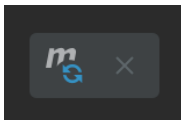


Задание 1. Напишите программу, которая будет скачивать с сайта все изображения

- В рамках этого задания вам предстоит написать код, который будет получать веб-страницу из Интернета, парсить её код, получать из него пути к изображениям и скачивать их в папку.
- Создайте новый Maven-проект ImagesDownloader, создайте также в нём класс Main с методом main допишите после круглых скобок метода “throws Exception”.
- Подключите к вашему проекту библиотеку Jsoup при помощи Maven (<https://mvnrepository.com/artifact/org.jsoup/jsoup>). Перейдите в документации на страницу последней версии библиотеки и скопируйте оттуда код для вставки в файл pom.xml, например:

```
<dependency>
  <groupId>org.jsoup</groupId>
  <artifactId>jsoup</artifactId>
  <version>1.15.3</version>
</dependency>
```

- Поместите этот код в тэг <dependencies></dependencies> файла pom.xml вашего проекта. Если слова “org.jsoup”, “jsoup” и номер версии отображаются в коде XML-файла красным, нажмите на кнопку обновления (“Load Maven Changes”):



- В методе “main” напишите код, который будет получать страницу какого-нибудь сайта в виде объекта класса Document, например:

```
String url = "https://skillbox.ru/";
Document doc = Jsoup.connect(url).get();
```

- Получите все элементы веб-страницы, созданные при помощи тэга “img”, и добавьте ссылки на изображения из этих тэгов в какой-нибудь HashSet:

```
Elements images = doc.select("img");
for (Element image : images) {
```

```
links.add(image.attr("abs:src"));  
}
```

- Каждая такая ссылка является путём к файлу с изображением, содержащим имя этого файла, например, "prices.jpg", состоящая, собственно, из имени файла (prices) и его расширения — jpg. Все эти файлы вам нужно будет скачать и сохранить в какой-нибудь одной папке.
- Поскольку на одной странице могут быть изображения с одинаковыми именами файлов, мы предлагаем именовать их числами — от 1. После каждого такого числа необходимо писать расширение, поэтому первым делом напишите цикл по всем полученным ссылкам и для каждой из них сформируйте имя файла из числа и расширения. Расширение получите следующим образом:

```
String extension = link  
    .replaceAll("^.+\\.\\.", "")  
    .replace("?.+$", "");
```

Данное регулярное выражение уберёт из ссылки всё в начале до последнего слэша, включая этот слэш, и параметры в конце, начинающиеся со знака вопроса. Например, из ссылки:

<https://248006.selcdn.ru/MainSite/skillbox.svg?uid=472394>

будут удалены все части, отмеченные красным цветом.

- Затем напишите код, который сформирует имя файла, например:

```
String filePath = "data/" + number++ + "." + extension;
```

Число number создайте перед циклом и задайте ему начальное значение, равное 1.

- Создайте метод download, который будет скачивать файл по указанной ссылке и сохранять по переданному адресу, используя InputStream и OutputStream. InputStream для скачивания файла можно получить так:

```
URLConnection connection = new  
URL(link).openConnection();  
InputStream inStream = connection.getInputStream();
```

- Вызов метода `download` в цикле оберните в конструкцию `try...catch`, чтобы в случае ошибки скачивания какого-либо отдельного файла, вся программа целиком работать не переставала.
- Запустите получившуюся программа и проверьте, что она работает успешно. Если что-то не получилось, постарайтесь самостоятельно добиться необходимого результата.
- В случае возникновения трудностей, вы, как и всегда в заданиях для самостоятельного выполнения, можете воспользоваться рекомендациями под видео, под которым была указана ссылка на это задание.
- Если всё получилось, поздравляем! Теперь вы умеете парсить HTML-страницы в Интернете и скачивать из Интернета файлы.