**Практическая работа 9**

**Тема. Тестирование программного продукта методом «черного ящика»**

**Цель**: Научиться проводить тестирование «черного ящика» разными способами

**Инструктаж на рабочем месте** проводится согласно инструкции по охране труда при работе в лаборатории Технологии разработки БД, Автоматизированного проектирования технологических процессов и программирования №319 ИОТ – 029 2016**.**

**Перечень средств обучения:** персональный компьютер

**1.Предусловия и постусловия**

**Предусловия:**

В файлах Group.txt, Song.txt, Tour.txt могут храниться корректные данные.

Пользователь вводит данные через консоль.

Идентификаторы групп уникальны.

Формат дат — yyyy-MM-dd.

Целочисленные значения (ID, год, позиция в чарте) вводятся корректно.

**Постусловия:**

При успешном вводе данных группа, песня или тур добавляются в соответствующий список.

При ошибочном вводе — выводится сообщение об ошибке без аварийного завершения программы.

При сохранении данные записываются в соответствующие .txt файлы.

**2. Дерево разбиения и классы эквивалентности**

Пример для функции **AddGroup()**:

Вводимые данные:

1. Название (строка)
2. Год (целое число)
3. Страна (строка)
4. Позиция в чарте (целое число)

|  |  |  |  |
| --- | --- | --- | --- |
| **Поле** | **Условие/Диапазон** | **Допустимый класс** | **Недопустимые классы** |
| Год | 1900-2025 | [1900-2025] | <1900,>2025,текст |
| Позиция в чарте | 1-100 | [1-100] | <0,>100,текст |
| Название | Не пустое | Строка ≠ "" | “-” |
| Страна | Не пустая | Строка ≠ "" | “-” |

**3. Тестовые варианты**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Назначение теста** | **Исходные данные** | **Ожидаемый результат** | **Реакция программы** | **Вывод** |
| 1 | Корректное добавление группы | Name=Beatles, Year=1960, Country=UK, Chart=1 | Группа добавлена | “Группа добавлена.” | **+**  Прошла |
| 2 | Ошибка формата года | Name=ABBA, Year=abcd, Country=Sweden, Chart=2 | Ошибка формата | “Неверный формат года.” | **+** Обработка ошибок |
| 3 | Год вне диапазона | Name=Queen, Year=1800, Country=UK, Chart=5 | Ошибка | “Неверный формат года.” | **+** |
| 4 | Пустое название | Name=“”, Year=2000, Country=USA, Chart=10 | Ошибка | “Название не может быть пустым.” | ⚠ Добавить проверку |
| 5 | Позиция вне диапазона | Name=Metallica, Year=1981, Country=USA, Chart=200 | Ошибка | “Неверный формат позиции.” | **+** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Назначение теста** | **Исходные данные** | **Ожидаемый результат** | **Реакция программы** | **Вывод** |
| 6 | Добавление песни существующей группе | GroupID=1, Title=Yesterday, Composer=McCartney, Year=1965 | Песня добавлена | “Песня добавлена.” | **+** |
| 7 | Несуществующий ID группы | GroupID=99 | Ошибка | “Группа не найдена.” | **+** |
| 8 | Некорректный год | Year=abcd | Ошибка | “Неверный формат года.” | **+** |

Пример для **AddSong():**

Пример для **TourDetails():**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Назначение теста** | **Исходные данные** | **Ожидаемый результат** | **Реакция программы** | **Вывод** |
| 9 | Правильные даты | Start=2024-05-01, End=2024-05-10 | 9 дней | “Город: Париж, Длительность: 9 дней” | **+** |
| 10 | Неверный формат даты | Start=01.05.2024 | Ошибка | “Неверный формат даты.” | **+** |

**4. Анализ граничных значений:**

Пример для **ChartPosition:**

* Минимум = 1
* Максимум = 100

|  |  |  |
| --- | --- | --- |
| **Тест** | **Значение** | **Ожидаемый результат** |
| **A1** | **0** | **Ошибка** |
| **A2** | **1** | **Успешно** |
| **A3** | **100** | **Успешно** |
| **A4** | **101** | **Ошибка** |

**5. Вывод**

Метод тестирования **«черного ящика»** позволил:

1. Проверить корректность обработки пользовательского ввода.
2. Обнаружить отсутствие валидации пустых строк.
3. Подтвердить корректную работу при правильных данных.
4. Определить необходимость улучшения сообщений об ошибках.

**Контрольные вопросы**

1. **Что такое тестирование “черного ящика”?**Метод тестирования, при котором внутреннее устройство программы неизвестно тестировщику; тесты строятся по функциональности.
2. **Что такое класс эквивалентности?**Набор входных данных, для которых программа должна вести себя одинаково.
3. **Как формируются классы эквивалентности?**По диапазонам, конкретным значениям, множествам или булевым условиям входных данных.
4. **Что такое анализ граничных значений?**Проверка программы на корректность работы при минимальных, максимальных и предельных значениях входных данных.