

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 «Круг», который имеет атрибут радиуса и методы для вычисления площади, длины окружности и диаметра.

Текст программы:

```
"""Создайте класс «Круг», который имеет атрибут радиуса и методы для вычисления
площади, длины окружности и диаметра.
"""
import math

class Circle:
    def __init__(self, radius):
        self.radius = radius

    def area(self):
        return math.pi * (self.radius ** 2)

    def lenght(self):
        return 2 * math.pi * self.radius

    def diameter(self):
        return 2 * self.radius

radius = 20
print(f"Радиус круга: {radius}")
circle = Circle(radius)
area = circle.area()
print(f"Площадь круга: {area}")
lenght = circle.lenght()
print(f"Длина окружности: {lenght}")
diameter = circle.diameter()
print(f"Диаметр круга: {diameter}")
```

Протокол работы программы:

Радиус круга: 20

Площадь круга: 1256.6370614359173

Длина окружности: 125.66370614359172

Диаметр круга: 40

Process finished with exit code 0

Постановка задачи № 2:

Создайте класс "Человек", который содержит информацию о имени, возрасте и поле. Создайте классы "Мужчина" и "Женщина", которые наследуются от класса "Человек". Каждый класс должен иметь метод, который выводит информацию о поле объекта.

Текст программы:

```
""" Создайте класс "Человек", который содержит информацию о имени, возрасте и поле.
Создайте классы "Мужчина" и "Женщина", которые наследуются от класса
"Человек". Каждый класс должен иметь метод, который выводит информацию о
поле объекта.
"""
class Human:
    def __init__(self, name, age, sex):
        self.name = name
        self.age = age
        self.sex = sex

class Man(Human):
    def __init__(self, name, age):
        super().__init__(name, age, "мужской")

    def sex_1(self):
        return f'{self.name}, {self.age} - мужчина'

class Woman(Human):
    def __init__(self, name, age):
        super().__init__(name, age, 'женский')

    def sex_1(self):
        return f'{self.name}, {self.age} - женщина'

man = Man('Олег', 12)
woman = Woman('Оля', 15)
print(man.sex_1())
print(woman.sex_1())
```

Протокол работы программы:

Олег, 12 - мужчина

Оля, 15 - женщина

Process finished with exit code 0

Постановка задачи № 3:

Создайте класс «Круг», который имеет атрибут радиуса и методы для вычисления площади, длины окружности и диаметра (с использованием pickle и функций).

Текст программы:

```
"""Создайте класс «Круг», который имеет атрибут радиуса и методы для вычисления
площади, длины окружности и диаметра
(с использованием pickle и функций)."""
import math
import pickle

class Circle:

    def __init__(self, radius):
        self.radius = radius

    def area(self):
        return math.pi * (self.radius ** 2)

    def length(self):
        return 2 * math.pi * self.radius

    def diameter(self):
        return 2 * self.radius

method = [Circle.area, Circle.length, Circle.diameter]

def save_def(method, filename):
    with open(filename, 'wb') as f:
        pickle.dump(method, f)

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

radius = 10
circle = Circle(radius)
area = circle.area()
length = circle.length()
diameter = circle.diameter()

save_def(Circle.area, 'area.bin')
save_def(Circle.length, 'length.bin')
save_def(Circle.diameter, 'diameter.bin')

loaded_area = load_def('area.bin')
loaded_length = load_def('length.bin')
loaded_diameter = load_def('diameter.bin')

print(f"Радиус круга: {radius}, Площадь: {loaded_area(circle)}, Длина
окружности: {loaded_length(circle)}, Диаметр круга: {loaded_diameter(circle)}")
```

Протокол работы программы:

Радиус круга: 10, Площадь: 314.1592653589793, Длина окружности: 62.83185307179586,
Диаметр круга: 20

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ с использованием ООП в IDE PyCharm Community. Были использованы языковые конструкции классов.

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

Готовые программные коды выложены на GitHub.