loaded_counters = load_def('counters.data')
38
39     # Демонстрация загруженных экземпляров
40     for counter in loaded_counters:
41         print(counter.counter_num)
42
```

2) Текст программы:

```
2 usages  Maksim Beloglazov
1  class Car:
    Maksim Beloglazov
2  def __init__(self, brand: str, model: str, year: int):
3      self.brand = brand
4      self.model = model
5      self.year = year
6
7
    Maksim Beloglazov
8  class Truck(Car):
    Maksim Beloglazov
9      def __init__(self, brand: str, model: str, year: int, carrying_capacity: float):
10         super().__init__(brand, model, year)
11         self.carrying_capacity = carrying_capacity
12
13
    Maksim Beloglazov
14  class PassengerCar(Car):
    Maksim Beloglazov
15      def __init__(self, brand: str, model: str, year: int, passenger_capacity: int):
16         super().__init__(brand, model, year)
17         self.passenger_capacity = passenger_capacity
18
```

**Вывод:** В процессе выполнения практического занятия №16 я закрепил понятия, алгоритмы, основные принципы составления программ и приобрел навыки составления программ с ООП в IDE PyCharm Community.