

Miguel Jose Noboa Hurtado

10137875

20/5/2023

Practica 1

Programación Web

Para esta práctica tuvimos que conectarnos con un servidor mediante su URL y obtener la información que queremos. Todo el siguiente código se encuentra en el main y puede ser corrido llamando a gradle run.

Punto 1) Lo primero que tenemos que identificar al tipo de documento que nos lleva el URL. Para esto usaremos el http cliente que viene java. Creamos un HttpClient, request y response. Dentro de un try catch, usando el httpcliente, llamamos al URL y preparamos el request. Si lo acepta hacemos un client.send(request) para mandar la solicitud.

Si todo queda bien, de adentro el header, obtenemos el “content-type” el cual nos dira que tipo de documento es.

Punto 2) Si es de tipo HTML, tenemos que buscar información.

Para esta parte use Jsoup.

Nos conectamos al URL con el método GET usando este Codigo adentro de un try catch.

```
doc = Jsoup.connect(URL).get();
```

Con esto conseguimos que contiene toda la información recibida.

- a) Con esto podemos conseguir la longitud de líneas. Volvemos el documento completo a un string y lo separamos donde encontremos el carácter “/n”. Contamos el length de la lista de strings e imprimimos el número. Línea 80
- b) Para la cantidad de párrafos podemos hacer un `doc.select(“p”)`. Este código nos dará una lista de todos los párrafos en el documento, hacemos un `.size()` e imprimimos la cantidad. Línea 85.
- c) El siguiente nos pide la cantidad de imágenes en los párrafos. Usando la lista de párrafos que hicimos previamente, hacemos un for-each y seleccionamos “img”. A este le hacemos un `.size()` y vamos agregándolo a un contador. Después imprimimos el resultado. Línea 89.
- d) Para los forms haremos un `doc.forms()` para obtener todos los forms. También vamos a hacer un select de los forms pero adentro agregaremos lo siguiente `foc.select(“form[method=post]”).forms()` y otro pero en vez de post es get. Esto nos dará 3 lista de forms. Con estas podemos saber el numero de forms totales, los que tienen post y los que tienen get. Línea 95
- e) Con la lista de formularios anterior, podemos hacer un for each en cada uno. Adentro de este for-each buscaremos todos los inputs con un `select`. `Form.select(“input”)`. Con esta lista de inputs haremos otro for-each e imprimiremos el type de este, que lo encontraremos con `img.attr(“type”)`. Línea 133

```
Integer count = 1;
System.out.println("\nFormularios:\n");
for(FormElement f : forms){
    System.out.println("Formulario " + count);
    Elements inputs = f.select("input");
    for(Element i : inputs){
        System.out.println("    input -> type: " + i.attr("type"));
    }
    System.out.println();
    count++;
}
```

f) Finalmente haremos la ultima parte. Linea

Usando los formsPost (los forms que tienen el método post) del punto d podemos conseguir todos los forms que hay que mandar. Para saber a donde hay que mandarlos, tenemos que encontrar el atributo "action" en cada formulario. Hacemos esto con

String act = form.attr("action").toString(); Linea 138.

```
String act = f.attr("action").toString();
```

En la línea 140 hacemos un test para asegurarnos que el action tenga el url completo, pero si no lo completamos.

Con el url de redirección completo crearemos un objeto response con el siguiente código.

```
Connection.Response res =  
    Jsoup.connect(act).method(Connection.Method.POST).  
    header("matricula-id", "10137875").data("asignatura", "practical1").execute();
```

Le ponemos el método Post, le damos el header con el nombre de matricula-id y el valor de mi id que es 10137875. También le pasamos el atributo con el nombre asignatura y el valor practica1.

Al final imprimiremos el código de estado, y el body que recibimos.

Con esto terminamos.