**Short url service** **(Sus)** – сервис создания короткий ссылок.

Реализовано на Java + Spring, сборка на Maven.

**Обладает следующим функционалом:**

1. Зарегистрироваться (получить UUID)

2. Выйти из аккаунта

3. Авторизоваться (по UUID)

4. Создать короткую ссылку

5. Перейти по короткой ссылке

6. Изменить количество переходов по короткой ссылке

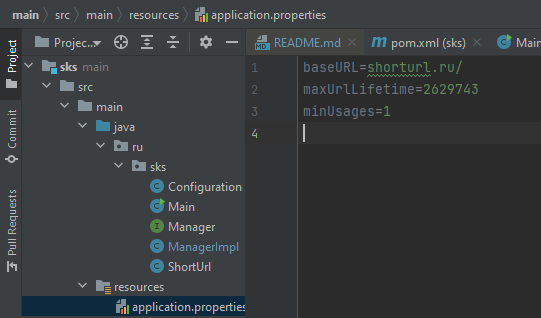
7. Удалить короткую ссылку

8. Изменить время валидности короткой ссылки

9. Получить UUID текущего пользователя

10. Список актуальных коротких ссылок текущего пользователя

**Конфигурация вынесена в файл application.properties**

****

baseUrl – основа короткой ссылки, имитирует наш http-server.

maxUrlLifetime – максимальное время жизни короткой ссылки, дефолтное значение равно месяцу в секундах.

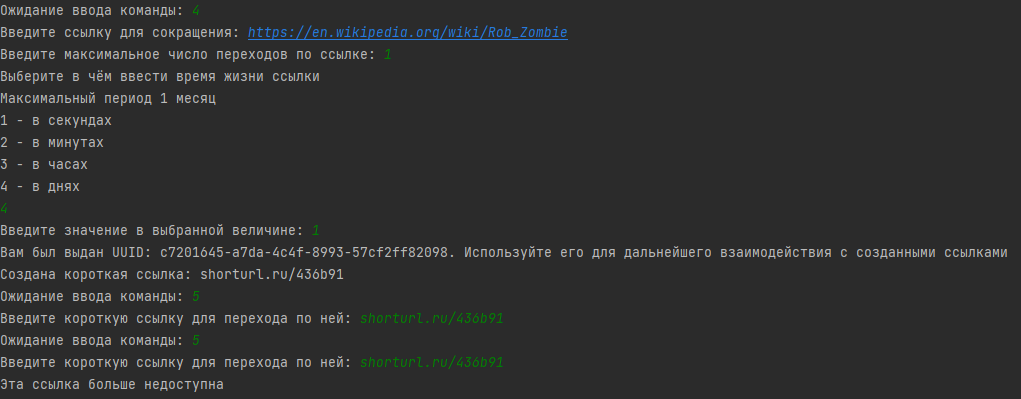
minUsages – минимальное число переходов по ссылке, дефолтное значение системы равно 1.

**Ограничение по времени жизни**

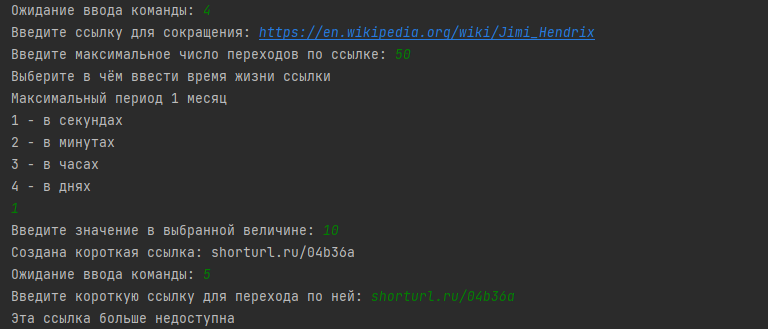
При создании короткой ссылки предлагается выбрать время её жизни в секундах, минутах, часах или днях. Выбранный период не может превышать месяц. Ввод значений больше месяца приводит к выставлению лимита равного месяцу, согласно конфигурации из application.properties.

**Автоматическое удаление ссылок и лимит переходов**

Ссылка будет удалена в “ленивом” режиме в двух случаях. Первый – произошел запрос к ссылке, лимит использований которой достиг 0, такой запрос будет отклонён, а сущность ссылки удалена. Второй – произошел запрос к ссылке, время жизни которой истекло, такой запрос будет отклонён, а сущность ссылки удалена.



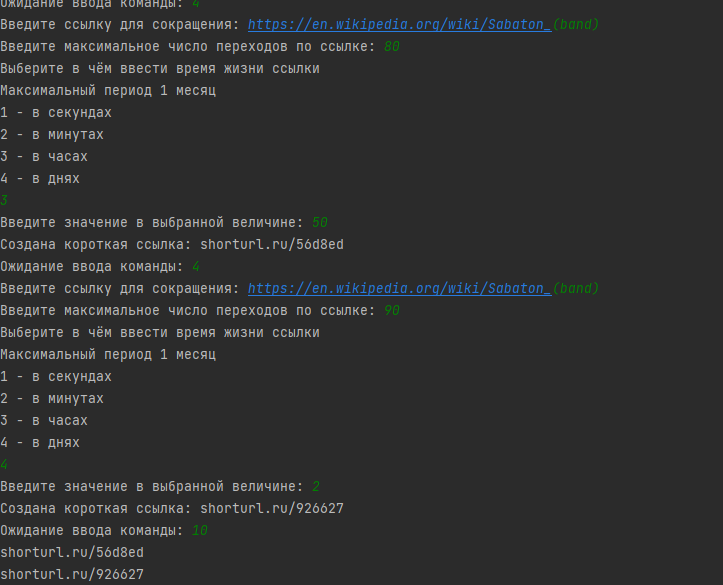
После 1 запроса ссылка стала недоступной.



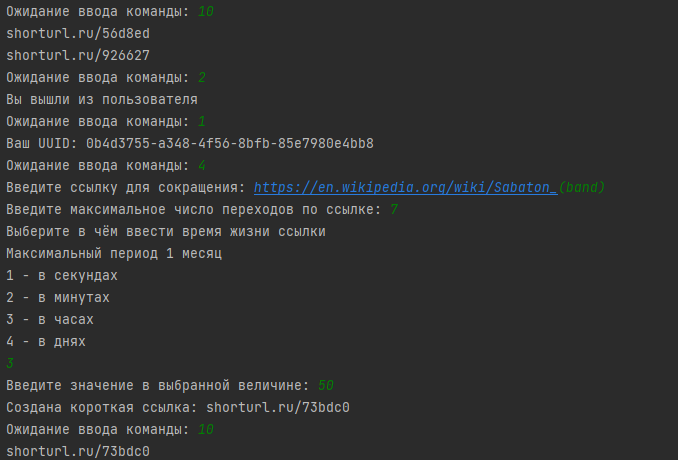
10 секунд вышло, и ссылка перестала быть доступной.

**Уникальность ссылок и повторное создание ссылок**

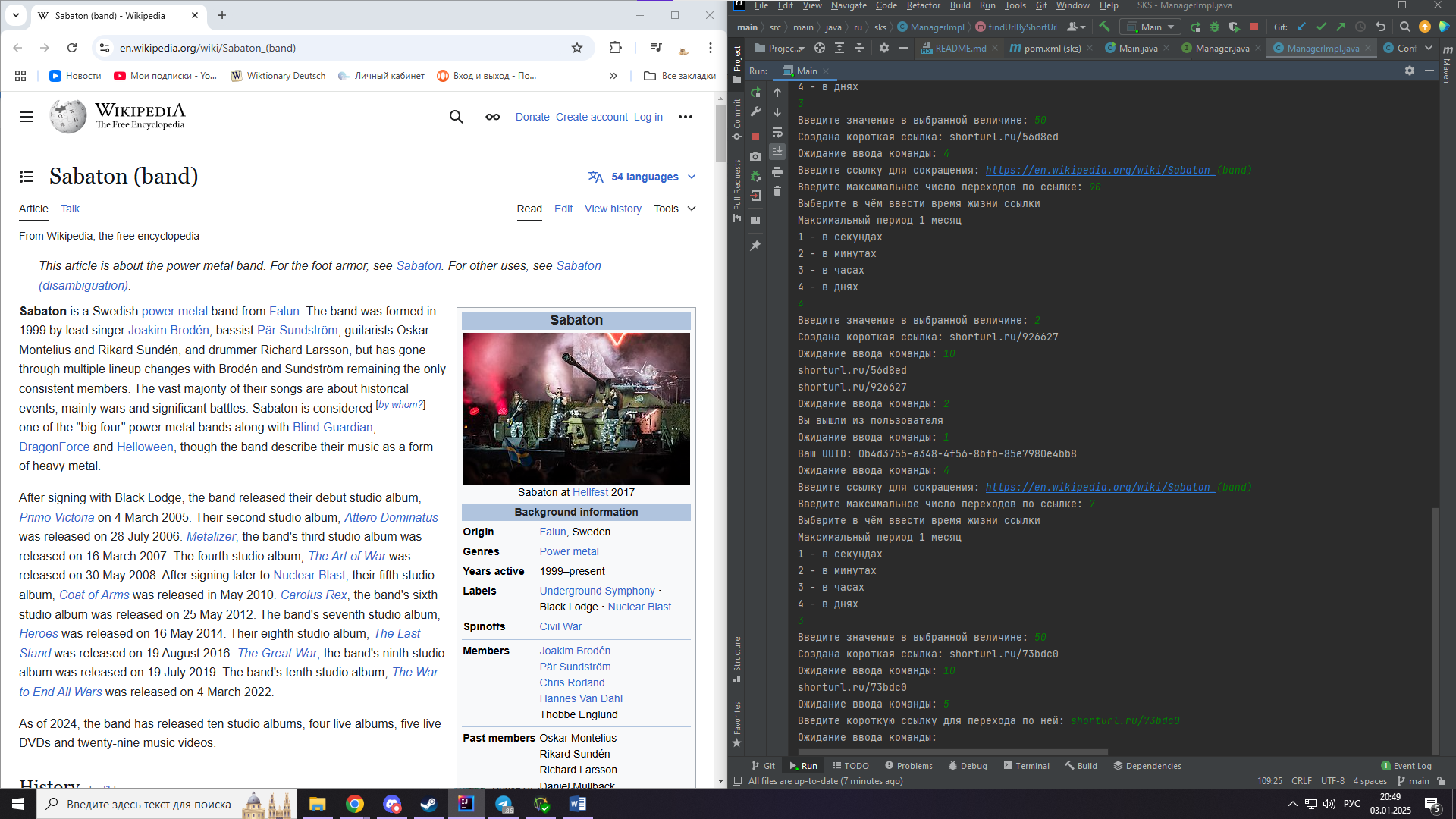
Под одну ссылку на Sabaton сгенерированы две разные короткие.

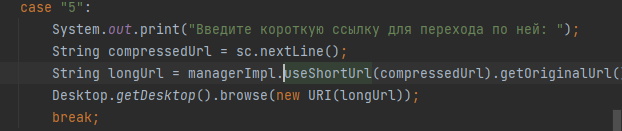
****

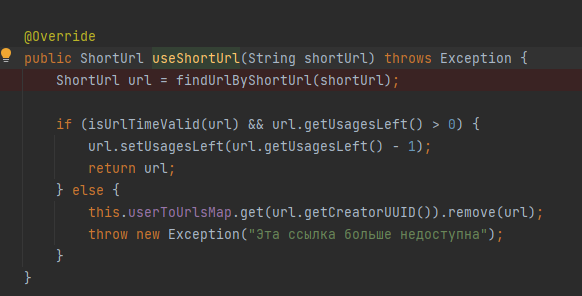
Также работает и для двух разных пользователей.

****

**Переход по ссылке**

****

****



Происходит поиск оригинальной ссылке по короткой и открытие страницы в браузере.

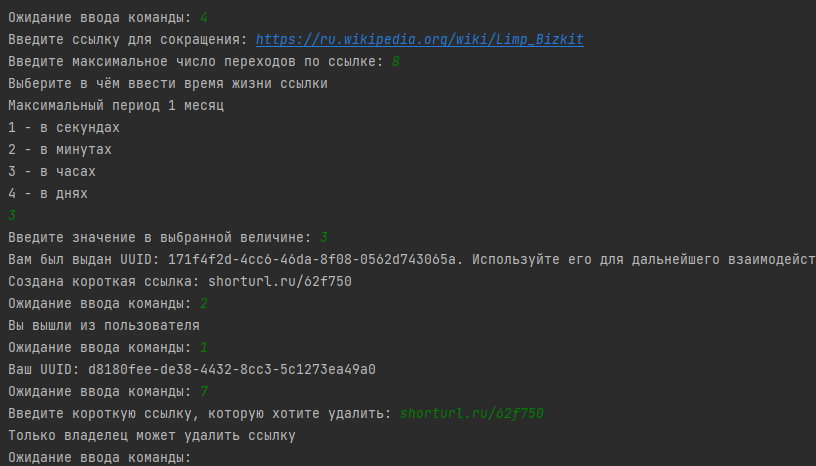
**Уведомление пользователя**

В момент запроса к невалидной ссылке пользователь получает уведомление.



**Удаление ссылки**

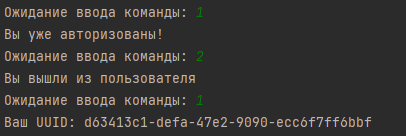
Ссылка может быть удалена только создателем.



**Идентификация по UUID**

UUID автоматически выдается при создании первой ссылки либо при исполнении команды 1 неавторизованным пользователем.

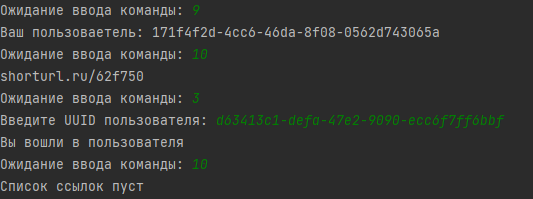




Узнать UUID текущего пользователя можно командой 9.



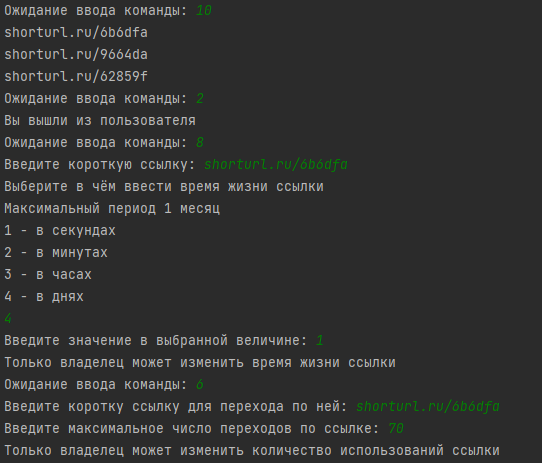
**Работа нескольких пользователей**

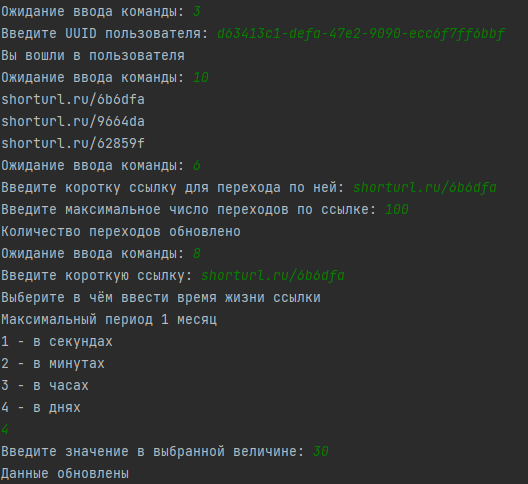
****

У каждого пользователя свой список ссылок, между пользователями можно переходить при помощи команды 3. Посмотреть список ссылок через команду 10.

**Администрирование параметров ссылки, редактирование лимита переходов и лимита времени жизни ссылки**

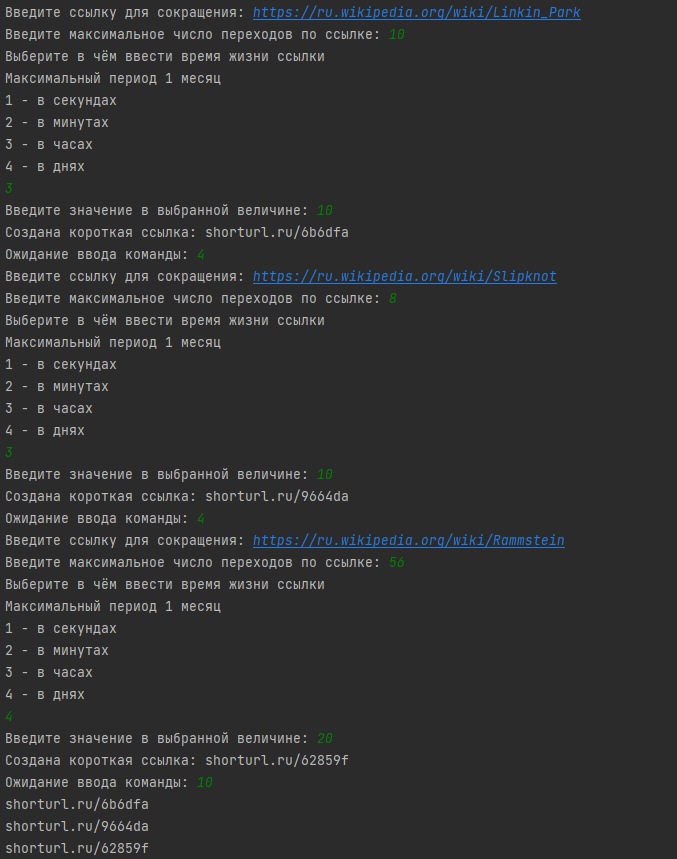
По командам 6 и 8 доступно изменение параметров ссылки, изменение доступно только создателю.

****

****

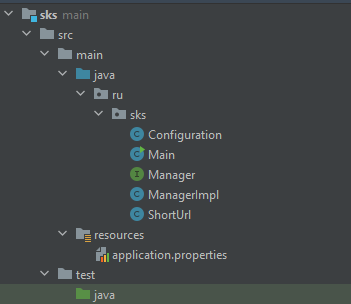
**Создание нескольких ссылок и идентификация по UUID при повторных запросах**

Как видно из скриншота все создаваемые ссылки привязываются к текущему пользователю по его UUID, то есть система идентифицирует пользователя по UUID.

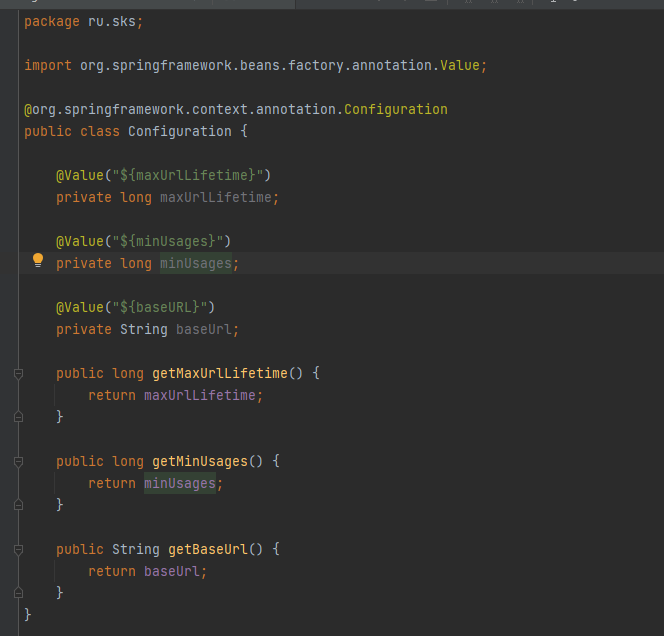
****

Ссылка на проект: <https://github.com/SavageSyrop/SKS>

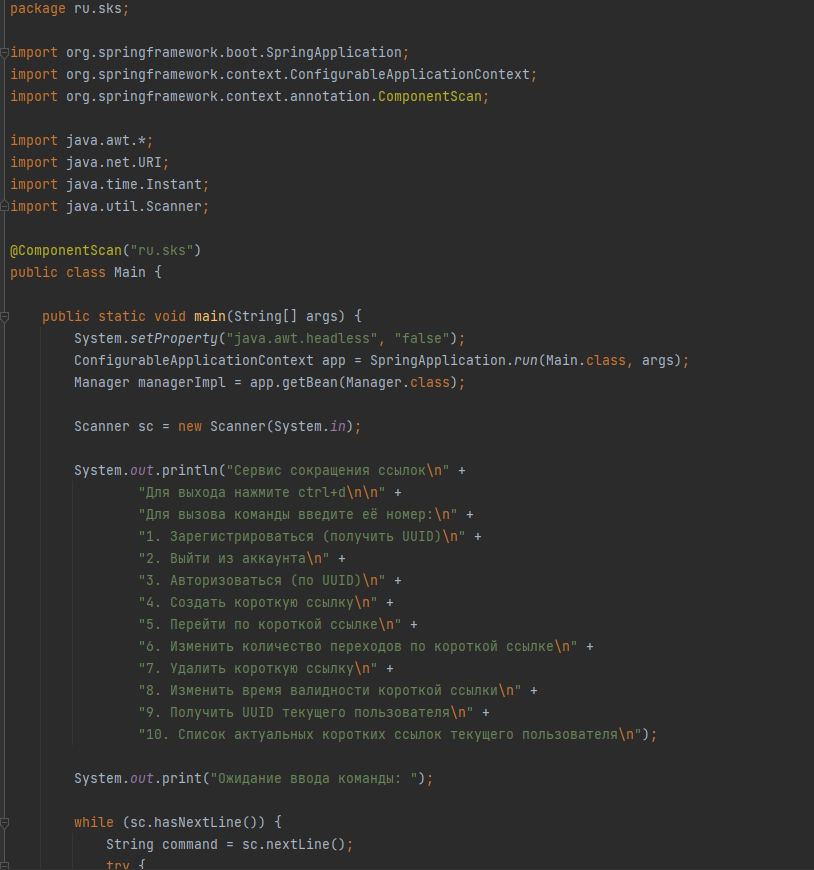
**Листинги кода**

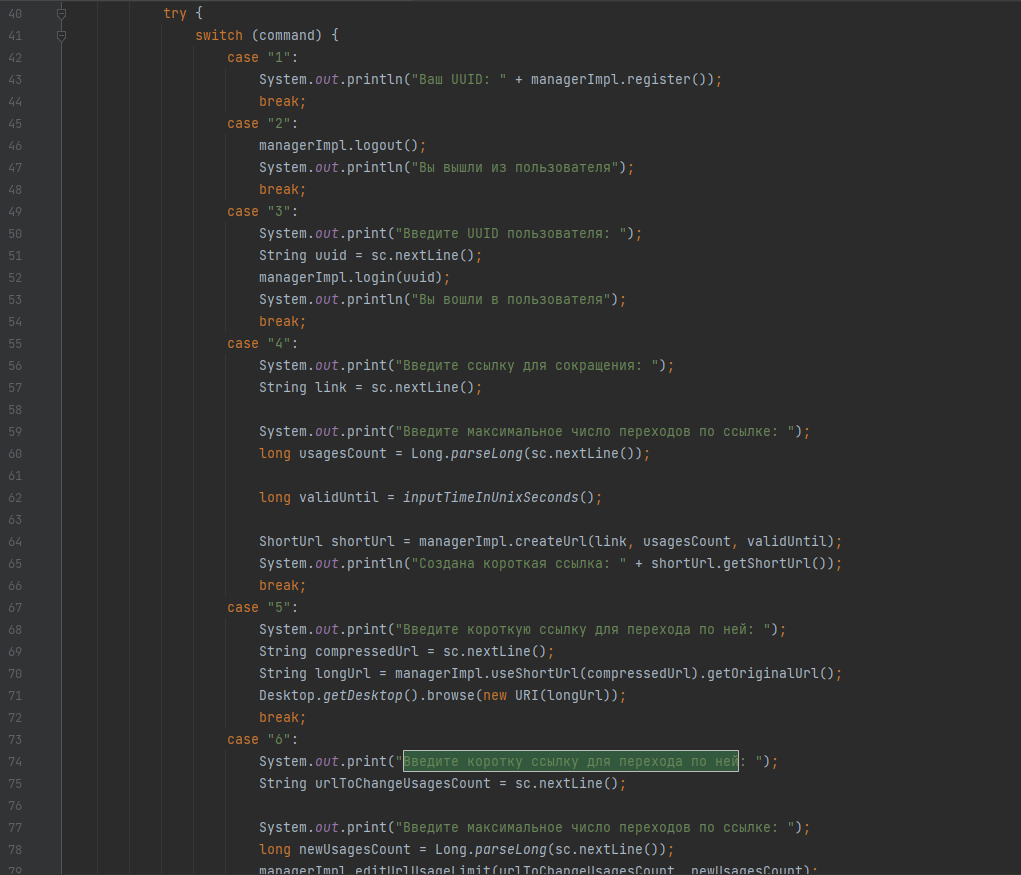
****

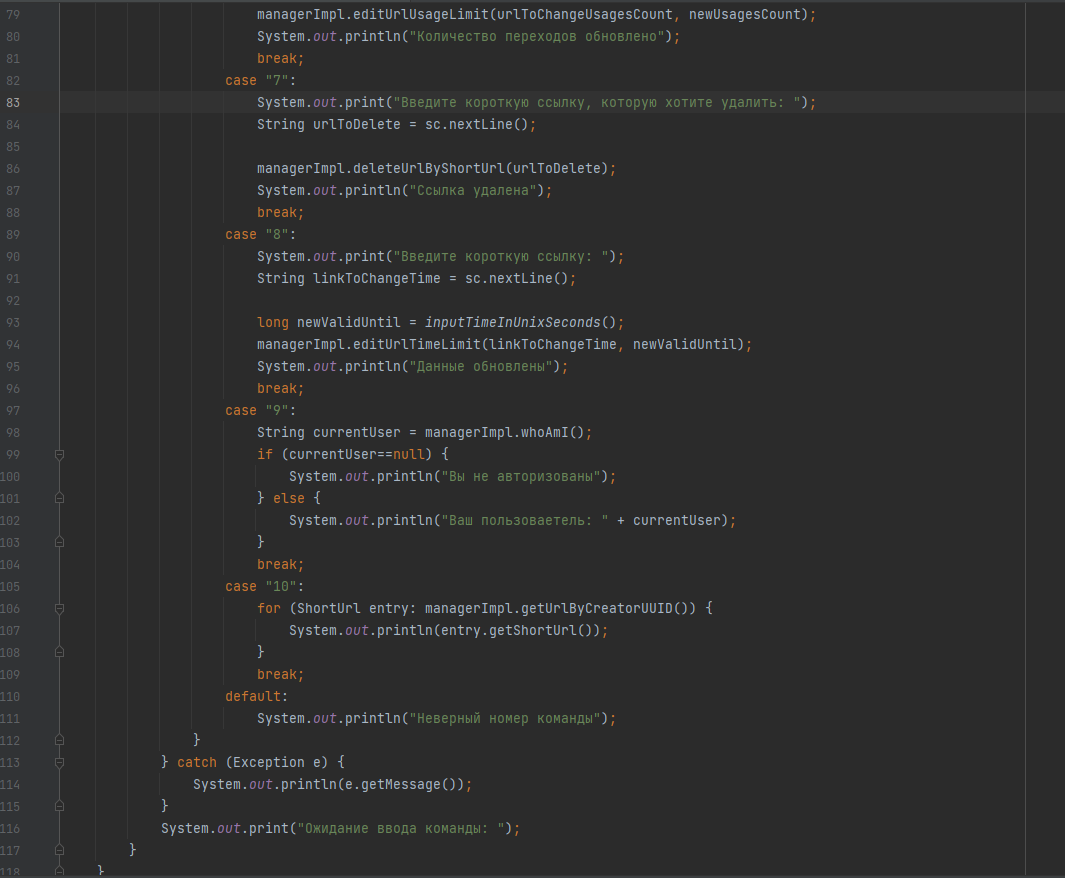
**Configuration.class**

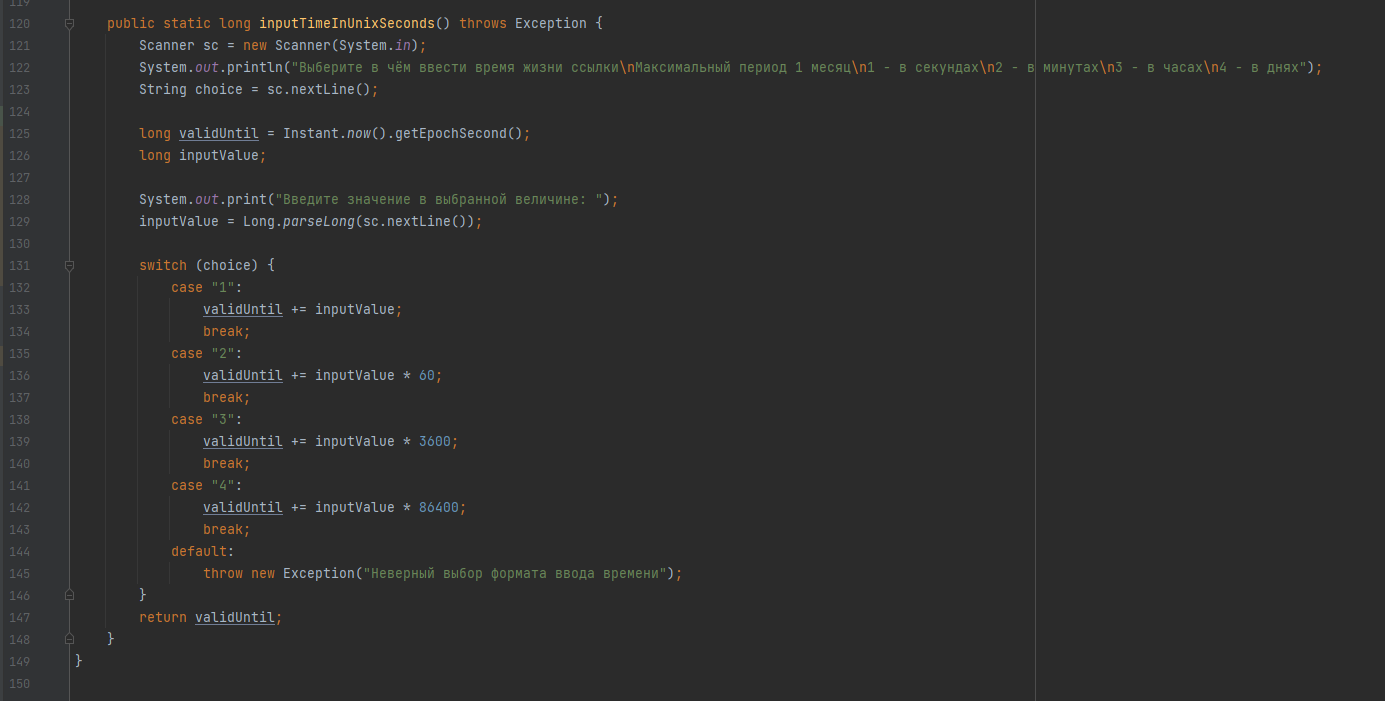
****

**Main.class**

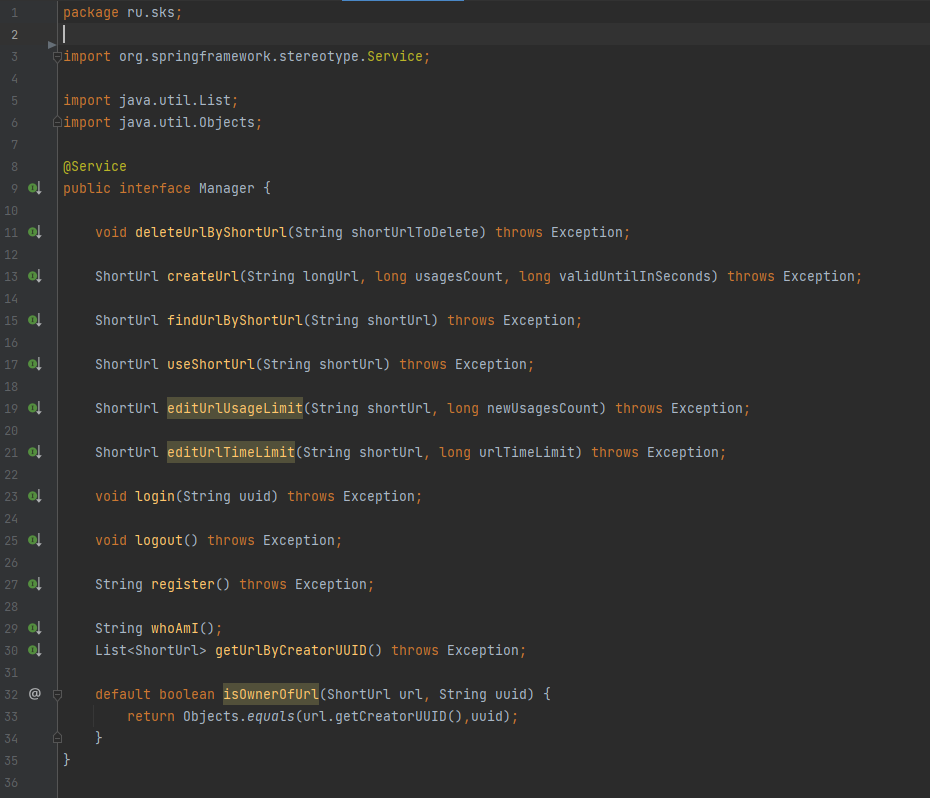
****

****

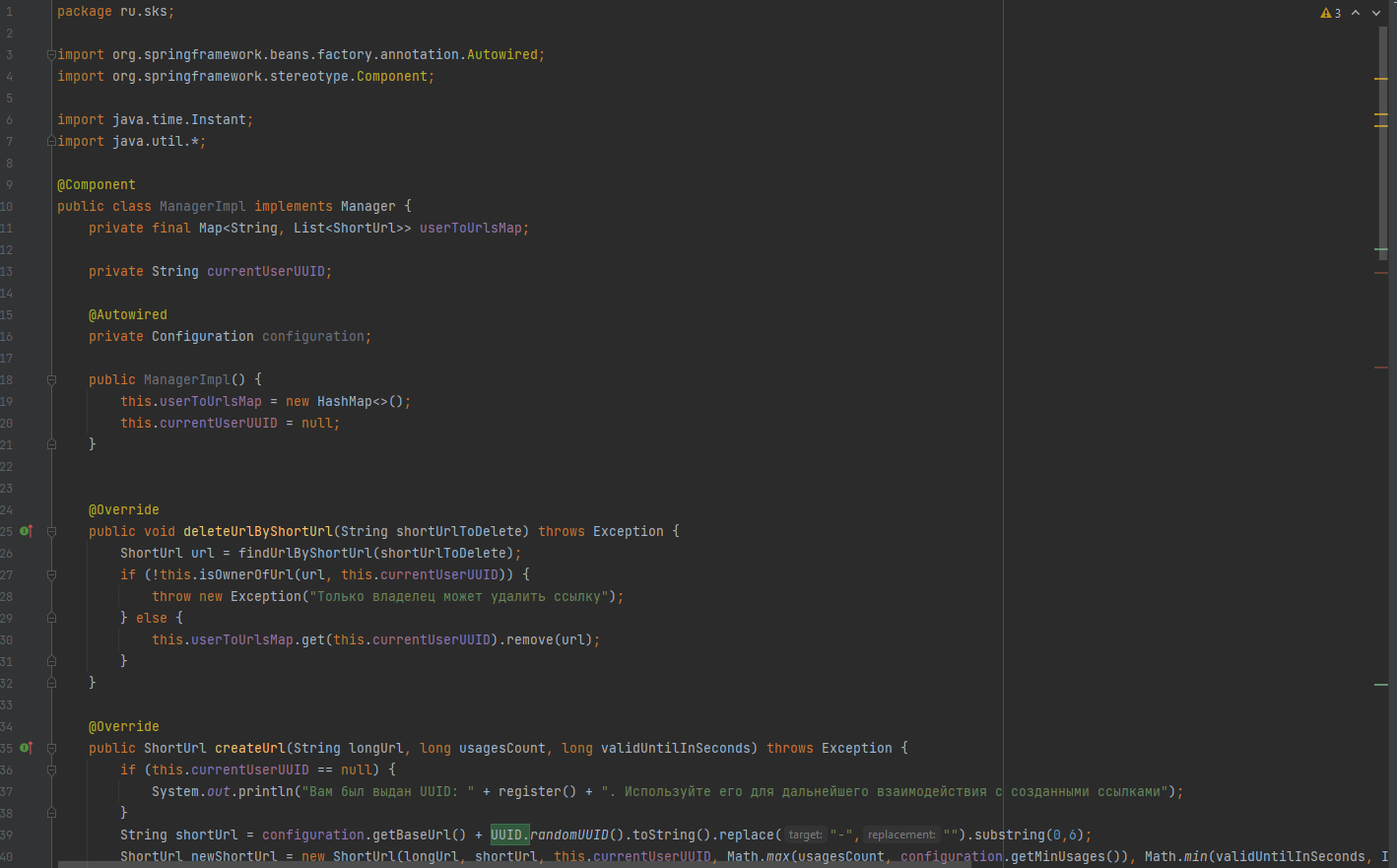
****

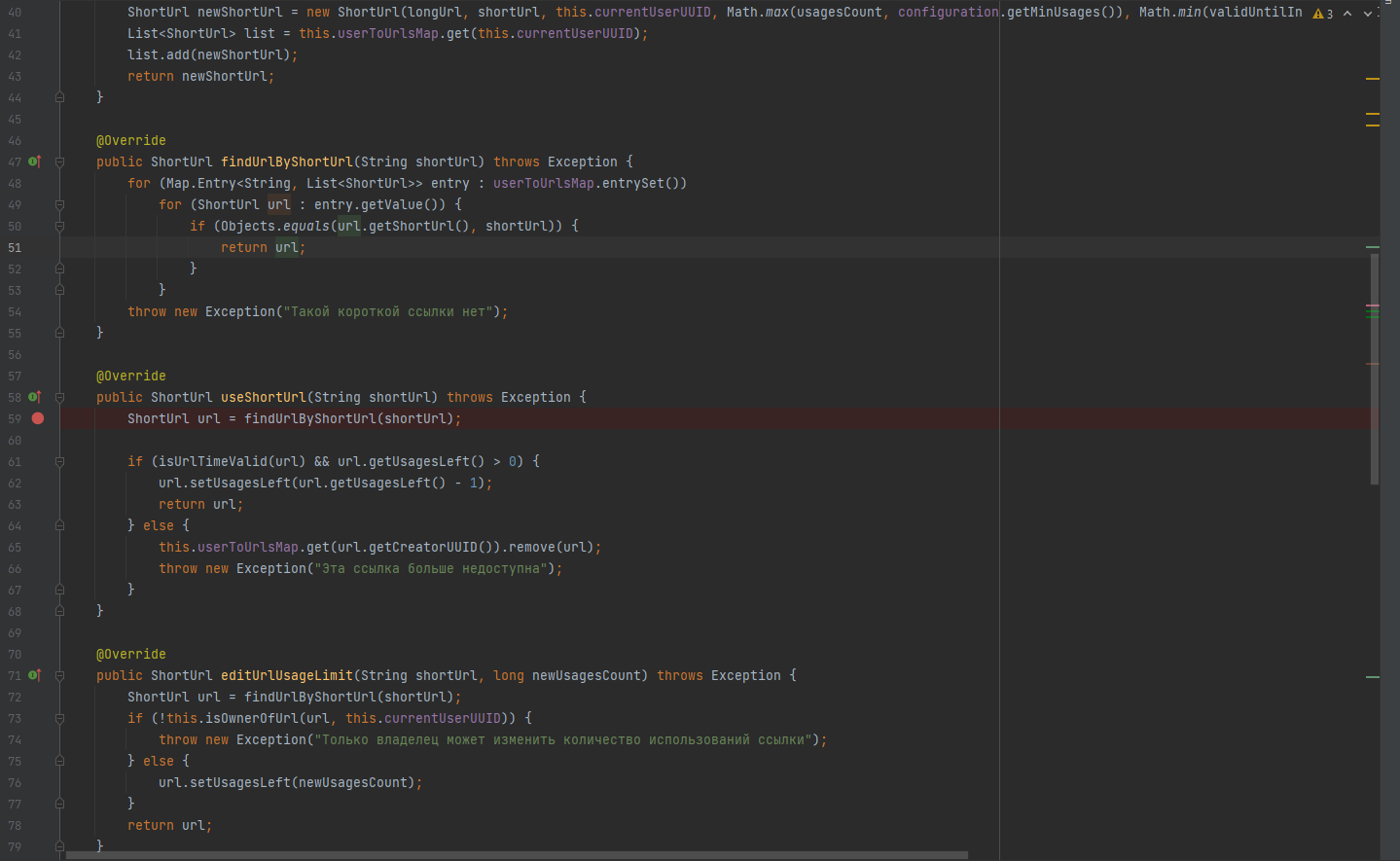
****

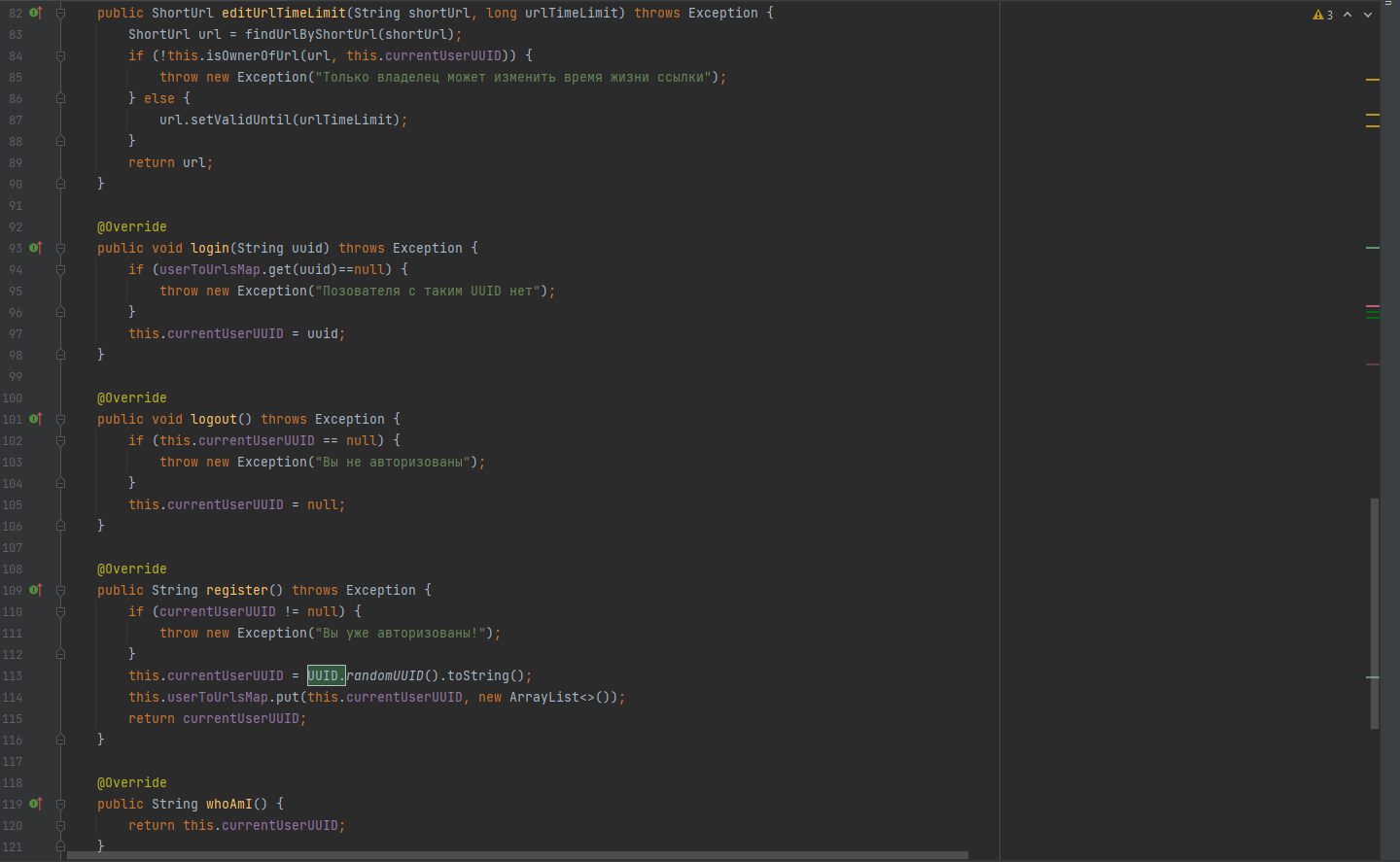
**Manager.class**

****

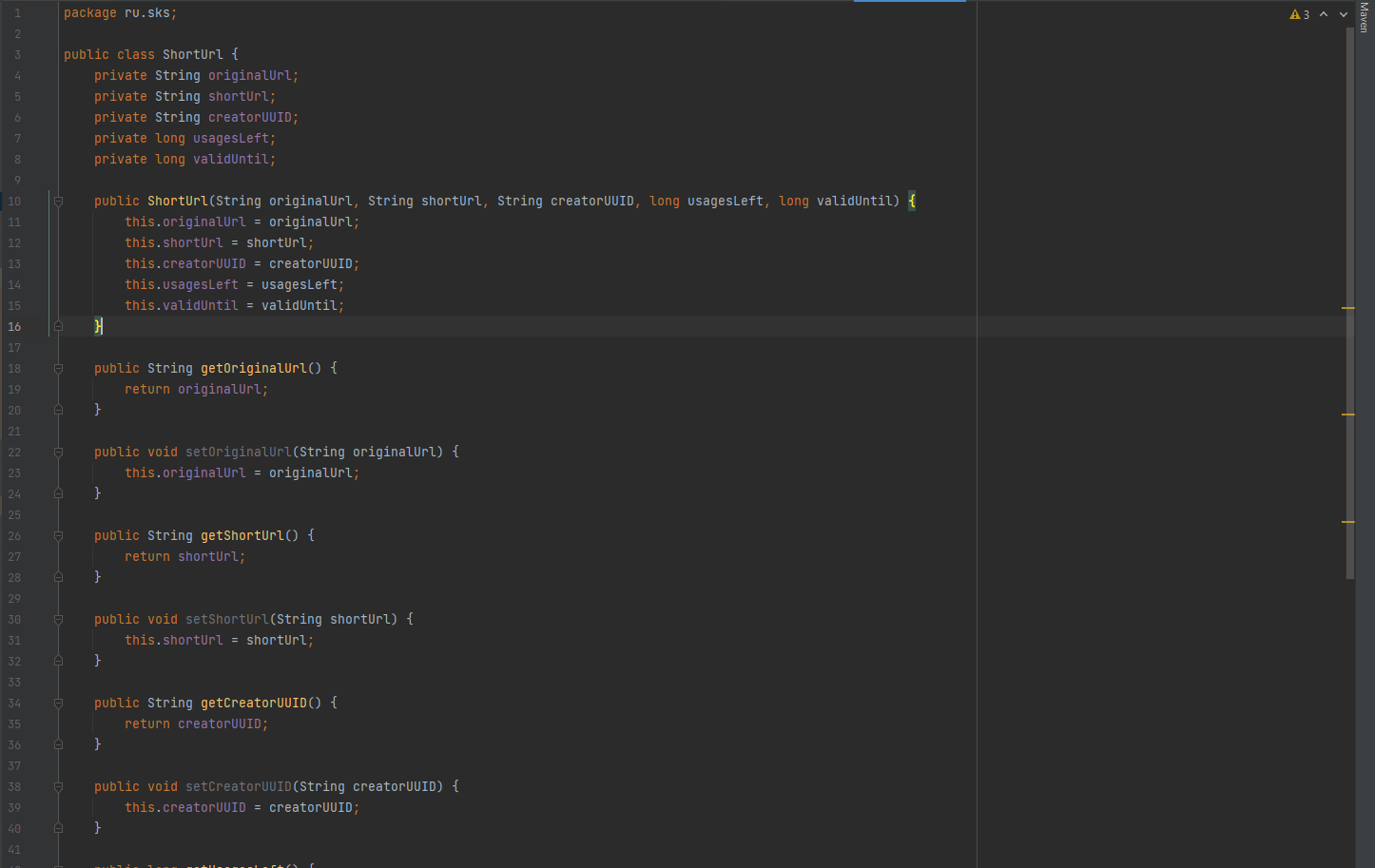
**ManagerImpl.class**

****

****

**  
**

**ShortUrl.class**

****

****