

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.

Код программы: 1)

```
# Приложение ВЫДАЧА КРЕДИТОВ для некоторой организации. БД должна
# содержать таблицу Клиент со следующей структурой записи: ФИО клиента, ФИО
# сотрудника банка, срок кредита, процент кредита, сумма кредита

class Animal:
    def __init__(self, name, species):
        self.name = name
        self.species = species

    def info(self):
        return f"Name: {self.name}, Species: {self.species}"

animal_list = [Animal("Tiger", "Predator"), Animal("Elephant",
"Herbivore"),
                Animal("Cheetah", "Predator")]

def get_animal(valid_animals):
    while True:
        print("Available animals:")
        for i, animal in enumerate(valid_animals):
            print(f"{i + 1}: {animal.name}")
        choice = input("Enter the animal number (1-3): ")
        if choice.isdigit():
            choice = int(choice)
            if 1 <= choice <= 3:
                return valid_animals[choice - 1]
            else:
                print("Invalid animal number. Please choose a number
between 1 and 3.")
        else:
            print("Please enter a numerical value.")

chosen_animal = get_animal(animal_list)

print(chosen_animal.info())
```

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.
class Person:
    def __init__(self, name, age, gender):
        self.name = name
        self.age = age
        self.gender = gender

    def display_info(self):
```

```

        print(f"Name: {self.name}")
        print(f"Age: {self.age}")
        print(f"Gender: {self.gender}")

class Man(Person):
    def __init__(self, name, age):
        super().__init__(name, age, "Male")

    def display_info(self):
        print("Man's Information:")
        super().display_info()

class Woman(Person):
    def __init__(self, name, age):
        super().__init__(name, age, "Female")

    def display_info(self):
        print("Woman's Information:")
        super().display_info()

person1 = Person("Alex", 30, "Male")
print("Person's Information:")
person1.display_info()
print("-----")

man1 = Man("John", 25)
man1.display_info()
print("-----")

woman1 = Woman("Emma", 28)
woman1.display_info()

```

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