

Отчет о практическом занятии

Практическое занятие № 16 Вариант 5

Тема: составление программ для работы с классами в IDE PyCharm Professional.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ для работы с классами IDE PyCharm Community.

Постановка задачи.

Вариант 5.

#1 Создайте класс «Банк», который имеет атрибуты суммы денег и процентной ставки.

Добавьте методы для вычисления процентных начислений и снятия денег.

Текст программы:

```

import pickle
class Bank():
    def __init__(self, amount, interest_rate):
        self.amount = amount
        self.interest_rate = interest_rate
    def calculate_interest(self):
        self.amount += (self.amount * self.interest_rate) / 100
    return self.amount
    def withdraw(self, amount):
        if amount <= self.amount:
            self.amount -= amount
        else:
            print('Недостаточно средств на балансе')

# Использование функций
bank = Bank(100, 5)
rate = bank.calculate_interest()
print("Прибыль от процентов", rate)
bank.withdraw(20)
print('Сумма счета после снятия денег',
bank.amount)
def save_def(bank_list):
    with open('bank_data.pkl', 'wb') as f:
        pickle.dump(bank_list, f)
def load_def():
    try:
        with open('bank_data.pkl', 'rb') as f:
            bank_list = pickle.load(f)
            return bank_list
    except FileNotFoundError:
        return None

# Пример использования
bank1 = Bank(1000, 5)
bank2 = Bank(2000, 10)
bank3 = Bank(3000, 15)

bank_list = [bank1, bank2, bank3]

# Сохранение экземпляров класса в файл
save_def(bank_list)

# Загрузка экземпляров класса из файла
loaded_bank_list = load_def()

# Проверка загруженных экземпляров класса
for bank in loaded_bank_list:
    print(bank.amount, bank.interest_rate)

```

Протокол работы программы:

Прибыль от процентов 105.0

Сумма счета после снятия денег 85.0

1000 5

2000 10

3000 15

Process finished with exit code 0

#2 Создайте класс "Фрукт", который содержит информацию о наименовании и весе фрукта. Создайте классы "Яблоко" и "Апельсин", которые наследуются от класса "Фрукт" и содержат информацию о цвет

```
class Fruit:
    def __init__(self, name, weight):
        self.name = name
        self.weight = weight

class Apple(Fruit):
    def __init__(self, name, weight, color):
        super().__init__(name, weight)
        self.color = color

class Orange(Fruit):
    def __init__(self, name, weight, color):
        super().__init__(name, weight)
        self.color = color

apple = Apple("Hasky", 150, "зеленый")
orange = Orange("toyota camry 200", 200, "оранжевый")
print(apple.name, apple.weight, apple.color)
print(oranged.name, orange.weight, orange.color)
```

Протокол работы программы:

Hasky 150 зеленый

toyota camry 200 200 оранжевый

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ для работы с классами в IDE PyCharm Community. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.