

Домашнее задание по теме «Иерархия исключений в Java. Работа программиста с исключениями»

Формулировка задания:

1. Напишите приложение, которое будет запрашивать у пользователя следующие данные в произвольном порядке, разделенные пробелом:

Фамилия Имя Отчество датарождения номертелефона пол возраст

Форматы данных:

- фамилия, имя, отчество — строка;
- датарождения — строка формата dd.mm.yyyy;
- номертелефона - целое беззнаковое число без форматирования;
- пол - символ латиницей f или m;
- возраст — целое число.

Данные читаются из консоли.

2. Приложение должно проверить введенные данные. Если количество полей не совпадает с требуемым, вернуть код ошибки, обработать его и показать пользователю сообщение, что в файле меньше и больше данных, чем требуется.

3. Приложение должно попытаться распарсить полученные значения и выделить из них требуемые параметры. Параметры записываются в поля класса Person. Если форматы данных не совпадают, нужно бросить исключение, соответствующее типу проблемы.

Можно использовать встроенные типы java и создать свои. Исключение должно быть корректно обработано, пользователю выведено сообщение с информацией, что именно неверно.

4. Если всё введено и обработано верно, должен создаваться файл с названием, равным фамилии, в него в одну строку должны записаться полученные данные, вида

<Фамилия><Имя><Отчество><датарождения> <номертелефона><пол>

5. Дополнительно. Однофамильцы должны записаться в один и тот же файл, в отдельные строки.

6. Не забудьте закрыть соединение с файлом. При возникновении проблемы с чтением-записью в файл, исключение должно быть корректно обработано, пользователь должен увидеть стектрейс ошибки.

7. Программа реализуется в отдельной ветке `git homeworks/homework012`. При сохранении состояния программы (коммиты) пишется сообщение с описанием хода работы по задаче.

В корне папки с программой должен быть файл `.gitignore`.

Дополнительная задача. Проверка логина и пароля

1. Создать класс `User` для получения данных о пользователе.

2. Создать статический метод который принимает на вход три параметра: `login`, `password` и `confirmPassword`. Требования к методу:

- `Login` должен содержать только латинские буквы, цифры и знак подчеркивания.
- Длина `login` должна быть меньше 20 символов. Если `login` не соответствует этим требованиям, необходимо выбросить `WrongLoginException`.
- `Password` должен содержать только латинские буквы, цифры и знак подчеркивания. Длина `password` должна быть меньше 20 символов. Также `password` и `confirmPassword` должны быть равны. Если `password` не соответствует этим требованиям, необходимо выбросить `WrongPasswordException`.
- `WrongPasswordException` и `WrongLoginException` - пользовательские классы исключения с двумя конструкторами – один по умолчанию, второй принимает сообщение исключения и передает его в конструктор класса `Exception`.
- Обработка исключений проводится внутри метода.
- Метод возвращает `true`, если значения верны или `false` в другом случае.

3. Программа реализуется в отдельной ветке `git`

homeworks/homework012Addition. При сохранении состояния программы (коммиты) пишется сообщение с описанием хода работы по задаче.

В корне папки с программой должен быть файл .gitignore.

Планируемый результат:

1. Ссылка на программу в репозитории GitHub;
2. Отчёт со скринами выполнения задач - постановка задачи, код задачи и результат в консоли IntelliJ Idea.

Описания плана работы:

Работа с исключениями в программе на языке Java.

Перечень инструментов, необходимых для реализации деятельности:

Персональный компьютер, JDK 17 (либо OpenJDK 17), IntelliJ Idea для разработки на Java, GIT, Tortoise GIT.