Use of API recommendations in development tasks

This document describes two exercises assigned to the participants in order to evaluate whether automatically-generated API recommendations help developers with their implementation tasks.

We kindly ask you to complete two simple Java projects. These projects include two well-known libraries, i.e., **jsoup** and **apache-cli**. In one case, you have recommendations, in the other case, you do not.

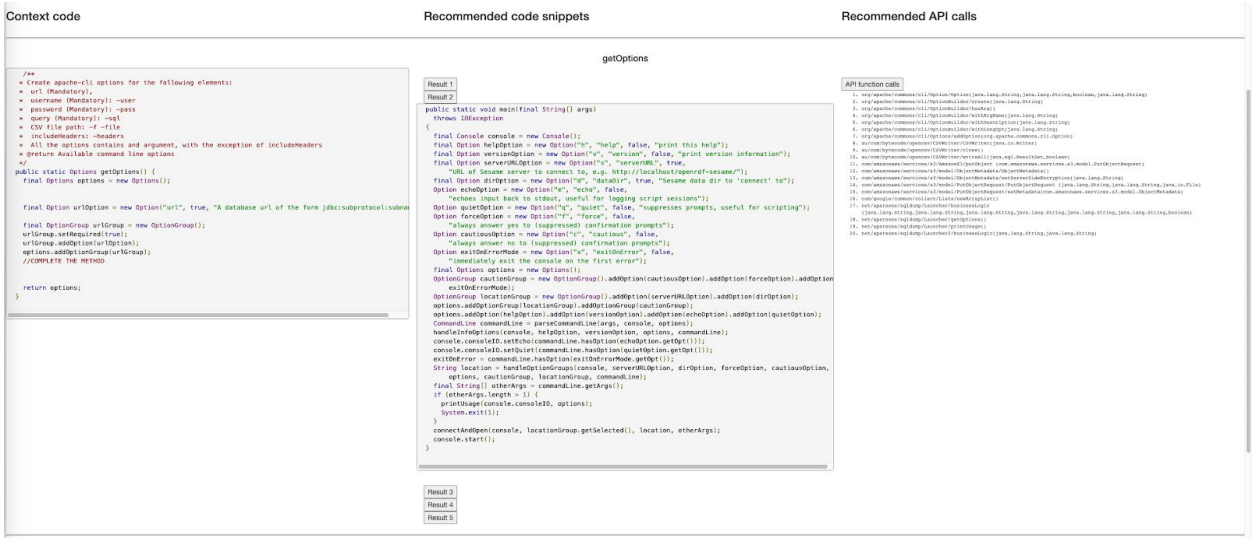
You can use both API function calls and code snippet recommendations provided by the tool FOCUS.

Projects sketches are available at https://github.com/MDEGroup/FOCUS-user-evalaution.

Once finished, the exercises have to be pushed to the assigned branch.

How to use the recommendations

For each task, FOCUS recommends top-5 code snippets and top-20 method invocations.
The following picture shows how the recommendations are shown to you.



A recommendation consists of the method to be completed (context), the top-5 code snippet recommendations (visible by clicking the corresponding button), and top-20 suggested method invocations.
Method invocations are represented with the AST canonical name. For instance, the following method invocation refers to the method `create` of the `OptionBuilder` class, which is

contained in the package `org.apache.commons.cli`. This method accepts a string as a parameter.
`org/apache/commons/cli/OptionBuilder/create(java.lang.String)`
Disclaimer: The recommended method invocations do not discriminate between instance and class methods.

Please carefully inspect the recommendations (if available) to complete your tasks.

Preliminaries

As first steps, you have to perform the following steps:

- Create your fork of https://github.com/MDEGroup/FOCUS-user-evalaution
- Clone it
- Checkout the branch corresponding to your ID, named “Evaluator-<ID>” e.g., “Evaluator-1”
- After that, you can edit the project using the IDE you are most comfortable with, and build it in the IDE or using the pom.xml

Exercise 1: Scraping HTML pages by jsoup

This time, you will perform the task having API recommendations provided.
jsoup-example is a dummy maven project that uses the open-source library **jsoup** to scrape information from Web pages.
jsoup is a Java library for working with real-world HTML. It provides a convenient API for extracting and manipulating data, using the best of DOM, CSS, and jquery-like methods.

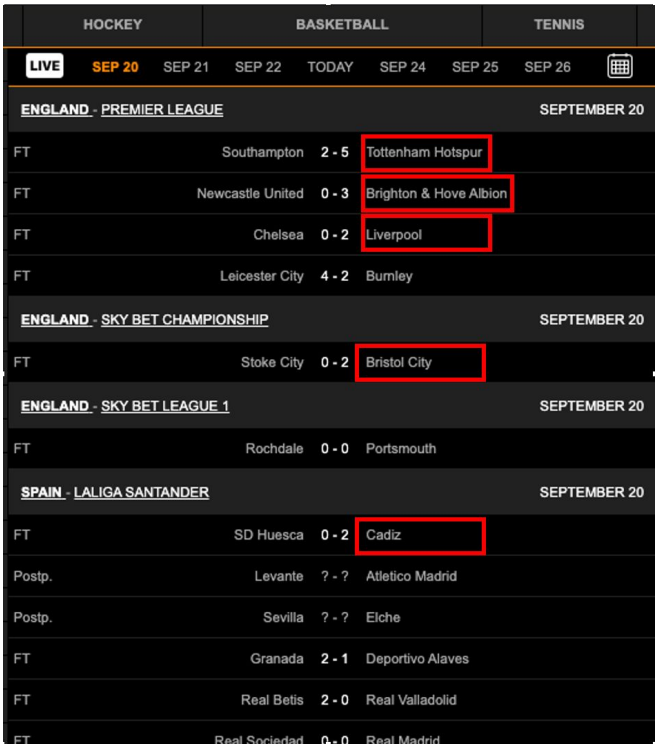
Please finalize the three methods in the App class to complete the following tasks. Each task comes with an example, and a table provides the method name, the list of recommendations, and the page that you have to parse.

1. Complete the `getCalciomercatoNews`" method: scrape the Juventus supporter's mobile page and count all news from the "calciomercato" category.

   In the following picture, the highlighted elements representing two pieces of news belong to the "calciomercato" category.

Method name	`getCalciomercatoNews`
Recommendation url	https://mdegroup.github.io/FOCUS-Appendix/userEvaluation_json.html#query1
Web page url	https://mdegroup.github.io/FOCUS-Appendix/tuttojuve.htm

2. Complete the `getWinningAwayTeams` method: scrape live football results website: extract the list of teams that won the match. Use "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)" as a user agent string.



Method name	`getWinningAwayTeams`
Recommendation url	https://mdegroup.github.io/FOCUS-Appendix/userEvaluation_json.html#query3
Web page url	https://mdegroup.github.io/FOCUS-Appendix/livescore.html

3. Complete the `getWinningAwayTeams` method: scrape a weather website to discover the day (i.e., "Martedì 29")* with the biggest temperature gap between max and min.

Method name	`getDayWithBiggerTemperatureDifference`
Recommendation url	https://mdegroup.github.io/FOCUS-Appendix/userEvaluation_json.html#query4
Web page url	https://mdegroup.github.io/FOCUS-Appendix/meteo.html

For this task, please use FOCUS and keep track of the time required to implement each method.

Exercise 2: Parsing command line parameters by apache-cli library

This time, you will perform the task without having API recommendations.
SQLDump is a simple java project that exports SQL data to CSV files. It is a command-line utility to execute SQL queries and export results to a CSV file. Apache-cli is used for taking parameters from the command line.

The Apache Commons CLI library provides an API for parsing command-line options passed to programs. It's also able to print help messages detailing the options available for a command-line tool.

SQLDump usage by command line:

```
java -jar SQLDump-0.4.jar -url [jdbc:oracle:thin:@hostname:port:sid] -user [username] -pass [password] -sql [query]
```

Please finalize the three methods in the Launcher class to complete the following tasks. Each task comes with an example, and a table provides the method name.

1. Complete the `getOptions` method: create an `Options` object that contains the following Option:
   - url (Mandatory),
   - username (Mandatory): `-user <user>`
   - password (Mandatory): `-pass <password>`
   - query (Mandatory): `-sql <query>`
   - CSV file path: the argument can be passe by `-f <filepath>` or `-file <filepath>`
   - includeHeaders: `-headers`

   All the options include an argument, with the exception of `includeHeaders` one.

Method name	`getOptions`

2. Print the command-line options to the console and return print usage as a string.

Method name	`printUsage`

3. This method parses the command line parameters and puts them to a hash map. The hash map consists of the parameter name as key and the argument as value. In this case, the option does not contain arguments please use “true” as value (e.g., headers).

   The hash map must contain all the required parameters. If a mandatory one is missed, it prints the parameter usage and throws an exception.

Use `getOption` and `printUsage` methods that you defined in the previous tasks.

Method name	`parse`

For this task, please use FOCUS and keep track of the time required to implement each method.

After you have completed

When you have completed both tasks,

1. Push the changes in your fork
2. Open a pull request the same branch of the origin i.e., on https://github.com/MDEGroup/FOCUS-user-evalaution
3. Finally, please kindly fill the form at the following URL: https://forms.gle/UDVxaHFUegJeABfS9