

Лабораторная работа №5

Тема: «Анализ ошибок и обработка исключений. Обработка и генерация исключений. Модульное тестирование».

Требования к выполнению лабораторной работы №5

1. Изучите теоретическую часть к четвёртой лабораторной работе:
 - а. Теоретическая часть к четвёртой лабораторной работе (notebook_5.ipynb).
 - б. Лекция №5.
2. Создайте новый проект.
3. Запустите примеры из лабораторной работы.
4. Выполните задание согласно вашему варианту:
 - а. Вычислите свой вариант (*согласно формуле ниже*).
Если сделали не свой вариант => работа не засчитывается.
 - б. Отправьте выполненное задание в ОРИОКС (*раздел Домашние задания*).

Формат защиты лабораторных работ:

1. Продемонстрируйте выполненные задания.
2. Ответьте на вопросы по вашему коду.
3. При необходимости выполните дополнительное (*дополнительные*) задание от преподавателя.
4. Ответьте (*устно*) преподавателю на контрольные вопросы.

Список вопросов

1. Генерация исключений.
2. try / except / else
3. Сцепление исключений.
4. Оператор assert.
5. Диспетчер контекстов.
6. Что такое автономный тест?
7. Параметризация тестовых функций.
8. Test fixture.
9. Teardown/Cleanup
10. Autouse
11. Маркировка тестов
12. Маркеры skip и xfail

Задания

№ Варианта = номер_студенческого % 2 + 1

Вариант №1

Доработайте свой программный код из ЛР №4 следующим образом:

1. Необходимо проверять корректность вводимых данных и выводить соответствующие сообщения об ошибках.
2. Добавьте метод, который считывает данные из файла.
3. Добавьте метод, который записывает данные в файл.
4. Добавьте обработку исключений при чтении и записи файла.
5. Добавьте обработку исключений на случай некорректных данных.
6. Напишите модульные тесты для каждого метода из ваших классов. Обязательно используйте фикстуры и параметризованные аргументы функций.

Вариант №2

1. Разработайте класс User. В рамках этого класса создайте статический метод `authorize`, который будет возвращать значение `True`, если вход был успешным, и записывать в файл сведения о попытке входа (логин, пароль, дата и время), если вход не удался.
2. Необходимо проверять корректность вводимых данных и выводить соответствующие сообщения об ошибках.
3. Добавьте метод, который считывает данные из файла.
4. Добавьте обработку исключений при чтении и записи файла.
5. Добавьте обработку исключений на случай некорректных данных.
6. Напишите модульные тесты для каждого метода из ваших классов. Обязательно используйте фикстуры и параметризованные аргументы функций.