**a. Цель работы.**Ознакомиться с методами тест-дизайна, выделение эквивалентных классов для одной из форм приложения, расчет минимального количества тестов для проверки формы с учетом эквивалентных классов и минимизации количества тестов.

**Выделение эквивалентных классов:**

Рассмотрим поля для ввода данных:  
- ввод даты

- ввод города

1. Поле «Дата»  
 Эквивалентный класс 1: Валидная дата формата yyyy-mm-dd  
 Эквивалентный класс 2: Невалидная дата формата dd-mm-yyyy, пустая строка или другой вариант ввода даты

2. Поле «Город»  
 Эквивалентный класс 1: Валидный ввод города «London»  
 Эквивалентный класс 2: Невалидный ввод города – пустая строка или неверно введенный город

**Расчет количества тестов:**Для сокращения проводимых тестов можно воспользоваться методикой попарного тестирования. Таким образом. Мы сократим время для создания тестов и тест-кейсов.

**Эквивалентные классы:**

1. Город:
   * E1.1: Валидный город
   * E1.2: Невалидный город
2. Дата:
   * E2.1: Валидная дата
   * E2.2: Невалидная дата

Минимальное количество тестов для попарного тестирования: 4

**Тестовые комбинации:**

1. (E1.1, E2.1) - Валидный город, валидная дата
2. (E1.1, E2.2) - Валидный город, невалидная дата
3. (E1.2, E2.1) - Невалидный тикер, валидная дата
4. (E1.2, E2.2) - Невалидный тикер, невалидная дата

**b. Список используемых тест-кейсов.**

1. Тест-кейс 1 - Валидный город, валидная дата
2. Тест-кейс 2 - Валидный город, невалидная дата
3. Тест-кейс 3 - Невалидный тикер, валидная дата
4. Тест-кейс 4 - Невалидный тикер, невалидная дата

**c. Описание эквивалентных классов.**

- Город:

Валидный город: London, Moscow

Невалидный город: пустая строка, неправильно написанный город.

• Дата:

Валидная дата: "2023-01-01".

Невалидная дата: пустая строка, неверный формат.

**d. Расчет количества тестов.**

Благодаря методу попарного тестирования было сделано минимальное количество необходимых тестов - 4 теста

**e. Выводы по работе.**

Анализ проекта позволил определить эквивалентные классы для полей формы ввода данных, а также рассчитать минимальное количество тестов для их проверки. Попарное тестирование позволяет уменьшить количество тестов, что покрывает все условия.

**f. Список использованных источников.**

1. Документация по библиотеке Telebot - <https://pytba.readthedocs.io/en/latest/>

2. Документация по ConfigData - https://www.gb.nrao.edu/~mwhitehe/dibas/html/configdata.html

3. Документация по pandas - https://pandas.pydata.org/docs/