|  |
| --- |
| **Автономное учреждение  профессионального образования**  **Ханты-Мансийского автономного округа – Югры**  **«СУРГУТСКИЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»**  **(АУ «Сургутский политехнический колледж»)** |
|  |
| СТРУКТУРНОЕ ПОДРАЗДЕЛЕНИЕ - 4  (Энергетическое отделение) |

**Отчет по самостоятельной работе №7**

**по дисциплине МДК 01.02 “** **Инструментальные средства разработки программного обеспечения”.**

Выполнил: студент

группы 319

Климентов Иван Вячеславович

Дата 28.11.2024

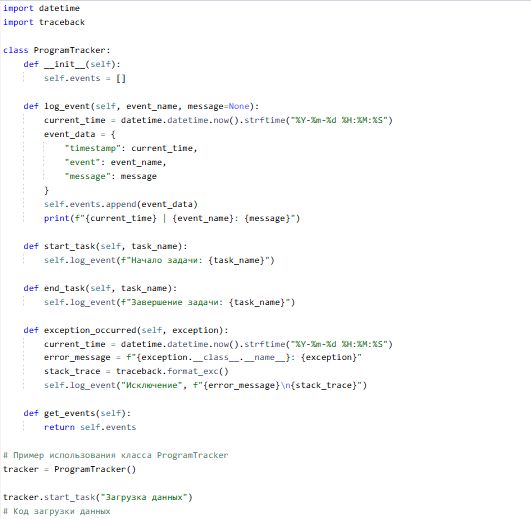
# Цель работы:

Изучить и применить отладочные классы в проекте для улучшения процесса поиска и исправления ошибок в коде.

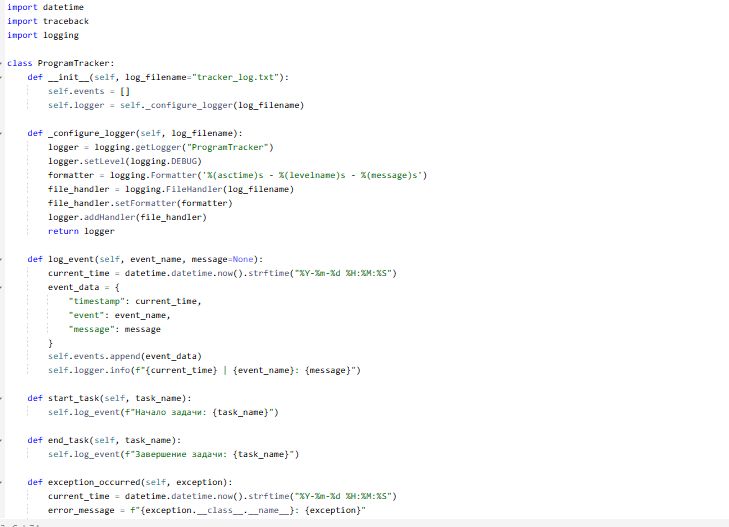
# Основная структура задания:

Задание №1: Создание отладочных классов

1. Был реализован класс для отслеживания выполнения программы.

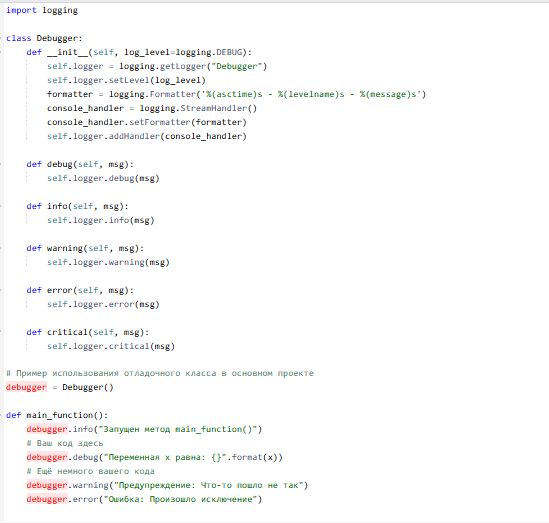


1. Была добавлена возможность логирования исключений в классе.

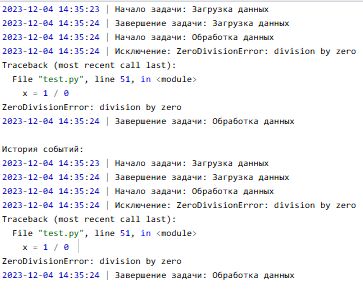


Задание №2: Отладка проекта с использованием отладочных классов

1. Отладочный класс был введён в основной проект.



1. Проведя тестирование, используя функционал класса для отслеживания ошибок в консоли получаем следующую информацию.



Задание №3: Использование отладочного класса для анализа производительности

1. Была использована возможность измерения времени выполнения методов в отладочном классе.



1. Проанализировав производительность одного из модулей проекта, понимаем, что время его выполнения находится в пределах нормы.

444444444444444444444444444444

Задание №4: Интеграция отладочного класса с логированием

1. Была добавлена возможность автоматического сохранения логов в файл для дальнейшего анализа.



1. Проверяем корректность логирования в проекте, намерено создав критическую ситуацию.

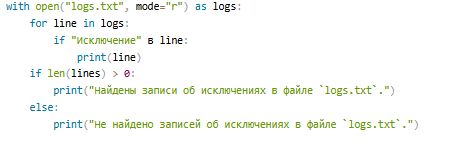


Задание №5: Отладка многопоточной программы с помощью отладочного класса

1. Был применён отладочный класс для отслеживания ошибок в многопоточной программе.



1. Проверяем, открыв файл логов, корректно ли отслеживаются ошибки.



**Вывод:**

В ходе практической работы были изучены и применены отладочные классы для улучшения процесса поиска и исправления ошибок в коде. Был создан класс Debugger, который позволяет отслеживать выполнение программы, логировать исключения, измерять время выполнения методов и сохранять логи в файл. Этот класс был интегрирован в основной проект, что позволило эффективно отслеживать ошибки и анализировать производительность модулей. Также была проведена отладка многопоточной программы с использованием отладочного класса, что доказало его эффективность в различных сценариях использования.