|  |
| --- |
| **Московский авиационный институт**  **(национальный исследовательский университет)**  **Институт №3** **—** «Системы управления, информатика и электроэнергетика» |

|  |  |  |  |
| --- | --- | --- | --- |
| УТВЕРЖДАЮ |  | УТВЕРЖДАЮ | |
| к.т.н., доцент каф. 307 |  | Студенты  гр. М3О-412Б-18 | |
| Максимов Н.А.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  личная подпись  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  расшифровка подписи |  | Усенко Д.Е. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  личная подпись  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  расшифровка подписи | |
|  |  | Акимов В. Н.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  личная подпись  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  расшифровка подписи | |
| Печать |  | Печать | |
| Дата |  | Дата | |
| Приложение | | |

наименование вида АС

|  |
| --- |
| Интернет магазин: Покупатели |

наименование объекта автоматизации

|  |
| --- |
| «I-mag» |

сокращенное наименование АС

**Программа и методика испытаний**

На 12 листах

Действует с 20.12.2021

Москва 2021

Содержание

[1. Объект испытаний 3](#_Toc91030516)

[2. Цель испытаний 3](#_Toc91030517)

[3. Требования к программе 3](#_Toc91030518)

[4. Средства и порядок испытаний 3](#_Toc91030519)

[5. Методы испытаний 4](#_Toc91030520)

[6. Источники разработки 12](#_Toc91030521)

# Объект испытаний

Объектом испытаний является система «I-mag».

# Цель испытаний

Испытания проводятся с целью проверки соответствия функциональных требований технического задания и фактической работы программного продукта, а также выявления уязвимостей системы от незапланированных действий пользователя и ошибок в программном коде.

# Требования к программе

В программе должны быть предусмотрен случай отсутствия нужного файла базы данных и создан unittest для класса MyDb, отвечающего за взаимодействие с базой данных. Требования к программной документации

Требования к программной документации не предъявляются.

# Средства и порядок испытаний

Для тестирования модуля программы, работающего с базой данных, был использован инструмент unittest. Это стандартный модуль для написания юнит-тестов на Python.

При тестировании модуля (database.py) создается новый файл с добавлением в названии test в начале или в конце имени файла (test\_database.py).

Так как тестируемый модуль представляет собой класс MyDb и его методы, в модуле с тестами (test\_database.py) необходимо создать класс TestMyDb, который наследует unittest.TestCase и в методе setUp создать объект класса MyDb. Суть метода setUp в том, что он будет вызываться в начале проведения любых тестов. Также есть противоположный ему метод setDown, который будет вызываться в конце проведения тестов.

Далее для методов класса MyDb проводятся тесты. Для этого используются методы assertEqual, который проверяет соответствие возвращаемого методом класса MyDb значение ожидаемому значению, и метод assertRaises, проверяющий, что метод класса MyDb с заданными в тесте аргументами порождает исключение.

Провести испытания для методов класса MyDb:

* select\_all
* get\_buyers
* get\_by\_name
* get\_by\_article
* check\_user\_in\_authorization
* insert\_into\_buyers

# Методы испытаний

* 1. **Испытание при отсутствии требуемой базы данных**

Так как для работы программы требуется база с уже имеющимися данными, необходимо проверять наличие этой базы и в случае её отсутствия поднимать ошибку ConnectionError.

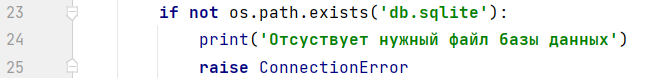


Рисунок . Вызов ошибки ConnectionError в случае отсутствия файла с базой данных.

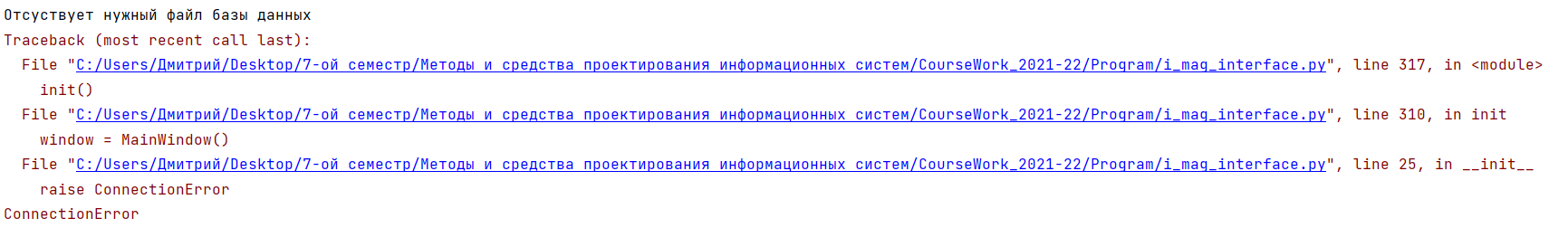


Рисунок . Вывод сообщения и выход из программы в случае отсутствия файла с базой данных.

* 1. **Тестирование метода select\_all**

В классе TestMyDb создан метод test\_select\_all и при помощи метода assertEqual проведена проверка соответствия возвращаемого типа ожидаемому. В случае, если бы возвращаемое значение было не равно ожидаемому – выводилась строка «Полученное значение не является списком!».



Рисунок . Тестирование метода select\_all

* 1. **Тестирование метода get\_buyers**

В классе TestMyDb создан метод test\_get\_buyers и при помощи метода assertEqual проведена проверка соответствия возвращаемого типа ожидаемому. В случае, если бы возвращаемое значение было не равно ожидаемому – выводилась строка «Полученное значение не является списком!».



Рисунок . Тестирование метода get\_buyers

* 1. **Тестирование метода get\_by\_name.**

Предварительно, в методе get\_by\_name класса MyDb создана проверка на то, что передаваемые методу аргументы должны быть строками, в ином случае будет подниматься ошибка TypeError.



Рисунок . Вызов ошибки TypeError при неправильном типе данных

В классе TestMyDb создан метод test\_get\_by\_name и при помощи метода assertRaises проводится проверка порождения исключения в зависимости от входных данных. В первом случае – в методе assertRaises метод get\_by\_name содержит аргументы с неправильным типом данных, во втором случае – с правильным типом данным. При помощи метода assertEqual проведена проверка соответствия возвращаемого типа ожидаемому. В случае, если бы возвращаемое значение было не равно ожидаемому – выводилась строка «Полученное значение не является списком!».

Из результатов теста видно, что программа обрабатывает случаи с вводом данных неправильного типа и поднимает ошибку TypeError.

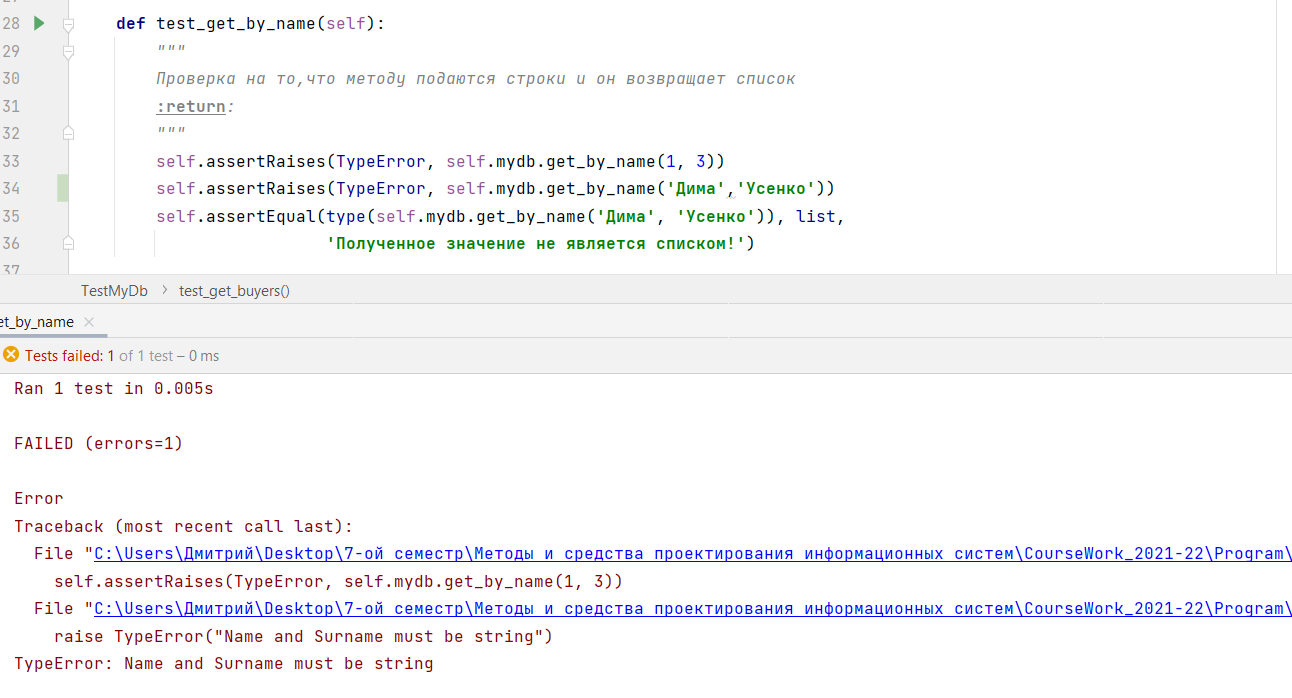


Рисунок . Тестирование метода get\_by\_name

* 1. **Тестирование метода get\_by\_article**

Предварительно, в методе get\_by\_article класса MyDb создана проверка на то, что передаваемый методу аргумент должен быть строкой, в ином случае будет подниматься ошибка TypeError.



Рисунок . Вызов ошибки TypeError при неправильном типе данных

В классе TestMyDb создан метод test\_get\_by\_article и при помощи метода assertRaises проводится проверка порождения исключения в зависимости от входных данных. В первом случае – в методе assertRaises метод get\_by\_article содержит аргумент с неправильным типом данных, во втором случае – с правильным типом данным. При помощи метода assertEqual проведена проверка соответствия возвращаемого типа ожидаемому. В случае, если бы возвращаемое значение было не равно ожидаемому – выводилась строка «Полученное значение не является списком!».

Из результатов теста видно, что программа обрабатывает случаи с вводом данных неправильного типа и поднимает ошибку TypeError.

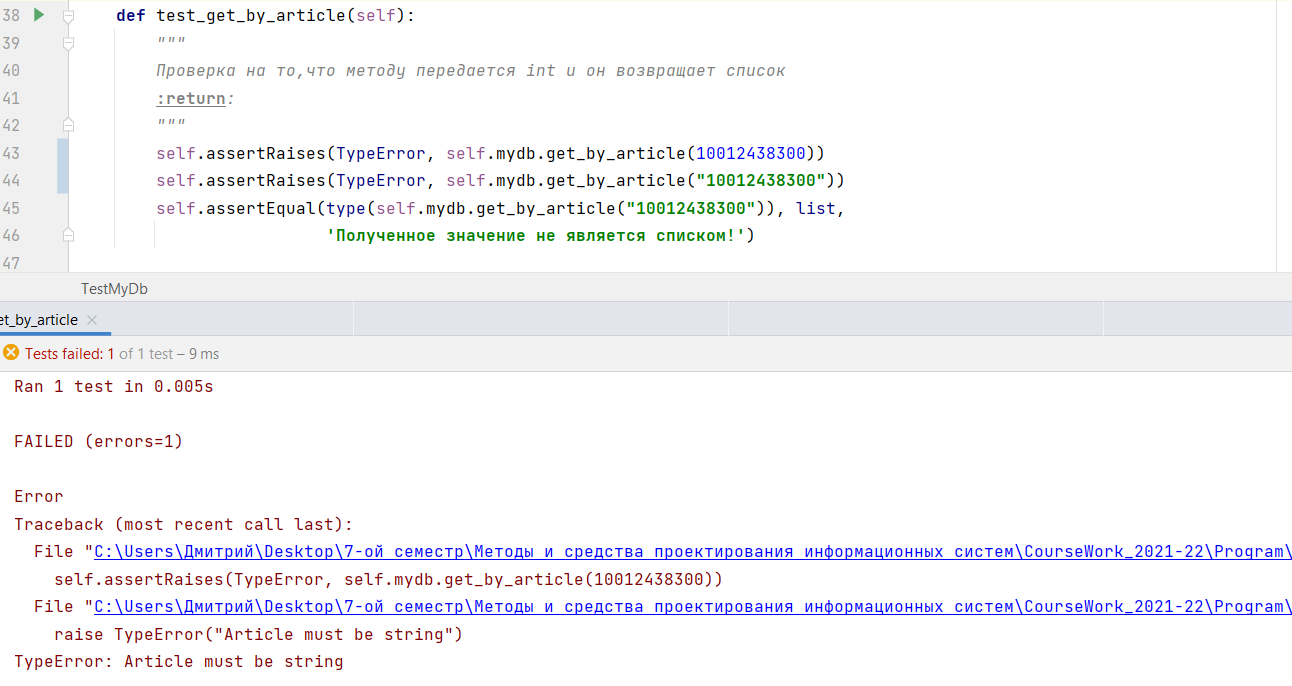


Рисунок . Тестирование метода get\_by\_article

* 1. **Тестирование метода check\_user\_in\_authorization**

Предварительно, в методе check\_user\_in\_authorization класса MyDb создана проверка на то, что передаваемые методу аргументы должны быть строками, в ином случае будет подниматься ошибка TypeError.



Рисунок . Вызов ошибки TypeError при неправильном типе данных.

В классе TestMyDb создан метод test\_check\_user\_in\_authorization и при помощи метода assertRaises проводится проверка порождения исключения в зависимости от входных данных. В первом случае – в методе assertRaises метод check\_user\_in\_authorization содержит аргументы с неправильным типом данных, во втором случае – с правильным типом данным. При помощи метода assertEqual проведена проверка соответствия возвращаемого значения ожидаемому. В случае, если бы возвращаемое значение было не равно ожидаемому – выводилась строка «Возвращается неверное значение».

Из результатов теста видно, что программа обрабатывает случаи с вводом данных неправильного типа и поднимает ошибку TypeError.

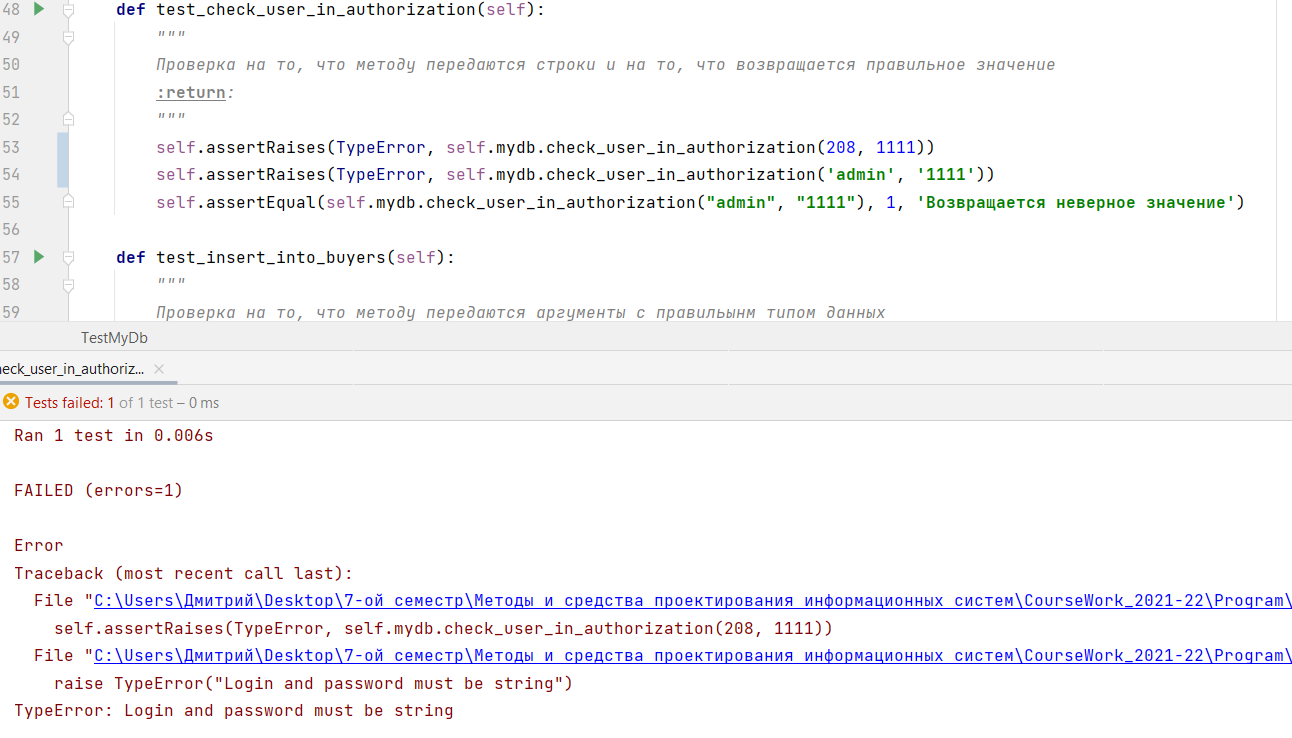


Рисунок 10. Тестирование метода check\_user\_in\_authorization

* 1. **Тестирование метода insert\_into\_buyers**

Предварительно, в методе insert\_into\_buyers класса MyDb создана проверка на то, что передаваемые методу аргументы должны быть строками, в ином случае будет подниматься ошибка TypeError.

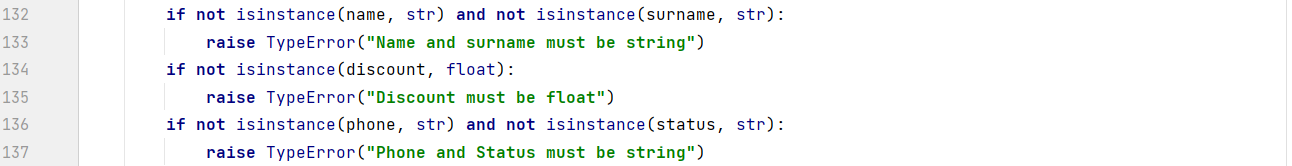


Рисунок . Вызов ошибки TypeError при неправильном типе данных

В классе TestMyDb создан метод test\_insert\_into\_buyers и при помощи метода assertRaises проводится проверка порождения исключения в зависимости от входных данных. В первом случае – в методе assertRaises метод insert\_into\_buyers содержит аргументы с неправильным типом данных, во втором случае – с правильным типом данным.

Из результатов теста видно, что программа обрабатывает случаи с вводом данных неправильного типа и поднимает ошибку TypeError.

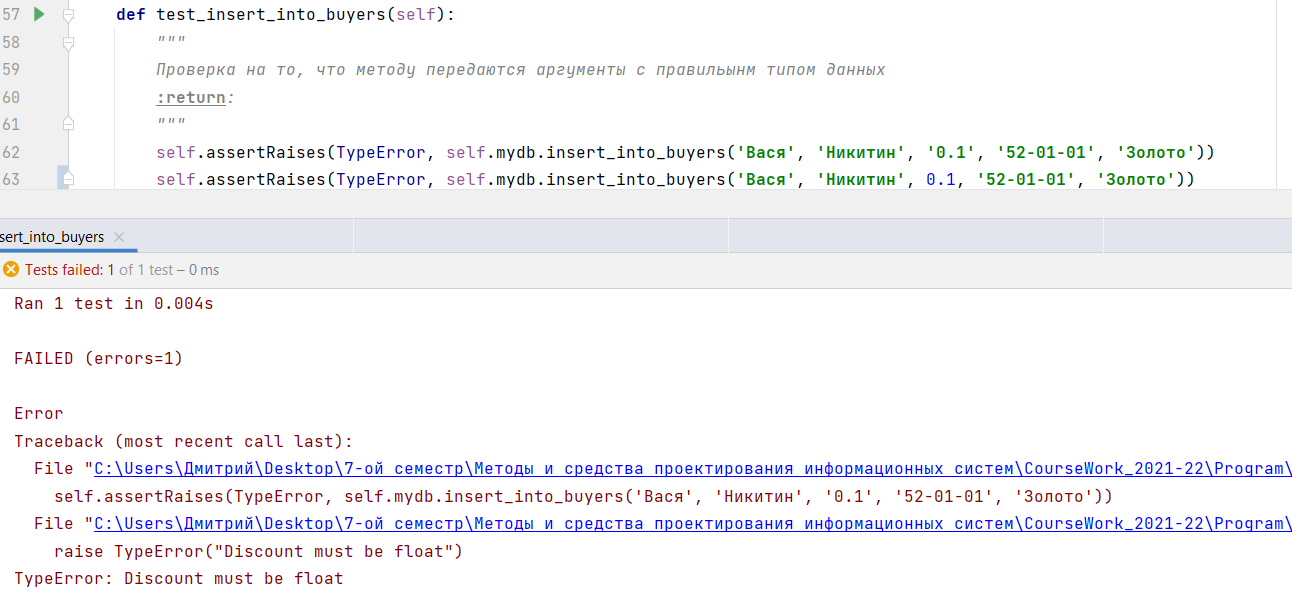


Рисунок . Тестирование метода insert\_into\_buyers

Таблица 1—Таблица соответствия функциональных требований технического задания и фактической работы системы.

|  |  |  |  |
| --- | --- | --- | --- |
| **Пункт** | **Условия корректного выполнения** | **Выполнение** | **Поднятие исключений** |
| Испытание при отсутствии требуемой базы данных | Наличие файла с базой данных и возможность доступа к нему программой | Запуск программы. | Вывод ошибки ConnectionError |
| Тестирование метода select\_all | Наличие соединения с базой данных | Возвращает список во всеми данными таблицы |  |
| Тестирование метода get\_buyers | Наличие соединения с базой данных | Возвращает список со всеми покупателями из базы данных |  |
| Тестирование метода get\_by\_name | Тип передаваемых аргументов должен являться строкой | Возвращает список со всеми покупками конкретного покупателя. | Вывод ошибки TypeError |
| Тестирование метода get\_by\_article | Тип передаваемого аргумента должен являться строкой | Возвращает список со всей историей покупок конкретного товара | Вывод ошибки TypeError |
| Тестирование метода check\_user\_in\_authorization | Тип передаваемых аргументов должен являться строкой | Возвращает 1 в случае наличия пользователя в таблице авторизации и 0 при отсуствии | Вывод ошибки TypeError |
| Тестирование метода insert\_into\_buyers | Тип передаваемых аргументов name, surname, phone, status должен являться строкой. discount – число с плавающей точкой | Запись внесётся в базу данных | Вывод ошибки TypeError |

# Источники разработки

Настоящая «Программа и методика испытаний» разработана на основе следующих документов и информационных материалов:

1. Задание на выполнение курсового проекта по дисциплине «Методы и средства проектирования информационных систем и технологий»;
2. Техническое задание на разработку информационной системы;
3. ГОСТ 2.105-95 «Общие правила к текстовым документам»;
4. ГОСТ 19.301.79 «Программа и методика испытаний».