

Практическая работа №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.

Текст программы:

```
""" Создать класс «Счетчик», который имеет атрибут текущего значения и методы для инкремента и декремента значения """

class Counter:
    def __init__(self, value=0):
        self.value = value

    def increment(self):
        self.value += 1

    def decrement(self):
        self.value -= 2

counter = Counter(5)
print(f'Исходное значение: {counter.value}')
counter.increment()
print(f'Значение после инкремента: {counter.value}')
counter.decrement()
print(f'Значение после декремента: {counter.value}')
```

Протокол работы программы:

Исходное значение: 5

Значение после инкремента: 6

Значение после декремента: 5

Process finished with exit code 0

Постановка задачи 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.

Текст программы:

```
"""
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

class Dog(Animal):
    def __init__(self, vid, age, poroda):
        super().__init__(vid, age)
        self.poroda = poroda

class Cat(Animal):
    def __init__(self, vid, age, poroda):
        super().__init__(vid, age)
        self.poroda = poroda

dog = Dog('Собака', 5, 'Немецкая овчарка')
print(f'Вид: {dog.vid}.  Возраст: {dog.age}.  Порода: {dog.poroda}.')
cat = Cat('Кошка', 7, 'Русская голубая')
print(f'Вид: {cat.vid}.  Возраст: {cat.age}.  Порода: {cat.poroda}.')
```

Протокол работы программы:

Вид: Собака. Возраст: 5. Порода: Немецкая овчарка.

Вид: Кошка. Возраст: 7. Порода: Русская голубая.

Process finished with exit code 0

Постановка задачи 3:

Для задачи из блока 1 создать две функции, `save_def` и `load_def`, которые позволяют сохранять информацию из экземпляров класса (3 шт.) в файл и загружать ее обратно. Использовать модуль `pickle` для сериализации и десериализации объектов Python в бинарном формате.

Текст программы:

```
"""Для задачи из блока 1 создать две функции, save_def и
load_def, которые позволяют
сохранять информацию из экземпляров класса (3 шт.) в файл и
загружать ее обратно.
Использовать модуль pickle для сериализации и десериализации
объектов Python в
бинарном формате."""

import pickle

class Counter:
    """Задача блока 1"""
    def __init__(self, value=0):
        self.value = value

    def increment(self):
        self.value += 1
        return self.value

    def decrement(self):
        self.value -= 2
        return self.value

def save_def(count, name):
    with open(name, 'wb') as f:
        pickle.dump(count, f)

def load_def(name):
    with open(name, 'rb') as f:
        return pickle.load(f)

counter1 = Counter(5)
counter2 = Counter(10)
counter3 = Counter(15)

save_def([counter1, counter2, counter3], 'counters.bin')
loaded_counters = load_def('counters.bin')

for counter in loaded_counters:
    print(f'Значение счётчика: {counter.value}, Инкремент:
{counter.increment()}, Декремент: {counter.decrement()}')
```

Протокол работы программы:

Значение счётчика: 5, Инкремент: 6, Декремент: 4

Значение счётчика: 10, Инкремент: 11, Декремент: 9

Значение счётчика: 15, Инкремент: 16, Декремент: 14

Process finished with exit code 0

Так же создаётся файл counters.bin

Вывод: в процессе выполнения заданий я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием ООП в IDE PyCharm Community.

Выполнены разработка программного кода, отладка, тестирование, оптимизация кода.

Использованы языковые конструкции: import pickle, class, with open, def. Готовые программные коды выложены на GitHub.