Department Mathematik/Informatik, Abteilung Informatik
Software & System Engineering Group
Software Technology Group
Lecture Object-Oriented Software Engineering, SS 2024

Alexander Korn, Adrian Bajraktari,
Prof. Dr. Michael Felderer

Homework 0. Set Up and Git

Hand out: 08.04.2024, 18:00
Hand in: 19.04.2024, 14:00

Remarks: This sheet is intended to guide you through preparing the basics you need for this course and will neither be handed in, nor be graded. Nonetheless, you need to have everything on this sheet done to successfully work on the upcoming sheets. You must solve each task on your own, not in your submission group! We expect that you (tried to have) solved everything from this sheet before the first exercise session. Thus you should work on this sheet until Friday, the 19th of April, as the exercises will start next week already.

General Remark: Start early so you can ask questions to your tutor in the exercise groups.

Exercise 1. Install Java

Check if you have Java 21 installed:

```
java -version
javac -version
```

If you do not have Java 21 installed, continue as follows:

1) Download the JDK from https://www.oracle.com/java/technologies/javase/jdk21-archive-downloads.html or https://adoptium.net/de/temurin/releases/?version=21 and make sure to select the one that fits your system. Download and install the JDK by following the instruction of the installer.
2) Check if you succeeded by running the above commands again.
3) Set the JAVA_HOME variable in your system (if not set already). Consult the internet (e.g., here) for how to do this, as this is specific to your system.

[0 points]

Exercise 2. Install Git

Install git from https://git-scm.com/downloads. Select the version specific to your system, download the installation package and follow the instructions of the installer. Make sure to set

"merge" as default merge strategy!

[0 points]

Exercise 3. Install the JetBrains IDE Toolbox

We will use different programming languages in this course. JetBrains provides different variants of their IntelliJ IDE for different languages, all sharing the same look and feel. The community editions may suffice for our purposes, but you can get the ultimate editions of all IDEs by regstring with your student email address, see here: https://www.jetbrains.com/de-de/lp/cpp-leaflet/students/. Install the JetBrains toolbox: https://www.jetbrains.com/de-de/toolbox-app/. Then, install the IDEs:

- IntelliJ https://www.jetbrains.com/help/idea/installation-guide.html
- PyCharm https://www.jetbrains.com/help/pycharm/quick-start-guide.html
- WebStorm https://www.jetbrains.com/help/webstorm/installation-guide.html
- CLion https://www.jetbrains.com/help/clion/quick-tutorial-on-configuring-clion-on-windows.html

[0 points]

Exercise 4. Optional: Install an IDE

If you do not wish to install JetBrains products, you may use an IDE for each language on your own. But be warned that we did not test our exercises for any other IDEs!

[0 points]

Exercise 5. Set Up Git in your IDE

Create an account on GitHub if you have not already. Create a personal access token and advise your IDE to use it when connecting to a GitHub repository.

See https://docs.github.com/en/enterprise-server@3.6/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens to know how to generate a personal access token. Create a "classic" token, set an expiration date past the end of September, and select all permissions. This way, you won't have any problems later on if your token misses a permission for a certain task. See https://www.jetbrains.com/help/idea/github.html to know how to use the token in IntelliJ.

[0 points]

Exercise 6. Homework Participation

Create a public repository on GitHub where you will upload your homework solutions (https://docs.github.com/en/repositories/creating-and-managing-repositories/quickstart-for-repositories). In a later step, we will ask you to add us to your repository. **Hint for groups**: If you plan to submit homework in groups (up to 5 people), please only submit to one repository.

[0 points]

Exercise 7. Working with Git

In this exercise, you will perform some common workflow steps with git to integrate a toy project in your repository. Go through the following steps. You do not need to submit any additional "protocol" of your working, but you should indicate (e.g., via a txt file) whether you needed any additional steps not given below.

1) Clone your GitHub repository.

2) Create a new branch with your name in CamelCaseFormat and check it out.

3) In the source folder, create a new class `Main.java`, open it and insert the following code:

```
1 public class Main {
2   public static void main(String[] args) {
3     System.out.println("Hello, world!");
4   }
5 }
```

   Run the project (with the green play button). If everything works, the text `"Hello, world"` should be displayed in the console.

4) When the project is executable, it is time to share it in GitHub. Create a new commit with the file you created and push the whole branch to the group repository.

[0 points]

$\sum$ **0.0 points**