WHITE BOX TESTING
What Every Tester Should Know

1. **Know what you are testing.**
   - Act as a consumer of the class and test that it fulfills its requirements.
   - Ask yourself:
     - What is the purpose of the class?
     - Who are the consumers?
     - Is the class designed by contract?   (Garbage In, Garbage Out)
       or does it follow a defensive design?   (Handle All Garbage In)

2. **Test behaviors and results, not implementation.**
   - Prevents tests from failing due to internal changes that do not represent defects:
     - Focus on the public interface, not the private members.
     - If there is complex private logic that needs to be explicitly tested, consider refactoring the code.

3. **Test one thing at a time.**
   - Each test should have a clear, concise singular objective.

4. **Make tests readable and understandable.**
   - Name tests using a short description of the scenario being tested and the expected result.
   - Using a fluent syntax allows test code to be self-documenting.

5. **Make tests independent and self-sufficient.**
   - Setup, execution, and results of a test should not depend on running other tests before it.
   - Isolate classes when dependencies are: (a) not available or not implemented yet, (b) spread across system boundaries, (c) difficult to setup or reproduce, (d) non-deterministic, (e) long running or (f) highly complex.

6. **Make tests deterministic.**
   - Avoid conditional logic (if, then, try, catch), loops, and sleeps.

7. **Repeat yourself when necessary.**
   - It is okay to violate the "Do not Repeat Yourself" (DRY) principle if it keeps tests simple and understandable.
   - Maintaining three or four similar tests is preferable to not understanding one non-duplicated test when it fails.
   - Parameterized tests are a good way to reduce duplication without obscuring the tests.

8. **Measure code coverage but focus on test coverage.**
   - Do not be misled that achieving 100% coverage of statements and branches indicate that you have a strong test set.
   - Good test coverage is achieved by executing methods with a variety of values.
   - Use testing methodologies (boundary, equivalence class, etc.) to improve test coverage.
   - This is likely to improve code coverage and is a better measure of the test effectiveness.
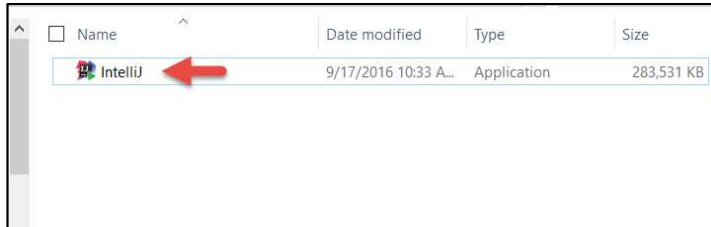
WHITE BOX TESTING
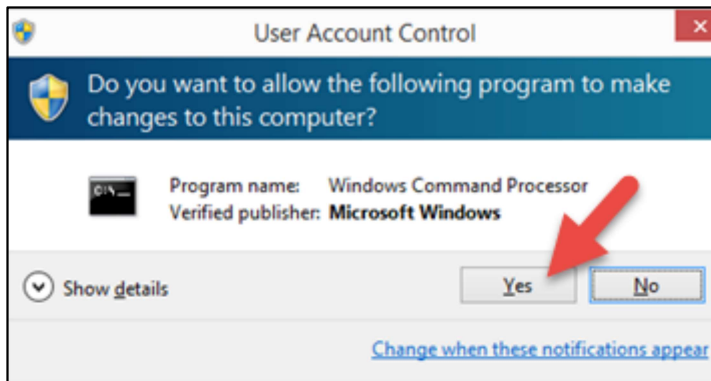## What Every Tester Should Know

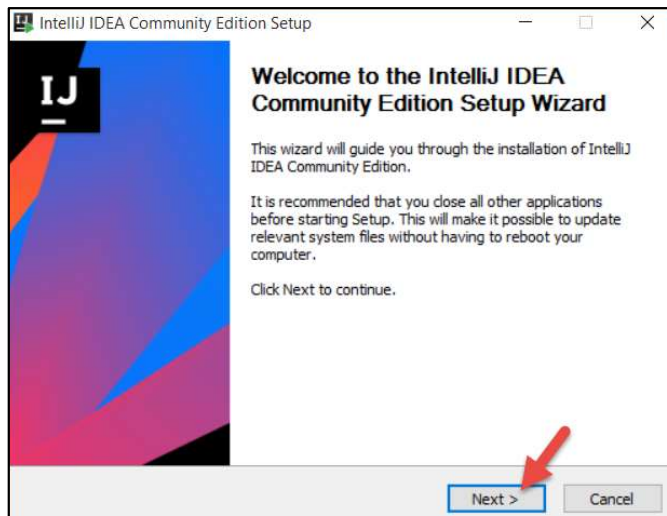## Installing IntelliJ IDEA Community Edition

Setup > Windows

1. **Double click the file IntelliJ.exe to run the installer.**

2. **If User Account Control is enabled, a pop-up will appear, click Yes to allow Windows to run the installer.**
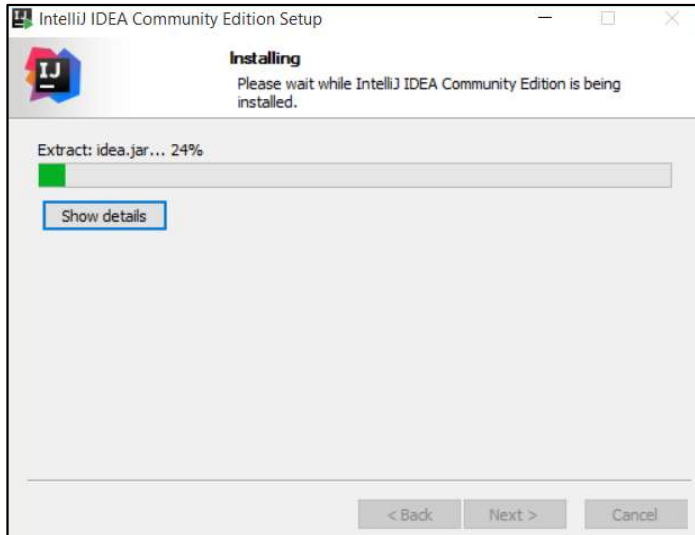

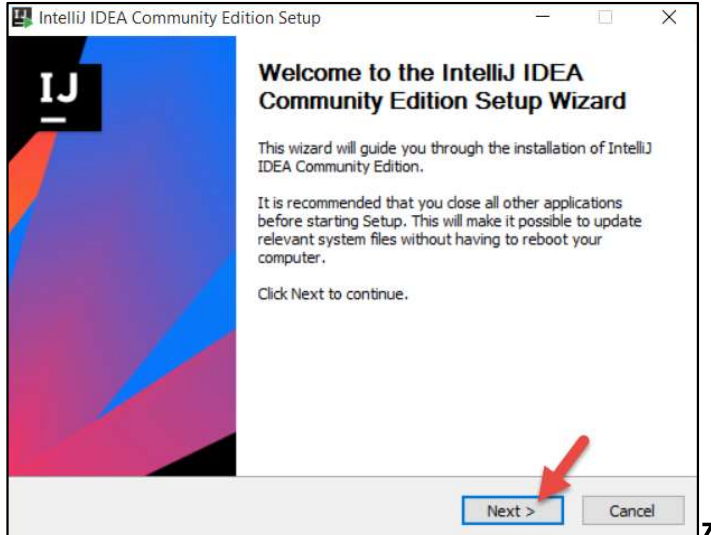3. **Proceed through the installation wizard by clicking Next.**

4. **Keep clicking Next through the wizard until the installation starts.**

5. **Once the installer you can run IntelliJ by checking the checkbox and clicking Finish.**

Installing Java Development Kit Community Edition

Setup > Windows

1. Depending on your CPU architecture, install JDK8-32Bit or JDK8-64Bit.
   If you are unsure use the 32bit version of the installer.

2. Once the Java setup window appears, click Next.

3. Keep clicking Next through the wizard until the installation starts.
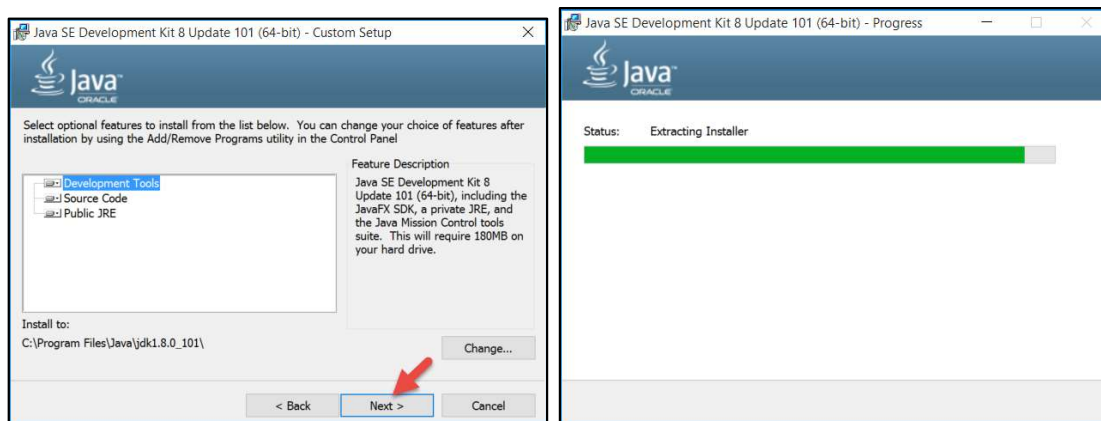
4. When the Java Setup – Destination Folder window appears, ensure the installation path is as below, then click Next.
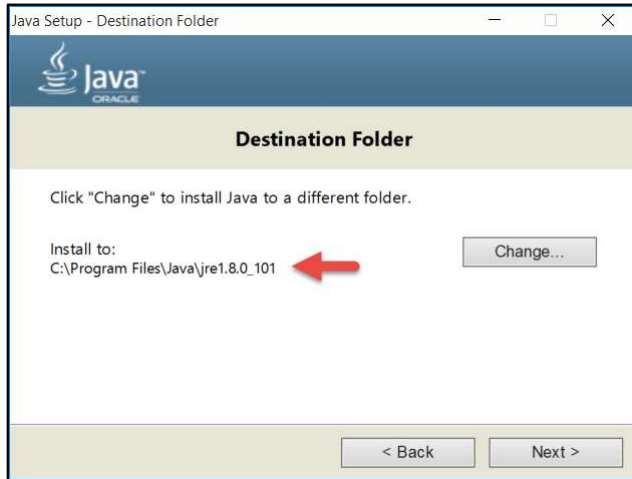
5. Once the installation completes, click Close to exit the installation wizard.
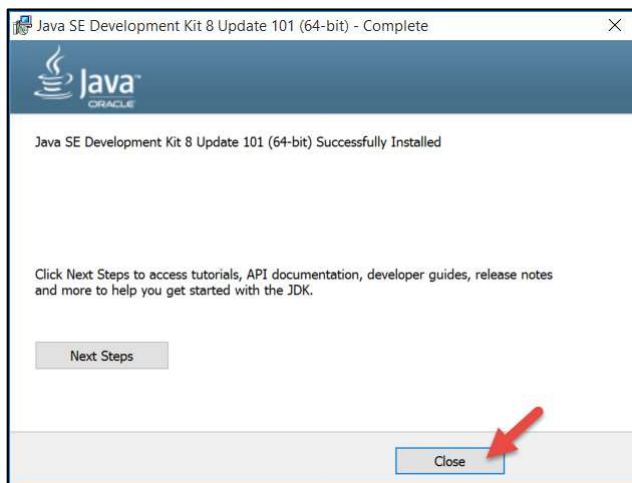
WHITE BOX TESTING
What Every Tester Should Know

Importing the Tutorial Code Project

1. **Launch IntelliJ IDEA from your start menu.**
   Choose your preferred import settings and click OK.
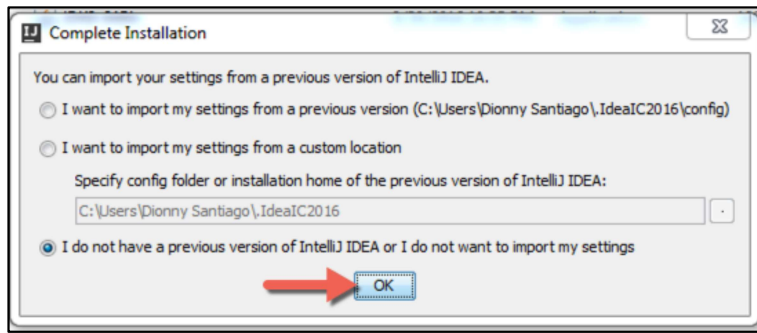
2. **On the Customize IntelliJ IDEA screen, select Skip All and Set Defaults.**
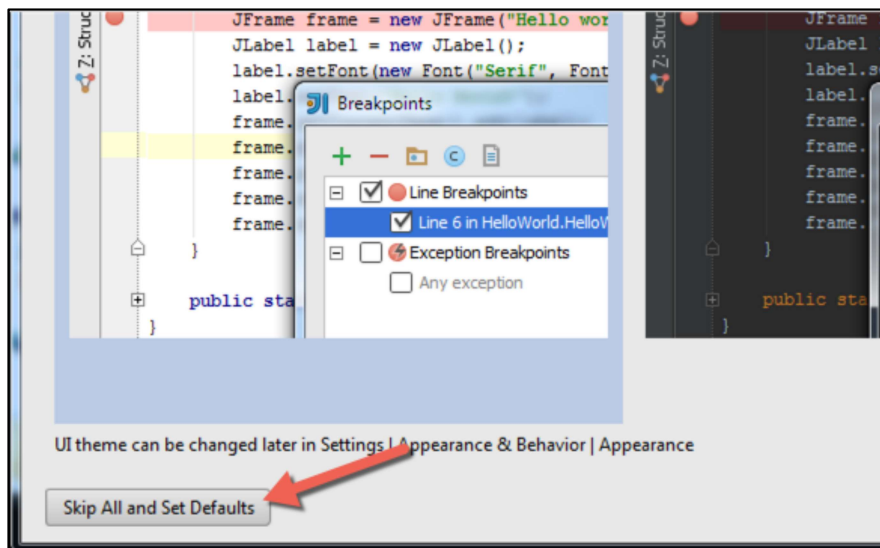
3. **From the welcome screen, click on Configure → Project Defaults → Project Structure.**
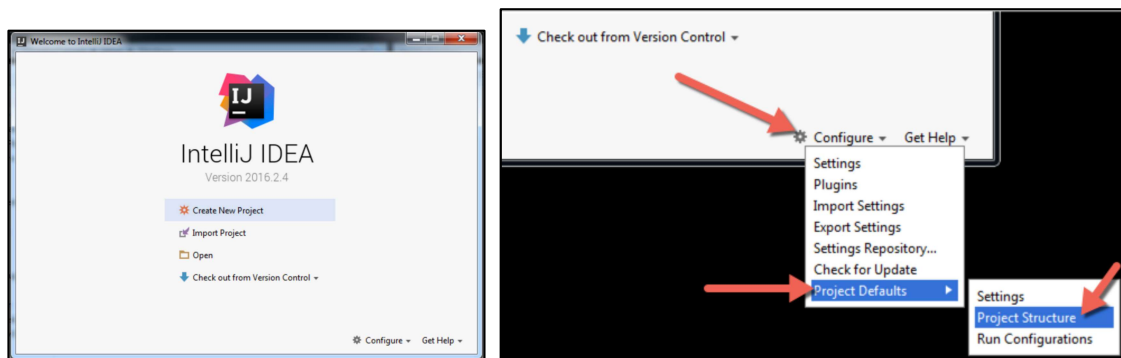
4. **On the Default Project Structure window, under Project SDK, click on New…, then on JDK.**
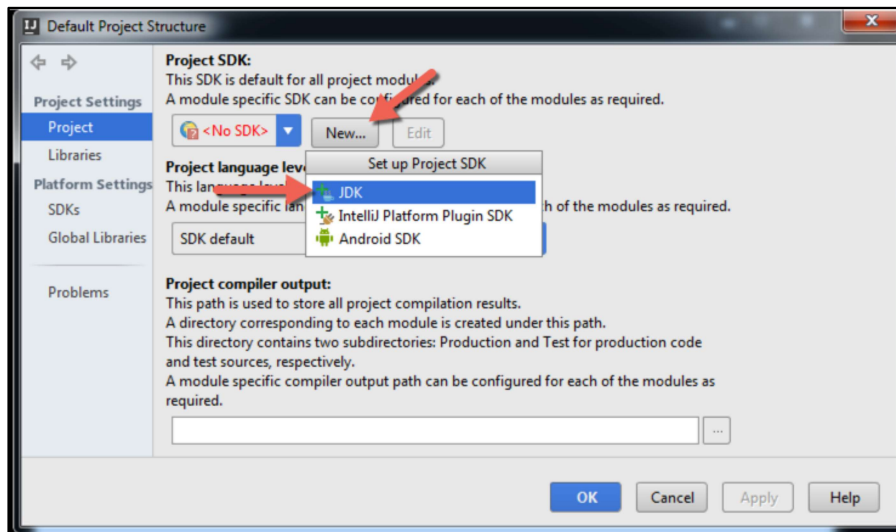
5. **Click on the jdk1.8.0 folder under Program Files → Java, then click OK.**
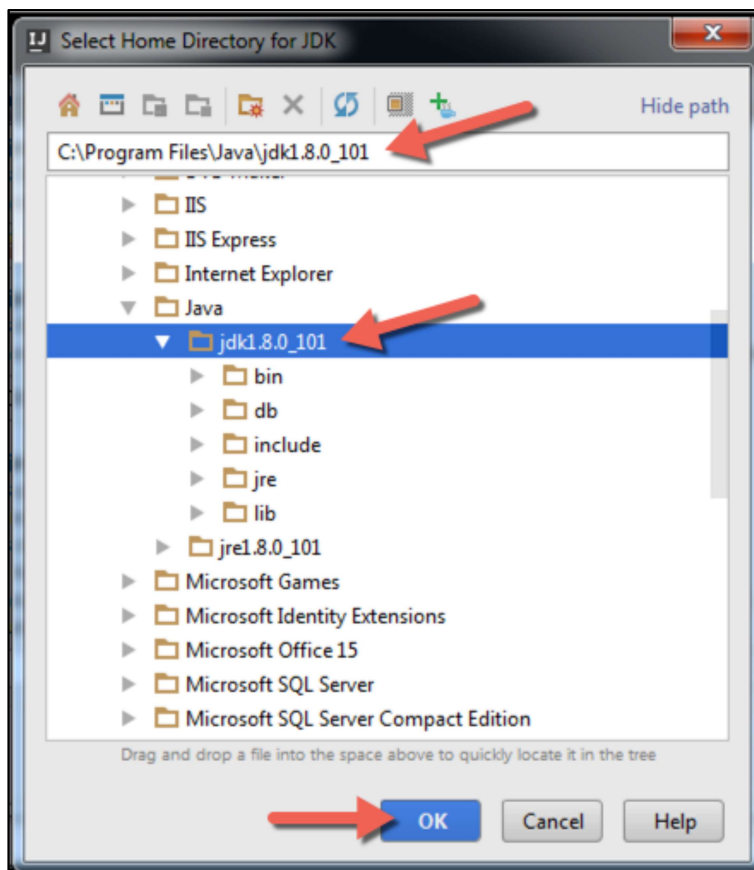
6. On the Default Project Structure window, confirm the Project SDK is set to 1.8, and the Project language level is set to SDK default (8 – Lambdas, type annotations etc.), then click OK.
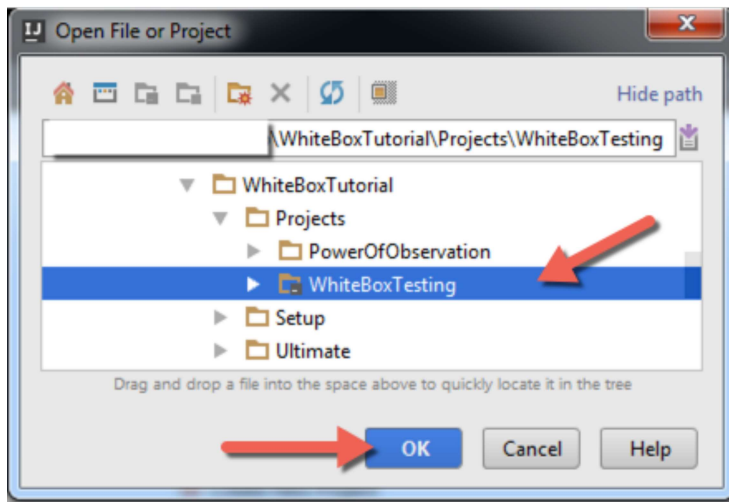

7. Now we are ready to import the project. On the IntelliJ welcome window, click Open.
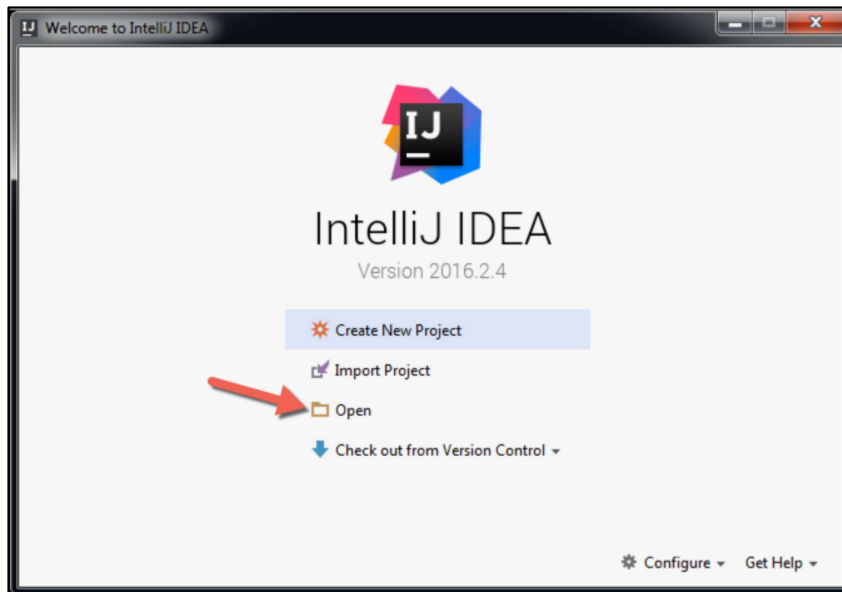
8. Navigate to the location of the WhiteBoxTutorial folder (available on the provided USB drive), then select Projects → WhiteBoxTesting and click OK.

Installing IntelliJ IDEA Community Edition

Setup > Macintosh

1. **Double click the file IntelliJ.dmg to run the installer.**
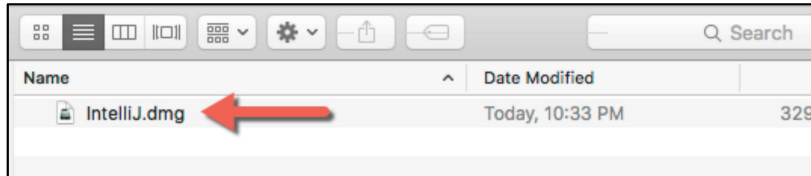
2. **Once the IntelliJ IDEA CE window appears, drag the IntelliJ IDEA CE icon to the Applications shortcut folder.**
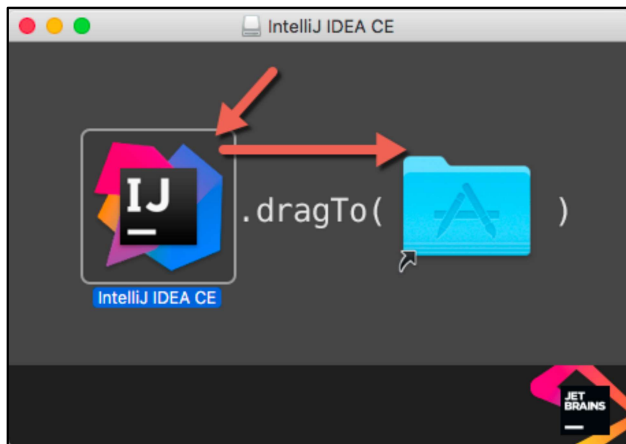
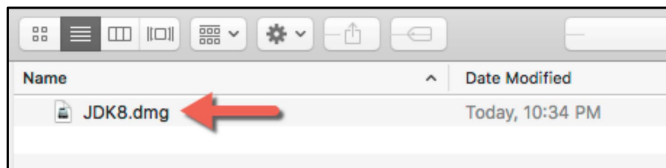3. **The installation will now start. Once it is finished, close the installation window.**
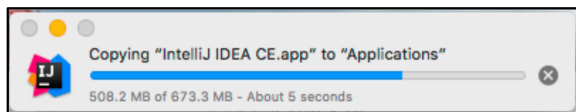
WHITE BOX TESTING
What Every Tester Should Know

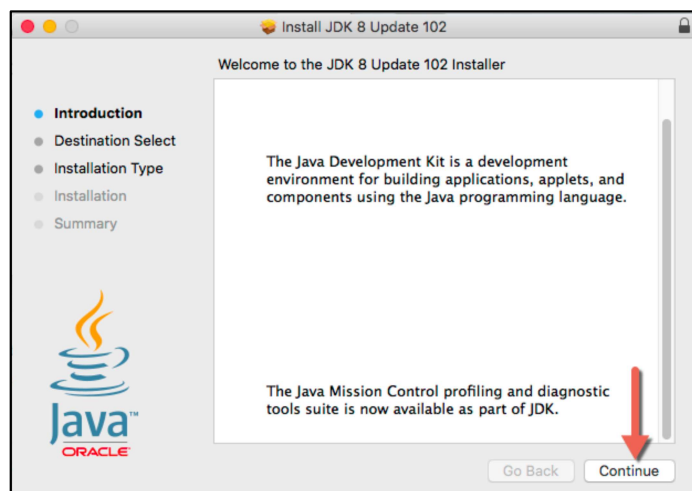## Installing Java Development Kit Community Edition

<u>Setup > Macintosh</u>

1. Double click the file JDK8.dmg to run the installer.

2. Once the JDK 8 Update 102 window appears, double click the JDK 8 Update 102.pkg to begin the installation.
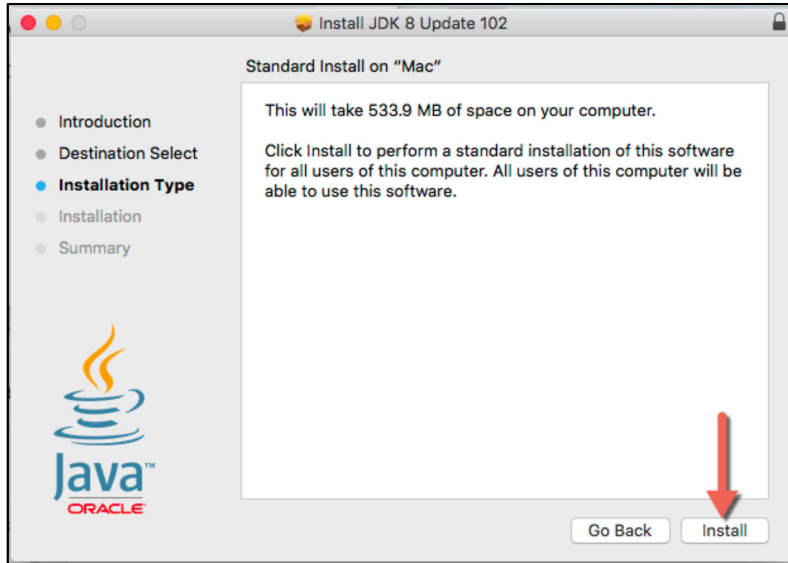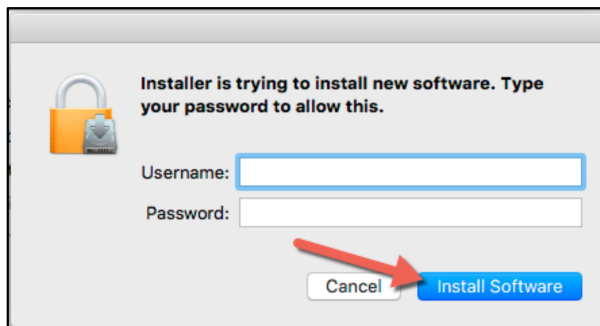

3. Click Continue on the installer window.

4. **Click Install on the installer window.**

5. **If prompted for your login credentials, input your username/password and click Install Software.**
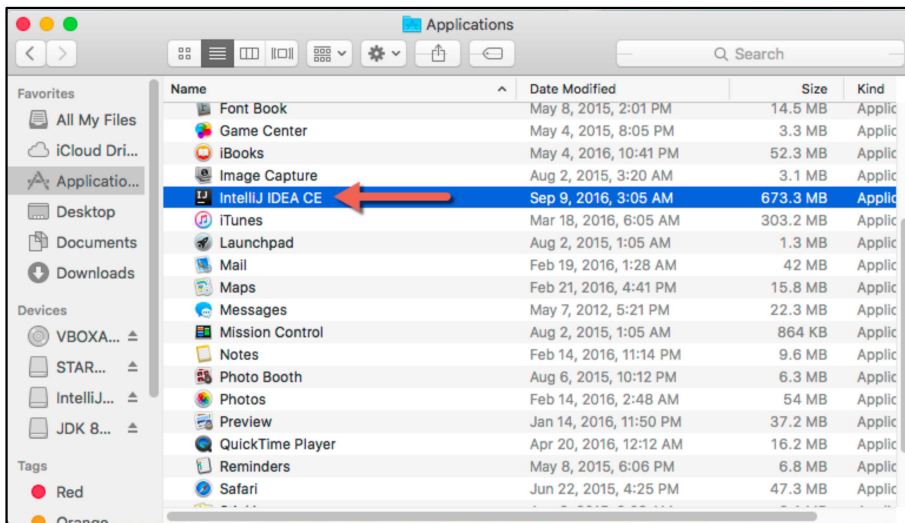
6. Once the installation completes, click Close on the installation window.
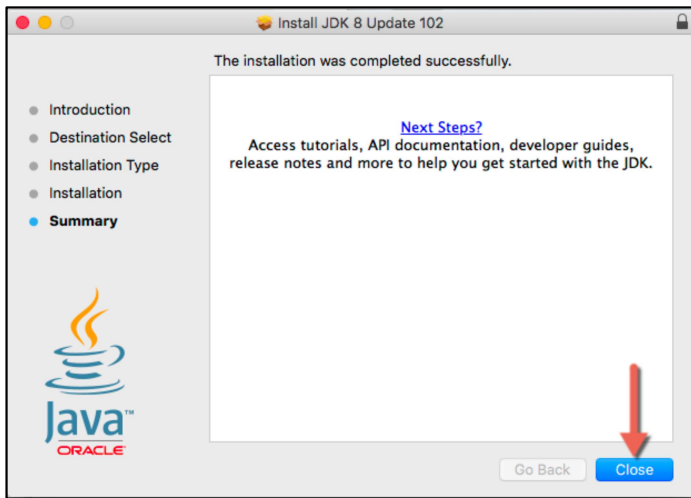
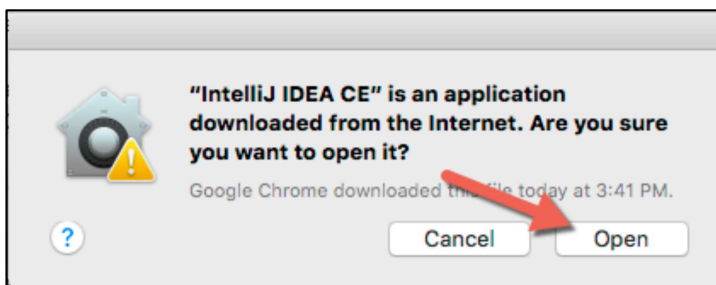## Importing the Tutorial Code Project

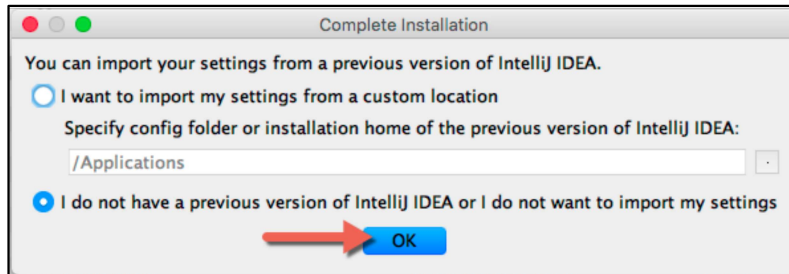1. Launch IntelliJ IDEA from your Applications folder.
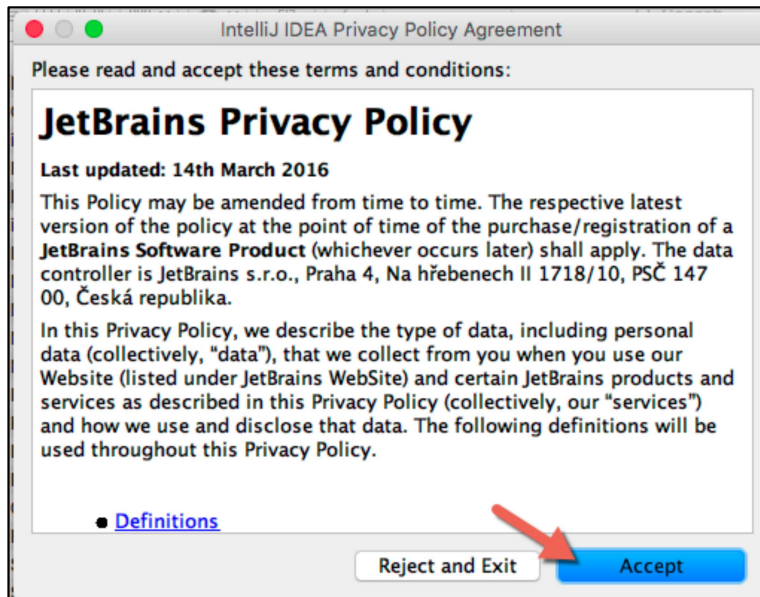

2. Click Open on the security prompt.

3. Choose your preferred import settings and click OK.

4. Accept the JetBrains Privacy Policy.

5. On the Customize IntelliJ IDEA screen, select Skip All and Set Defaults.
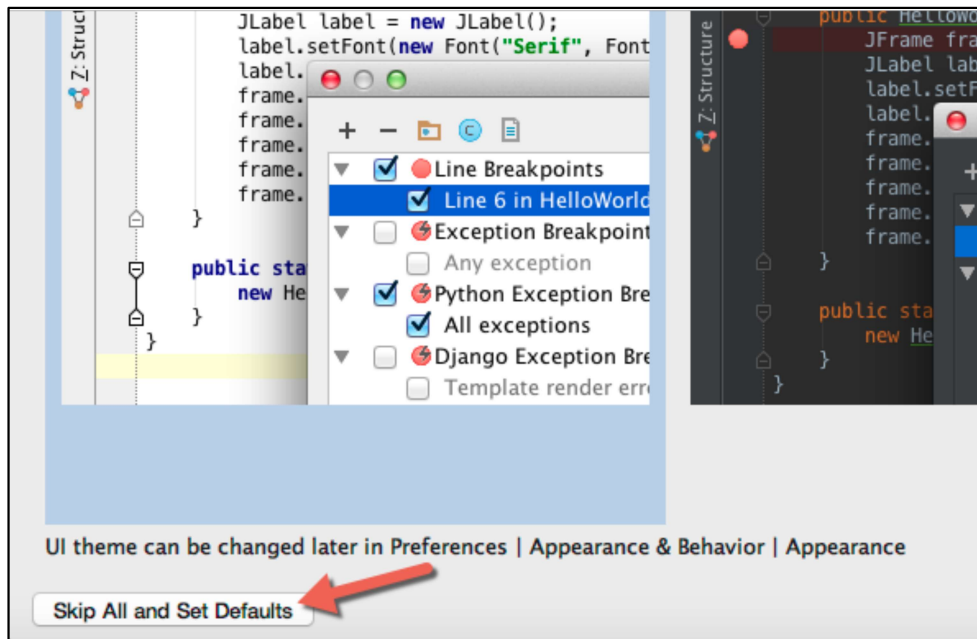
6. From the welcome screen, click on Configure → Project Defaults → Project Structure.
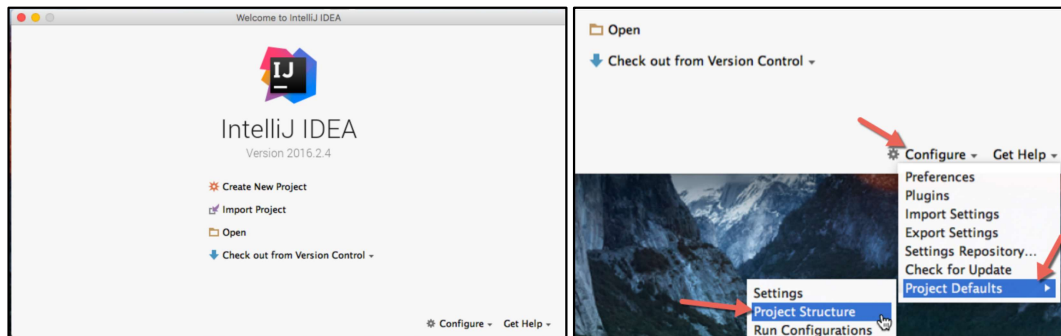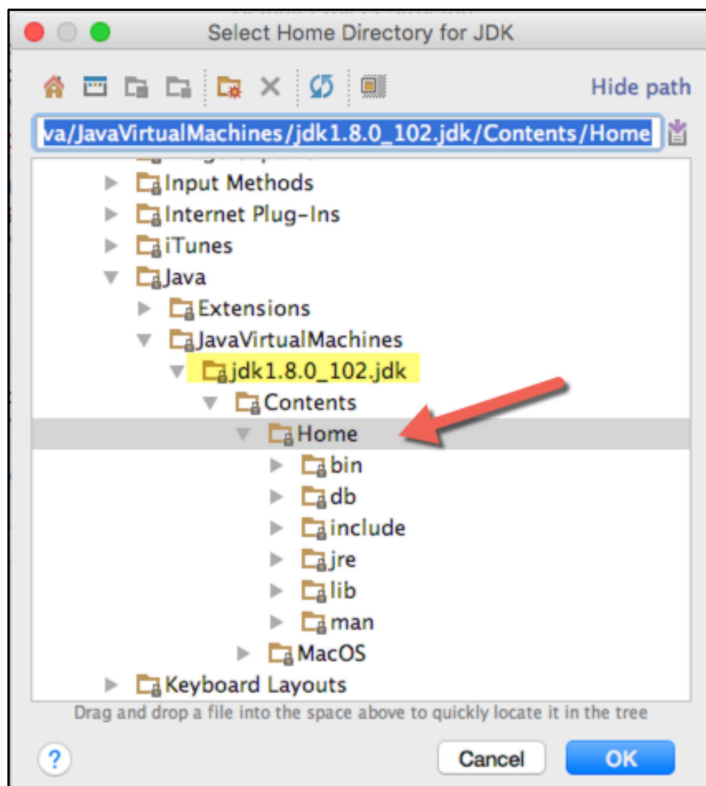
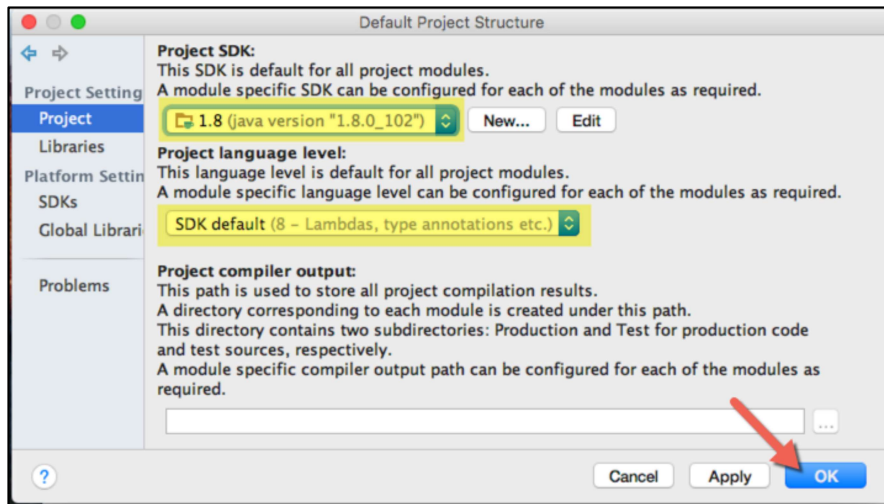7. On the Default Project Structure window, under Project SDK, click on New…, then on JDK.
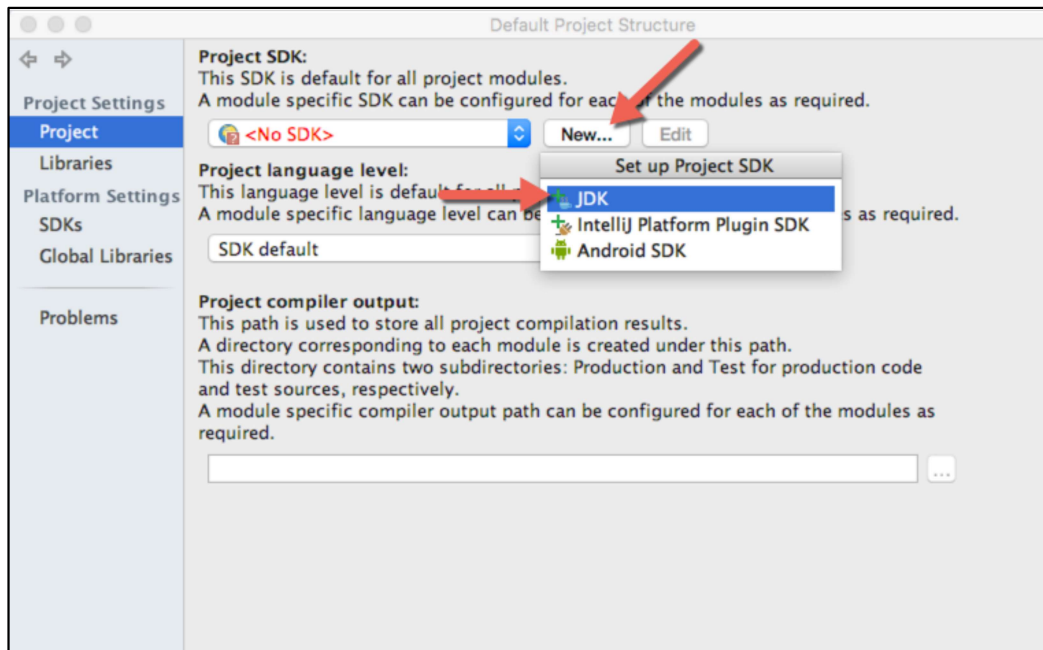
8. Locate the jdk1.8.0 folder under Java → JavaVirtualMachines, and choose the Home folder under Contents, then click OK.

9. On the Default Project Structure window, confirm the Project SDK is set to 1.8, and the Project language level is set to SDK default (8 – Lambdas, type annotations etc.), then click OK.

10. Now we are ready to import the project. On the IntelliJ welcome window, click Open.

11. Navigate to the location of the WhiteBoxTutorial folder (available on the provided USB drive), then select Projects → WhiteBoxTesting and click OK.