**Отчет по самостоятельной работе №**6

**по дисциплине МДК 01.02 “Технология разработки программного обеспечения”.**

Выполнил: студент

группы 319

Бакеев Рамзан Русланович

Дата 18.11.2024

**Цель работы**

Цель данной работы – изучить методы и подходы к обработке исключений, научиться применять конструкции try-catch, обрабатывать и регистрировать исключения для повышения стабильности программного обеспечения.

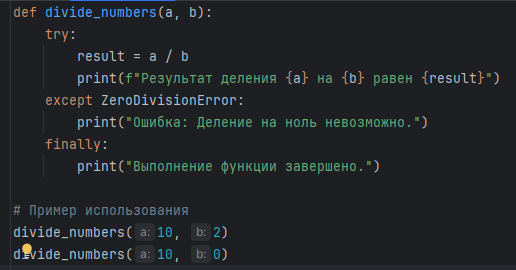
**Основная структура задания**

Задание 1.

1)Создаем простой программный модуль, в котором возникает ошибка (например, деление на ноль).

2)Реализуем обработку исключения с помощью конструкции try-catch, чтобы предотвратить аварийное завершение программы.

3)Выведем сообщение об ошибке и продолжим выполнение программы.

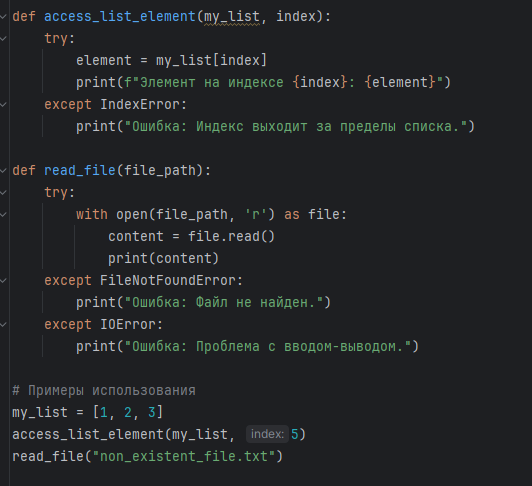


Задание 2.

1) Изучим, как работают различные типы исключений в нашем языке программирования (например, стандартные исключения и их наследники).

2) Создаем несколько примеров обработки различных исключений (например, деление на ноль, выход за границы массива, ошибки ввода-вывода).

3) Задокументируем различия между ними и предложим стратегию обработки каждой категории.



Документация различий:

• ZeroDivisionError: возникает при попытке деления на ноль. Стратегия обработки — вывести сообщение об ошибке и продолжить выполнение.

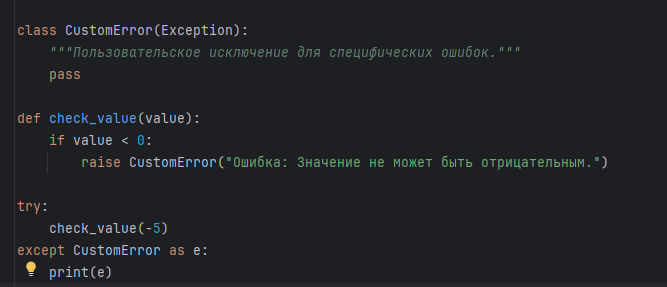
• IndexError: возникает при обращении к элементу списка по несуществующему индексу. Необходимо уведомить пользователя о неверном индексе.

• FileNotFoundError: возникает при попытке открыть файл, который не существует. Рекомендуется уведомить пользователя о том, что файл не найден.

• IOError: возникает при проблемах с вводом-выводом (например, отсутствие прав доступа). Также следует уведомить пользователя.

Задание 3.

1. Создаем пользовательский класс исключений для обработки специфических ситуаций в вашем программном проекте.
2. Напишем пример кода, который генерирует и обрабатывает это исключение.



Обоснование необходимости создания пользовательского исключения:

Создание пользовательских исключений позволяет более точно управлять ошибками в специфических ситуациях, улучшает читаемость кода и упрощает его отладку.

Задание 4

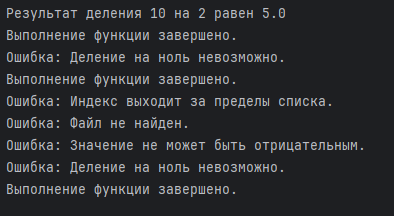
1) Реализуйте механизм логирования исключений в файл или в базу данных для их дальнейшего анализа.

2) Проверьте корректность логирования и убедитесь, что каждая ошибка записывается с указанием времени, типа ошибки и контекста.

3) Задокументируйте подходы к анализу и устранению исключений на основе логов.



Результаты всех предыдущих строк кода



**Выводы работы**

Изучены методы и подходы к обработке исключений, мы научились применять конструкции try-catch, обрабатывать и регистрировать исключения для повышения стабильности программного обеспечения.