

Eclipse Instructions

By Mr. Detrick

Opening Eclipse the first time in school on an HP

Go to the start menu at the lower left hand corner

Select Computer >>

Local Drive C >>

Eclipse folder (might be two inside each other)

Before opening, drag the purple circular icon to pin it to the bottom of the screen

Now open Eclipse from this icon at the bottom of the screen.

Your workspace will be saved on the server at H:\workplace

You should be able to access this from any computer in the school, but can only run it if the computer has Eclipse installed (most don't)

Click Ok, then click on the icon for WorkBench. This is where we will be doing most of our programs this year.

Creating new projects and programs

Programs will be inside of packages which will be inside of projects.

Create a new project

File >> New Java Project

>> name the project such as "Notes" or "Marking Period 1"

At this point we will **normally click finished**, but if you need to add a jar file, such as Karel J Robot.

Adding jar files when setting up the project (Specifically Karel J Robot)

Click next instead of finish

>> select the Library tab

>>Click button "Add External Jar"

>> find the "computer"

>> shared" folder on the network

>> Computer Science 2012

>> cs files only >> Karel J. Robot

>> KarelJRobot.jar

>> click Open

>>click Finished

Adding jar files to already created projects:

Click on the Project in the Package Explorer
Project tab at the top>>Properties>>Java Build Path
Same as steps above for adding project to new Project (Library>>...)

Create a package

Click on the project on the left side that you just created.
File >> new package >> name the package with a lowercase letters>>finished
With Karel J Robot, the name of the package must be kareltherobot.
All other packages should be named in a way so that it is helpful to find programs later.

Create a new class

Click on your package on the left hand side (Package Explorer)
- this step is often forgotten
File >> new class
>> The class must start with a capital letter and should pertain to the what the class does
Make sure to check the box for public static void main at the bottom, but none of the others.
>>finish

You are now ready to type your program

Changing Indentation and Braces

My preference for braces doesn't match the default in Eclipse, so here is how to change them. Start by clicking on the project you want to change in the Package Explorer

Project >>
Properties >>
Java Code Style (hit the arrow to the left)>>
Formatter

>>Click on "Configure Workspace Settings" in the upper right corner
>>Click on "Edit"
>>Click on the "Braces" tab >>Change all the positions to "next line"
>>Rename the profile name (any name is fine)
>>Click "OK"
>>Click "Apply"

Now all programs you create in this project should put the braces on a new line and line them up under each other.

Setting comments at the top of programs

Once a project is created, click on the project and follow the direction below:

In the menu go to:

```
Project >>  
  Properties >>  
    Java Code Style (hit the arrow to the left)>>  
      Code Template>>
```

Click on “Configure Workspace Settings” in the upper right corner

```
Code (hit the arrow to the left)>>  
  New Java Files>>  
    Edit...>>
```

Do not erase any of the code that is there. There should be two lines of code, a space and two more lines of code. In that space type in the following code to your specifications:

```
/*  
Your Name  
${date}           -from insert variable  
Period ____  
Description  
*/
```

Click on “OK”
Click on “Apply”
Click on “OK”
Click on “OK”

Now when you make a new class in that project, the above comment should come up automatically.

Transferring to a flashdrive

```
Find your server file (H:) under start>>computer  
Find workplace>>project name >> src >> whatever.java  
>>Organized >>copy >> find your flashdrive and paste
```

Important: Copy **the .java files, not the .class files**

Coding Style

Braces:

Opening and closing braces should line up vertically with the beginning letter of the line it is under. Each brace should be on a line by itself. Whatever is inside the braces should be indented. Make sure the program lines up appropriately.
See above for setting up braces in Eclipse

```
public void doSomething()
{
    System.out.println("Something");
}
```

Commenting:

Commenting is done to help explain what is happening in the program. Comments should be clear and concise.

Each class should have comments at the top explaining what the class does.

Each method should have comments at the top summarizing the method. Pre-conditions and post-conditions would be helpful, but not required in my class.

Throughout the program, there should be comments explaining code that might be difficult to understand and to help others follow the code.

Naming:

All variables should have names that clearly explain their purpose. When possible, variables should be defined at the top of the method or class in which they are to be used. I do not like variables declared on the fly except when used to count for loops.

Constants should be all capital letters with underscores between words

CLASS_CAPACITY or **AGE**

Class names should be nouns that have the first letter capitalized and the first letter of each word thereafter capitalized. There should be no spaces.

SummerVacation or **Trial** or **MyFirstProgram**

Methods and other variable names should NOT capitalize the first letter, but the first letter of each word thereafter should be capitalized. No spaces. Be sure to use the prefix get and set for getters and setters.

getName or **numberOfStudents** or **count**

Programming

1. Programs should be efficient. Avoid repeated code.
2. Keep classes and methods small
3. Explicitly initialize variables. Do not rely on default values
4. Clarify the order of operations with parenthesis
5. Be sure to write the program to specifications. No more, no less.
6. Programs should be easy to read and follow the logic
7. When running the program, check that it works for numerous scenarios before handing it in (Robustness)