

Санкт-Петербургский Национальный  
Исследовательский Университет Информационных  
технологий, механики и оптики

**Контрольная работа №1**  
**ООП**

Выполнил: Фоминцев  
Денис Русланович  
Группа № 3123  
Проверила: Казанова  
Полина Петровна

Санкт-Петербург  
2021

## Цель работы:

Требуется разработать программы согласно указанным ниже заданиям. Реализация классов должна быть в отдельных модулях. Клиентский код должен импортировать вашу библиотеку и посредством объектов воспользоваться реализуемой в них функциональностью. В тексте использовать комментарии, поясняющие ваши действия и принятые решения. Объем задания (конкретные указания, что надо выполнить) указан ниже. Программное решение общего задания требуется представить в виде отдельных решений по каждому заданию, которые должны показать развитие общего проекта. Сохранить проект для дальнейшего использования!

## Задачи:

### Постановка задачи

#### Задание 1. Базовое решение

- Реализовать класс с конструктором. Определить значения по умолчанию для некоторых аргументов, чтобы их можно было не указывать в тех случаях, когда какие-то определенные значения недоступны или бессмысленны.
- Реализовать с помощью свойств инкапсуляцию.
- Добавить в класс методы, определяющие поведение.
- Реализовать производные классы (не менее двух) и методы, характерные для них.
- Реализовать отношение композиции.
- Реализовать для базового класса перегрузку двух любых стандартных операторов, например, сложения и вычитания. Методы перегрузки должны возвращать новый объект того же класса.

#### Задание 2. Множественное наследование

- Реализовать некую сущность вашего решения (обосновать свой выбор) как абстрактный класс.
- Изменить реализацию наследования, добавив наследование от абстрактного класса.
- Применить в иерархии классов наследование от класса-миксина.
- Продумать и по возможности реализовать множественное наследование (в случае отказа от реализации множественного наследования обосновать свое решение).

#### Задание 3. Расширение функциональности

- Реализуйте в своем проекте декораторы, которые
  1. регистрирует вызовы важных сценариев. Результат декорирования – информация о выполнении сценария (например, факт отработки, время начала и окончания, длительность и т.д.).
  2. Устанавливает/проверяет пре- и постусловия функций.
  3. что-либо на ваше усмотрение.
- Реализуйте в своем проекте метаклассы для расширения классов дополнительными возможностями (на ваше усмотрение).

#### Задание. Банковские вклады

Банк предлагает ряд вкладов для физических лиц:

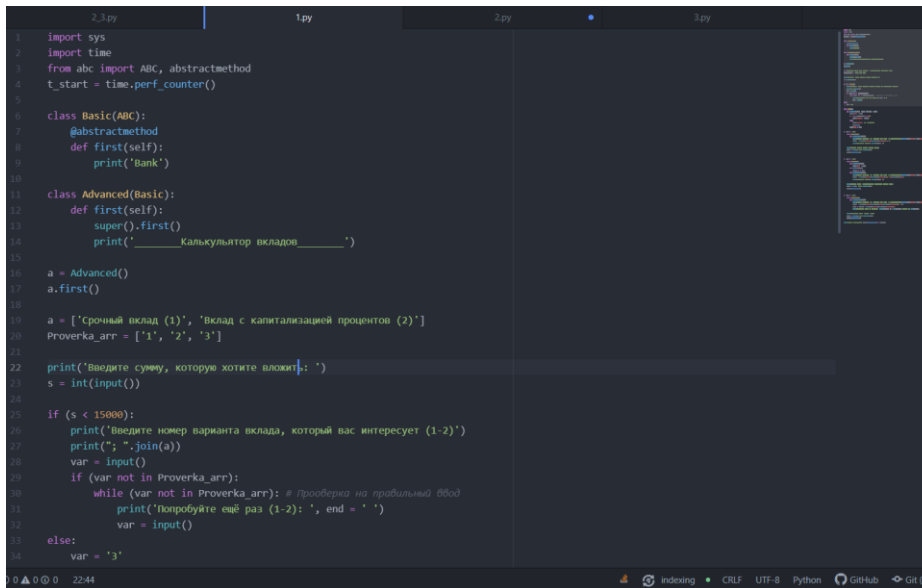
- Срочный вклад: расчет прибыли осуществляется по формуле простых процентов;
- Бонусный вклад: бонус начисляется в конце периода как % от прибыли, если вклад больше определенной суммы;
- Вклад с капитализацией процентов.

Требуется реализовать приложение, которое бы позволило подобрать клиенту вклад по заданным параметрам.

## Ход работы:

### Задание №1

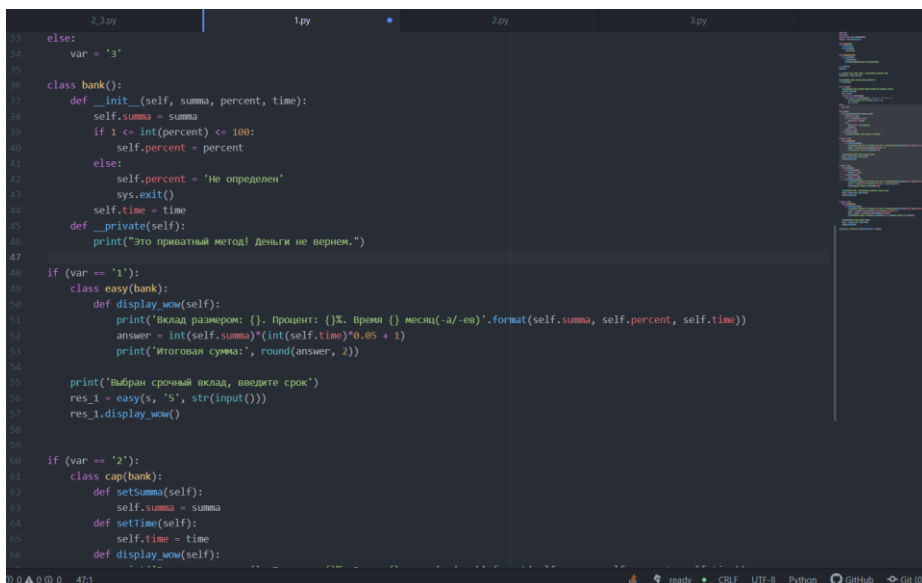
В самом начале данного кода создается абстрактный класс и его наследник для представления пользователю основной информации о работе программы. (class Basic(ABC) и class Advanced(Basic)) соответственно. Ввод основной суммы и ее проверка производится через переменную s и цикл if (см. рисунок 1 – 25 строчка)). Если сумма выше 15000, то автоматически выбирается бонусный вклад, в котором начисляется к основным процентам бонус. В противном случае пользователю представляется возможность выбора между остальными вкладами



```
1 import sys
2 import time
3 from abc import ABC, abstractmethod
4 t_start = time.perf_counter()
5
6 class Basic(ABC):
7     @abstractmethod
8     def first(self):
9         print('Bank')
10
11 class Advanced(Basic):
12     def first(self):
13         super().first()
14         print('_____Калькулятор вкладов_____')
15
16 a = Advanced()
17 a.first()
18
19 a = ['Срочный вклад (1)', 'Вклад с капитализацией процентов (2)']
20 Proverka_arr = ['1', '2', '3']
21
22 print('Введите сумму, которую хотите вложить: ')
23 s = int(input())
24
25 if (s < 15000):
26     print('Введите номер варианта вклада, который вас интересует (1-2)')
27     print(";", " ".join(a))
28     var = input()
29     if (var not in Proverka_arr):
30         while (var not in Proverka_arr): # Проверка на правильный ввод
31             print('Попробуйте ещё раз (1-2): ', end = ' ')
32             var = input()
33     else:
34         var = '3'
```

Рисунок 1. Приветствие. Ввод суммы

В основном классе указываем сумму, проценты и время. Также прописываем приватный метод с двумя нижними подчеркиваниями (см. рисунок 2). В трех зависимых классах (easy(), cap(), bonus()) пишем основной принцип работы – считаем проценты (см. рисунок 2 и рисунок 3). Через метод display\_wow() осуществляется вывод данных



```
33 else:
34     var = '3'
35
36 class bank():
37     def __init__(self, summa, percent, time):
38         self.summa = summa
39         if 1 <= int(percent) <= 100:
40             self.percent = percent
41         else:
42             self.percent = 'Не определен'
43             sys.exit()
44         self.time = time
45     def __private(self):
46         print("Это приватный метод! Деньги не вернем.")
47
48 if (var == '1'):
49     class easy(bank):
50         def display_wow(self):
51             print('Вклад размером: {}. Процент: {}%. Время {} месяц(-а/-ев)'.format(self.summa, self.percent, self.time))
52             answer = int(self.summa)*(int(self.time)*0.05 + 1)
53             print('Итоговая сумма:', round(answer, 2))
54
55             print('Выбран срочный вклад, введите срок')
56             res_1 = easy(s, '5', str(input()))
57             res_1.display_wow()
58
59 if (var == '2'):
60     class cap(bank):
61         def setSumma(self):
62             self.summa = summa
63         def setTime(self):
64             self.time = time
65         def display_wow(self):
```

Рисунок 2. Основной Класс и его наследники

```

4 res_1 = easy(s, '5', str(input()))
5 res_1.display_wow()
6
7
8 if (var == '2'):
9     class cap(bank):
10         def setSumma(self):
11             self.summa = summa
12         def setTime(self):
13             self.time = time
14         def display_wow(self):
15             print('Вклад размером: {}. Процент: {}%. Время {} месяц(-а/-ев)'.format(self.summa, self.percent, self.time))
16             answer = int(self.summa)*(int(self.percent)/100 + 1)**int(self.time)
17             print('Итоговая сумма:', round(answer, 2))
18
19     print('Выбран вклад с капитализацией процентов, введите срок')
20     res_1 = cap(s, '12', str(input()))
21     res_1.display_wow()
22
23
24 if (var == '3'):
25     class bonus(bank):
26         def display_wow(self):
27             print('Вклад размером: {}. Процент: {}%. Время {} месяц(-а/-ев)'.format(self.summa, self.percent, self.time))
28             answer = int(self.summa)*(int(self.time)*int(self.percent)/100 + 1)
29             bonus = (answer - int(self.summa))*int(self.percent)/100
30             print('Итоговая сумма (+ бонус):', round(answer, 2) + int(bonus), 'Бонус: ', int(bonus))
31
32     print('Бонусный вклад. Введите срок')
33     res_1 = bonus(s, '7', str(input()))
34     res_1.display_wow()
35
36     print('Время окончания:', time.perf_counter() - t_start)
37

```

Рисунок 3. Наследники класса Bank() и вывод данных

Результат работы программы:

- ```

Bank
    Калькулятор вкладов
Введите сумму, которую хотите вложить:
15000
Бонусный вклад. Введите срок
45
Вклад размером: 15000. Процент: 7%. Время 45 месяц(-а/-ев)
Итоговая сумма (+ бонус): 65557.0 Бонус: 3307
Время окончания: 8.3865821

```
- ```

Bank
    Калькулятор вкладов
Введите сумму, которую хотите вложить:
10000
Введите номер варианта вклада, который вас интересует (1-2)
Срочный вклад (1); Вклад с капитализацией процентов (2)
1
Выбран срочный вклад, введите срок
6
Вклад размером: 10000. Процент: 5%. Время 6 месяц(-а/-ев)
Итоговая сумма: 13000.0
Время окончания: 9.6338869

```
- ```

Bank
    Калькулятор вкладов
Введите сумму, которую хотите вложить:
1000
Введите номер варианта вклада, который вас интересует (1-2)
Срочный вклад (1); Вклад с капитализацией процентов (2)
2
Попробуйте ещё раз (1-2): awdaw
Попробуйте ещё раз (1-2): dadwa
Попробуйте ещё раз (1-2): 0
Попробуйте ещё раз (1-2): 2
Выбран вклад с капитализацией процентов, введите срок
7
Вклад размером: 1000. Процент: 12%. Время 7 месяц(-а/-ев)
Итоговая сумма: 2210.68
Время окончания: 14.2542311

```

## Вывод:

В данной работе создавали программу по заданным параметрам, используя основные принципы работы ООП.