Ways to run java code

Repository:

https://github.com/JasonWang19/cacc_java/blob/master/

Week2 repository:

https://github.com/JasonWang19/cacc_java/tree/master/week2

1. Command line

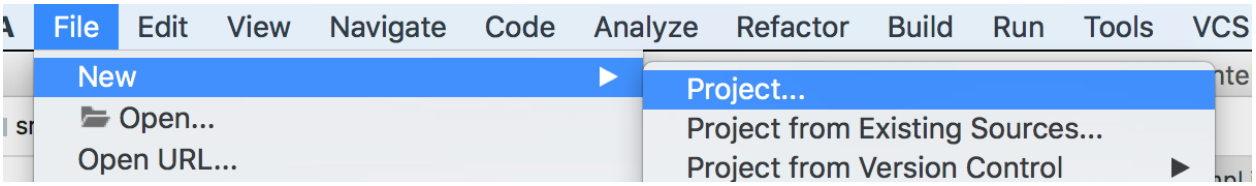
Create a local file, and type the HelloWorld.java with a text editor

In command line type

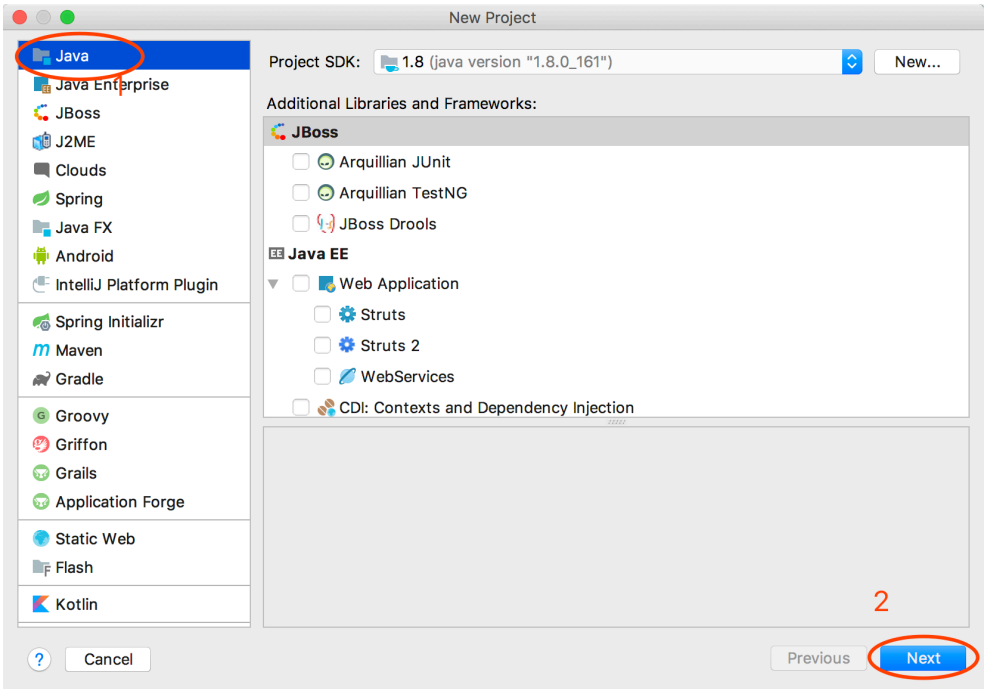
- javac HelloWorld.java
- java HelloWorld

2. Intellij or other IDE

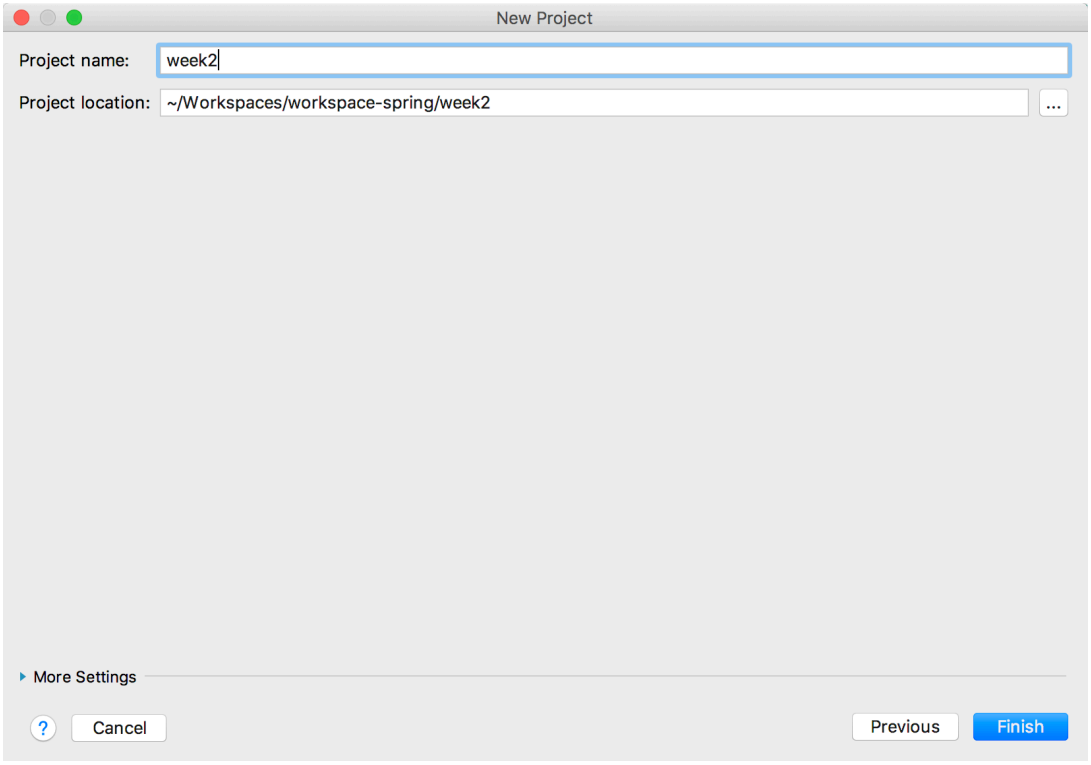
1) Choose create new project



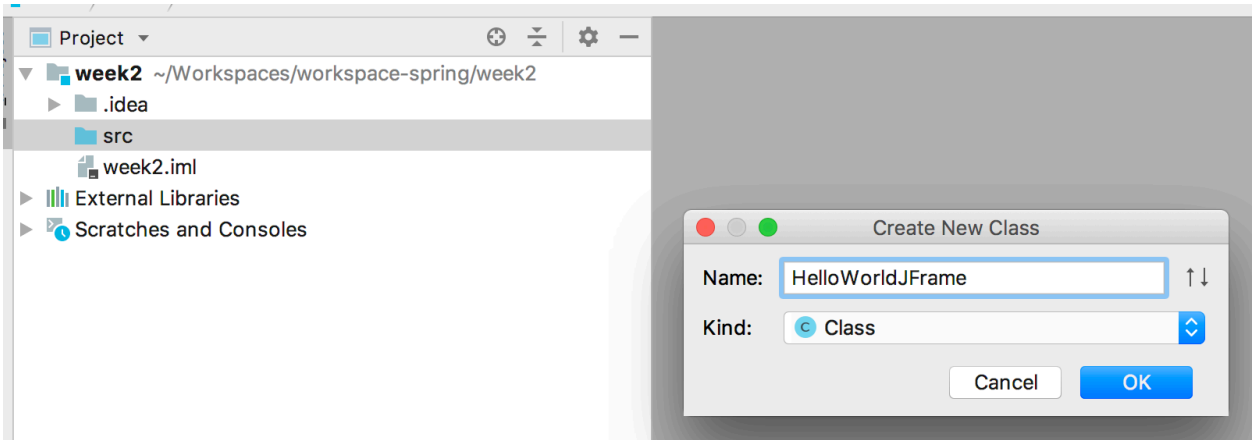
2) Create a java project



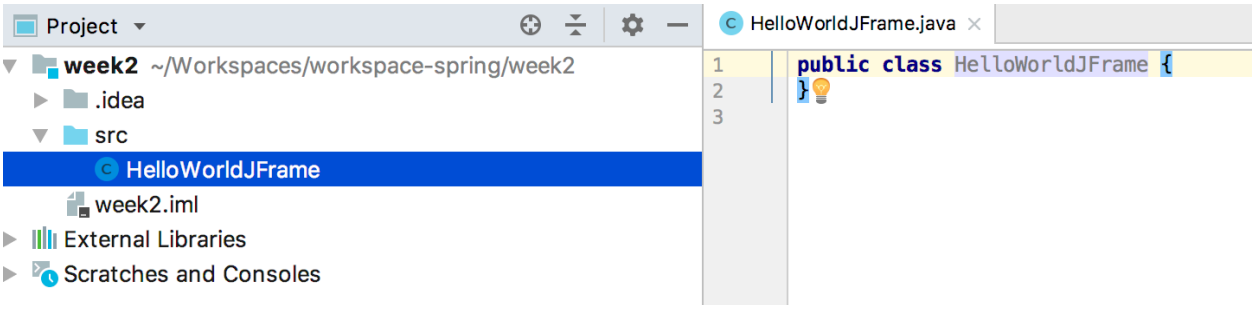
3) Finish creating the project, pick a name e.g. week2



4) Create a java class



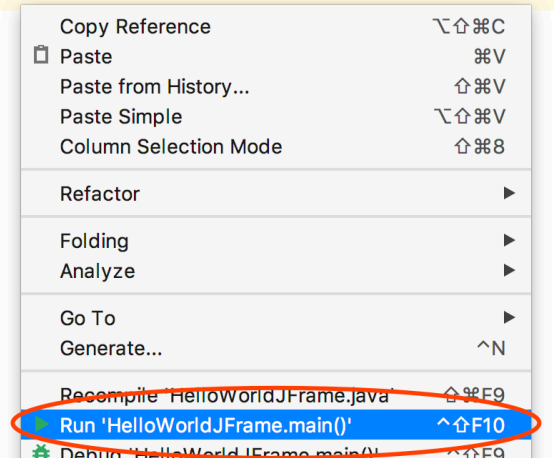
5) Now a new class is created, you can type in or copy the content of HelloWorldJFrame.java to editor.



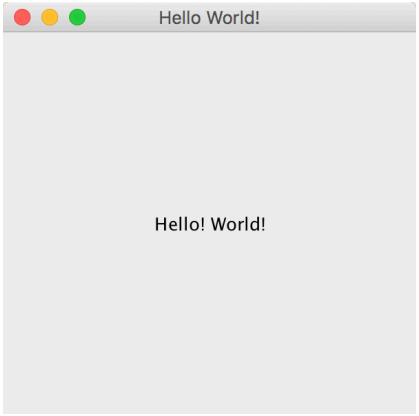
6) Run code

```java
import javax.swing.*;

public class HelloWorldJFrame {
    public static void main( String[] args ) {
        JFrame frame = new JFrame( title: "Hello World!" );
        JLabel label = new JLabel( text: "Hello! World!", JLabel.CENTER );
        frame.add(label);
        frame.setSize( width: 300, height: 300 );
        frame.setVisible( true );
    }
}
```



7) Output



Comments

Block comment

/* and */

e.g.

```
/* This is a
   multiline

comment. */
```

Line comment

line comments indicated by //

e.g.

```
// This is a single-line comment
   // and so // is this
```

Data types

Primitive Types

Numbers, characters, and Boolean values are fundamental elements in Java.
For those situations where it's desirable to treat a primitive value as an object, Java provides "wrapper" classes.
The major advantage of treating primitive values as special is that the Java compiler and runtime can more readily optimize their implementation.
An important portability feature of Java is that primitive types are precisely defined. For example, you never have to worry about the size of an `int` on a particular platform; it's always a 32-bit, signed, two's complement number.

- boolean true or false
- char 16-bit, Unicode character
- byte 8-bit, signed, two's complement integer
- short 16-bit, signed, two's complement integer
- int 32-bit, signed, two's complement integer
- long 64-bit, signed, two's complement integer float 32-bit, IEEE 754,
- floating-point value
- double 64-bit, IEEE 754

two's complement

To get the two's complement negative notation of an integer, you write out the number in binary. You then invert the digits, and add one to the result.

Variable declaration and initialization

```
int foo;
double d1, d2;
boolean isFun;
```

Variables can optionally be initialized with an expression of the appropriate type when they are declared:

```
int foo = 42;
double d1 = 3.14, d2 = 2 * 3.14;
boolean isFun = true;
```