Лекции Поиск

Карта квестов Лекции CS50 Android

Создание запроса с помощью HttpRequest

```
JSP & Servlets
10 уровень, 1 лекция
```

ОТКРЫТА

Методы newBuilder(), build()

Kласс HttpRequest используется для описания http-request, что легко понять из его названия. Этот объект сам по себе ничего не делает, он только содержит разнообразную информацию по поводу http-запроса. Поэтому, как ты уже, наверное, догадываешься, для его создания тоже используется шаблон Builder.

```
1  HttpRequest request = HttpRequest.newBuilder()
2    .method1()
3    .method2()
4    .methodN()
5    .build();
```

Где между вызовами методов newBuilder() и build() нужно вызвать все методы для конструирования объекта HttpRequest .

Пример простейшего запроса выглядит так:

Bce методы класса | HttpRequest | ты можешь найти по ссылке в официальной документации.

А дальше мы разберем самые популярные из них.

Метод uri()

С помощью метода uri() можно задать URI (или URL), к которому будет отправлен http-запрос. Пример:

```
1 HttpRequest request = HttpRequest.newBuilder()
2 .uri( URI.create("http://javarush.ru") )
3 .build();
```

Кстати, можно записать этот код еще немного короче, передав URI прямо в метод newBuilder():

```
1 HttpRequest request = HttpRequest.newBuilder( URI.create("http://javarush.ru") ).build();
```

Важно! URI можно создать двумя способами:

- new URI(String)
- URI.create(String)

Второй способ предпочтительнее. Первый способ, к сожалению, не очень хорош, потому-то конструктор URI объявлен так public URI(String str) throws URISyntaxException, a URISyntaxException — это checked-исключение.

Mетоды GET(), POST(), PUT(), DELETE()

Задать http-метод запроса можно с помощью методов:

```
• GET()
• POST()
• PUT()
• DELETE()
```

Вот как будет выглядеть простой GET-запрос:

```
1 HttpRequest request = HttpRequest.newBuilder()
2    .uri(new URI("https://javarush.com"))
3    .GET()
4    .build();
```

Meтод version()

Также можно задать версию НТТР-протокола. Их всего 2 варианта:

- HttpClient.Version.HTTP_1_1
- HttpClient.Version.HTTP_2

Допустим, ты хочешь создать запрос по протоколу HTTP/2, тогда тебе нужно будет написать:

```
1  HttpRequest request = HttpRequest.newBuilder()
2    .uri(new URI("https://javarush.com"))
3    .version( HttpClient.Version.HTTP_2 )
4    .GET()
5    .build();
```

Очень просто, не правда ли? :)

Mетод timeout()

Также можно задать время выполнения запроса. Если оно пройдет, а запрос так и не будет выполнен, то выкинется исключение **HttpTimeoutException**.

Само время задается с помощью объекта Duration из Java DateTime API. Пример:

```
1  HttpRequest request = HttpRequest.newBuilder()
2    .uri(new URI("https://javarush.com"))
3    .timeout( Duration.of(5, SECONDS) )
4    .GET()
5    .build();
```

Наличие этого метода показывает, что классы [HttpClient] и [HttpRequest] могут решать самые разнообразные задачи. Представь, что ты выполняешь запрос, а что-то случилось с сетью и он продлился 30 секунд. Куда полезнее сразу получить исключение и правильно на него среагировать.

Метод header()

Также к любому запросу можно добавить сколько угодно заголовков. И сделать это так же просто, как и все остальное. Для этого есть специальный метод — header(). Пример:

```
1 HttpRequest request = HttpRequest.newBuilder()
2 .uri(new URI("https://javarush.com"))
3 .header("name1", "value1")
```

```
4    .header("name2", "value2")
5    .GET()
6    .build();
```

Есть еще один альтернативный способ задавания сразу много заголовков. Может пригодиться, если, ты, допустим, преобразовал список заголовков в массив:

```
HttpRequest request = HttpRequest.newBuilder()
    .uri(new URI("https://javarush.com"))
    .headers("name1", "value1", "name2", "value2")
    .GET()
    .build();
```

Все элементарное просто.

< Предыдущая лекция

Ответить

Следующая лекция >

0 0



 Комментарии (1)
 популярные
 новые
 старые

 ЈаvaCoder

 Введите текст комментария
 5 декабря 2022, 10:14
 •••

 Саша И. Уровень 67
 5 декабря 2022, 10:14
 •••

 "Есть еще один альтернативный способ задавания задать сразу много заголовков."

ОБУЧЕНИЕ	СООБЩЕСТВО	К ИНАПМОЗ
Курсы программирования	Пользователи	О нас
Kypc Java	Статьи	Контакты
Помощь по задачам	Форум	Отзывы
Подписки	Чат	FAQ
Задачи-игры	Истории успеха	Поддержка

Активности



RUSH

JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

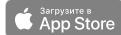
ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

Русский

СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ







"Программистами не рождаются" © 2023 JavaRush