Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г.В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ**

специальность 09.02.07 «Информационные системы и программирование»

Квалификация: Программист

ПРАКТИЧЕСКИЕ РАБОТЫ

ПО МДК 01.03 «Разработка мобильных приложений»

|  |  |
| --- | --- |
| Выполнил студент  группы П50-3-22  Савкова Кира Дмитриевна | Проверил преподаватель  \_\_\_\_\_\_\_\_\_\_\_ К.С. Образцова  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 года |

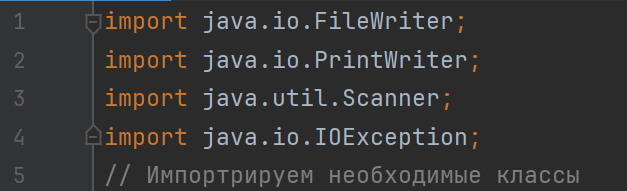
Москва 2024

# ПРАКТИЧЕСКАЯ РАБОТА №1

Цель работы: Создать программу на Java, которая будет запрашивать у пользователя информацию о студентах, такую как их имя, возраст и средний балл. Используя механизмы ввода данных в Java (например, класс Scanner), получить от пользователя данную информацию для нескольких студентов. Записать информацию о каждом студенте в отдельные строки в текстовый файл "students.txt".  Каждая строка должна содержать данные одного студента и быть отделена переносом строки. Для проверки успешной записи данных в файл, после завершения записи откройте файл "students.txt" и убедитесь, что данные сохранились корректно. В программе должны быть комментарии, поясняющие логику работы кода.

Ход работы:

1. Импорт библиотек: В начале программы импортируются необходимые классы из стандартных библиотек Java, такие как `java.io.FileWriter`, `java.io.PrintWriter`, `java.util.Scanner` и `java.io.IOException`.



*Рисунок 1 - Импортирование бибилиотек*

2. Метод `main`: Это основной метод программы, который запускается при её выполнении. Он содержит основную логику программы.

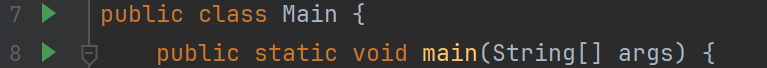


Рисунок 2 - Метод Main

1. Создание объекта Scanner: Создается объект `Scanner` для чтения данных с клавиатуры.



Рисунок 3 - Создание объекта Scanner

1. Блок try-catch-finally: Используется для обработки возможных исключений, которые могут возникнуть во время выполнения программы. В блоке `try` выполняется код, который может вызвать исключение. В случае его возникновения, оно обрабатывается в блоке `catch`. Блок `finally` выполняет код независимо от того, произошло ли исключение или нет.

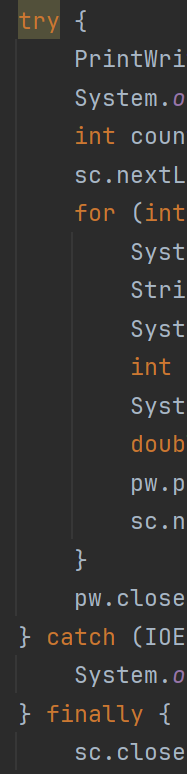


Рисунок 4 - try-catch-finally

1. Создание объекта PrintWriter: Создается объект `PrintWriter`, который будет использоваться для записи данных в файл "students.txt". Параметр `true` в конструкторе `FileWriter` указывает, что запись будет происходить в конец файла, не перезаписывая его содержимое.



Рисунок 5 - Создание объекта PrintWriter

5. Ввод информации о студентах: Пользователю предлагается ввести количество студентов, а затем информацию о каждом студенте: имя, возраст и средний балл. Введенные данные записываются в файл "students.txt" с помощью метода `printf()` объекта `PrintWriter`.

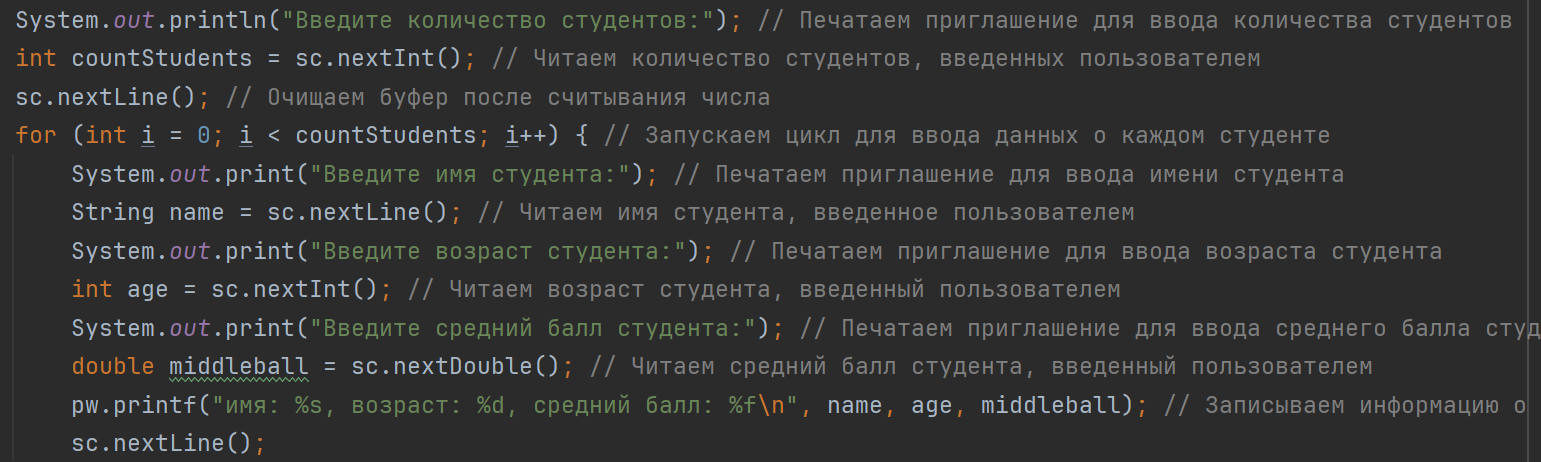


Рисунок 6 - Основной блок кода

6. Закрытие PrintWriter и Scanner: После завершения записи данных в файл объект `PrintWriter` закрывается для освобождения ресурсов. Объект `Scanner` также закрывается в блоке `finally`.

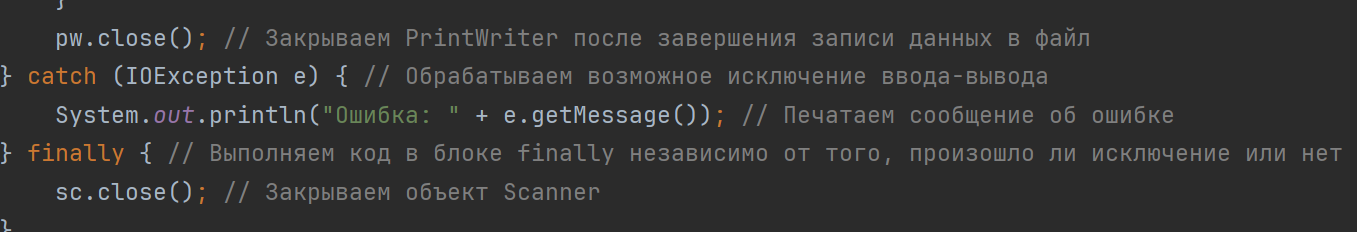


Рисунок 7 - Закрытие объектов

7. Обработка исключения IOException: Если возникает исключение ввода-вывода при работе с файлом, программа выводит сообщение об ошибке.

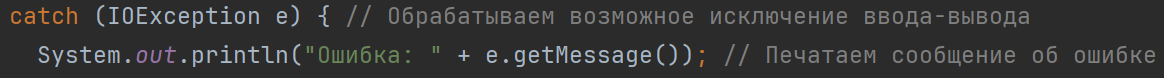


Рисунок 8 - Обработка исключения

Вывод: В результате выполнения данной практической работы была реализована программа на языке Java, которая позволяет пользователю ввести информацию о студентах (имя, возраст, средний балл) с клавиатуры и сохранить эту информацию в файл"students.txt".