



**Министерство науки и высшего образования Российской Федерации**  
**Федеральное государственное бюджетное образовательное**  
**учреждение высшего образования**  
**«Московский государственный технический университет**  
**имени Н.Э. Баумана**  
**(национальный исследовательский университет)»**  
**(МГТУ им. Н.Э. Баумана)**  
**Факультет «Информатика и системы управления»**  
**Кафедра «Системы обработки информации и управления»**

### **Лабораторная работа 4**

**по дисциплине «Базовые компоненты интернет-технологий»**

**«Шаблоны проектирования и модульное тестирование в  
Python»**

**Выполнил:**

**Студент группы ИУ5-35Б**

**Титов Е.А.**

**2021 г.**

**Постановка задачи**

### Цель лабораторной работы:

изучение реализации шаблонов проектирования и возможностей модульного тестирования в языке Python

### Задание:

1. Необходимо для произвольной предметной области реализовать от одного до трех шаблонов проектирования: один порождающий, один структурный и один поведенческий. В качестве справочника шаблонов можно использовать следующий каталог. Для сдачи лабораторной работы в минимальном варианте достаточно реализовать один паттерн.
2. Вместо реализации паттерна Вы можете написать тесты для своей программы решения биквадратного уравнения. В этом случае, возможно, Вам потребуется доработать программу решения биквадратного уравнения, чтобы она была пригодна для модульного тестирования.
3. В модульных тестах необходимо применить следующие технологии:
  - ⑩ TDD - фреймворк.
  - ⑩ BDD - фреймворк.
  - ⑩ Создание Моск-объектов.

### Текст программы Файл main.py

```
import numpy as np

class Data:

    __instance = None

    def __new__(cls, *args, **kwargs):
        if cls.__instance is None:
            cls.__instance = super().__new__(cls)
        return cls.__instance

    def __del__(self):
        Data.__instance = None # удаляем единственный объект класса Data

    def __init__(self, data_stamp: tuple):
        """data - кортеж из нумпаевских многомерных массивов X и y"""
        self.X, self.y = data_stamp

    def transform(self):
```

```

means = self.X.mean(axis=0)
stds = self.X.std(axis=0)
return (self.X - means) / stds

def train_test_split(self, train_size=0.7, random_state=42) -> tuple:
    np.random.seed(random_state)
    size = int(train_size*self.X.shape[0])
    mask = np.random.choice(range(self.X.shape[0]), size=size)
    return self.X[mask], self.y[mask]

data = np.array([[1,2,3],[3,4,5]])
d1 = Data(data)
d2 = Data((data[0]+1, data[1]+1))
print(d1 is d2)
# print(id(d1), id(d2))
# print(d1.transform())

```

## Результат выполнения программы

```

(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Установите последнюю версию PowerShell для новых функций и улучшения! https://aka.ms/PSWindows

PS C:\Users\egork> & C:/Users/egork/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/egork/O
True
PS C:\Users\egork>

```