# Guião (Inglês)

## Connection to the internet

Slide1

Welcome to the first class of our advanced java course. This introductory class is divided into three phases. In the first section we will discuss the topic of internet connection, starting with a review of some concepts, such as TCP, HTTP and HTML. Next you will be presented with an excerpt of java code that allows you to connect to the internet. Finally, the participant will be offered an exercise to improve their knowledge.

Slide2

We will then start with the TCP / IP model, responsible for all inter-entity links on the Internet. This is a complex model, composed of several layers that communicate with each other to ensure the connection and transfer of data between two machines connected to the Internet. We will, however, abstract ourselves from its complexity and concentrate on the peer-to-peer connection. For this we will use a java class, more specifically the URL class . This class allows the creation of a URL object from a string and also offers several methods for its manipulation and use. For those who are interested in more information, we provide here the documentation on this class, as well as a link to more information on the operation of the Internet. (and a last one for the source of the image relative to the TCP / IP model)

Slide3

Our communications over the Web will use HTTP, a protocol that is part of the application layer of the TCP / IP model. A simple protocol based on client requests, usually a web browser, to content hosted on a web server, which responds to requests by sending the requested content to the client. There is also, as seen in the image, the possibility of the server sending cookies to the client, which are then sent to the server along with future requests, allowing deeper communication between server and client. In Java there are three main classes that use this protocol for Internet connection: the URL / URL connection class, which will be our focus in this class, the HttpComponents class, belonging to the Apache library, and the Socket class.

Slide4

In addition to HTTP, responsible for communicating data, it is also in our interest to know HTML, the language used to format web pages. This is a language that allows the structured representation of text and multimedia contents, focused on the use of tags. A tag consists of a standardized text, delimited by the signs of less than and greater than, and is opened, after which we place the content we want the tag to apply to, and finally the tag is closed by rewriting the tag with a slash at the beginning of its text, marking the end of the content affected by the tag. This content can be text, links to images or to other websites, etc. When working with the internet it is essential to understand this language and know the main tags, and our classes will be no exception.

Slide5

Let's now move to the java code. This code snippet, encapsulated in a try / catch to avoid run-time errors that may occur while working in a web environment, begins by creating a new object of the URL class, which will be initialized with the link to the github page for this exercise. A String where we will store the web server response is also created at this point. Then we initiate a connection using the previous URL, which will result in a response from the web server containing the requested web page. The data returned by the server is then received by an InputStream, and stored in a buffer. Now it is only necessary to save the text, which at the moment is stored in the buffer, in the String created to receive the page information. A while loop allows you to receive all the text from the buffer, which is finally written to the console.

Slide6

The previous exercise allows us to receive, store and present the content of a page, but it is not an ideal solution. As can be seen in the html code at the top, which corresponds to the result of the previous java code, it is difficult to read the information contained in the page, since the HTML was not created to be used to store data, but to structure it for presentation in web pages. Thus, in one of the following weeks we will address XML and JSON, two interesting alternatives for this type of transfer.

Slide7

Finally, we have an exercise proposal, for those who want to practice what we learned in this video. This simple exercise is not very different from our example, but this time the purpose is specifically to get the images present on the indicated page. If you have difficulties and cannot resolve the exercise do not worry, since in our next video will begin with the resolution of this exercise. And so we reached the end of our first class, thank you for your presence and we hope to see you in the next video!

Exercise

Write a java program that accesses the url: https://java-mooc.github.io/Advanced-Java/ex2.html and downloads all the image links.

## Regular Expressions

Slide 1

Hello and welcome to our second class. Today we will address the theme of Regular Expressions, which, as we will see below, are a useful tool for the programmer, simplifying certain challenges that he may encounter. We will start by introducing the theme of regular expressions, delve into its benefits and usefulness, and leave you with some tools to practice. Before that, as mentioned in the previous lesson, let's start by showing a resolution to the problem presented in the previous lesson, which, as we will later see, can be improved by using regular expressions.

Slide 2

Let us then revisit the last class exercise. At the end of our last video, we proposed the creation of a program, from the example code explored in the lesson, that would obtain the links of the images from a page indicated by us. The code we have here is the necessary changes so that we have a String that stores the links of the images instead of all the content. However, this solution is confusing and ugly, and it’s not easy to perceive the purpose of the code. Therefore, it is possible to make it more elegant with the use of regular expressions, as we will now see.

Slide3

So, what are regular expressions? They are a mechanism, created by Stephen Cole Kleene, that allows searching and replacing expressions in a String, which can be characters, words or even patterns of characters. These expressions are written in a formal language, which can be interpreted by a regular expression processor. You can also validate text formats using this tool. With a focus on syntax and great flexibility that cannot be found in traditional search methods, which simply compare a fixed set of characters with a String, regular expressions are a great tool for information filtering.

Slide 4

These are the symbols and syntax of the regular expressions language, which can be interconnected between them to give rise to useful, flexible, and comprehensive expressions. The first symbols indicate the beginning and end of a line of text, the period symbol indicates the presence of any character, \ s indicates whitespace, while \ S indicates anything but whitespace. \* indicates zero or more occurrences of the previous expression, while + indicates one or more occurrences. These symbols are "greedy" by default, that is, if a regular expression is valid for part of a string and for another part contained in the first, a greedy expression will always return the largest possible result. To counteract this, a "?" can be added after \* or “.”, whichever is being used, which will result in the expression becoming “non-greedy” and returning the smallest possible result in case of conflict. Moving on, a set of characters enclosed in square brackets indicates the characters you want to find in that position, while a "^" before the characters indicates that you do not want those characters. A dash between two characters indicates the character set that exists between them, and finally curved parentheses indicate where to start and where to end extracting a String from the original text.

Slide 5

And so we finish the theory of regular expressions, as we know that this system can be confusing and hard to understand initially, we leave here a site where you can test regular expressions. In this site we just place a String that will be tested and the regular expression we want to test, and we immediately receive all matches found in the string, as well as an explanation of all parts of the expression written by us, indicating their meaning. This site has been tested by us and we believe it will be a very beneficial tool for those who are interested in practicing and testing regular expressions.

Slide 6

Finally, as we are in a Java course, here are examples of how to use this new tool. We have the more basic methods, which are part of the Java String class, and which allow us to find in and to split a String according to a regular expression. We also have access to a method that replaces the first occurrence of a regular expression with another String, and a final one which makes this same substitution, but for all occurrences of the regular expression. For those who are interested in more uses of regular expressions, there are also the Matcher and Pattern Classes, which can be very interesting for those of you who want to use regular expressions frequently or for specific purposes. And with that we have arrived at the end of our second lesson, thank you for your presence and we will see you in our next class!

Exercise