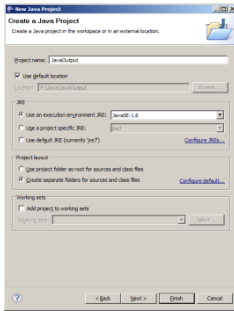


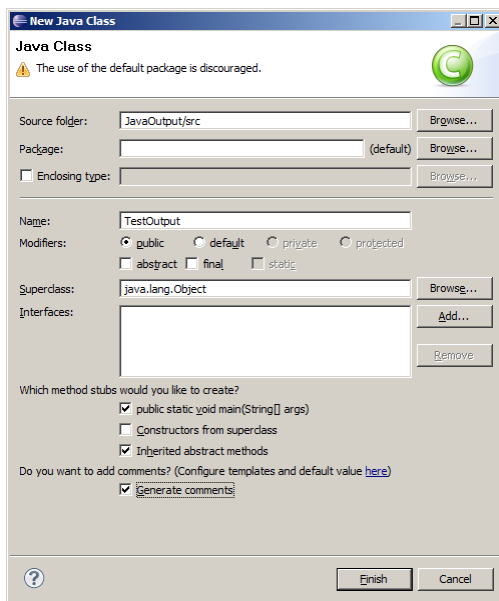
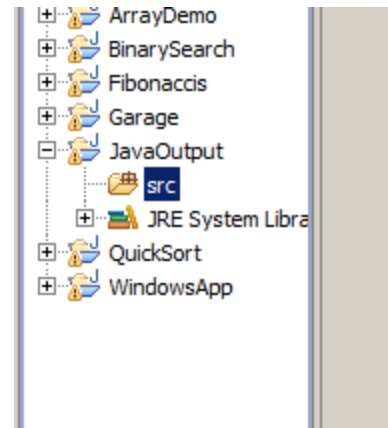
# Creating Java Projects with Eclipse

Eclipse is an IDE (Integrated Development Environment) designed to allow code to be created in several languages. Java is one of its key uses, other supported programming languages include Android, C++ and Python.

There are several versions of Eclipse based on what facilities they offer and the date of the build. These are generally similar but the Window and menu system will probably differ depending on the build.



We need to tell Eclipse to create a Java project. This will create a directory system including an empty directory (src) where the Java files are going to be.



One Java source file is enough to start with. Right click on the src directory and add a Java class.

Make sure that the check box concerning public static void main is checked.

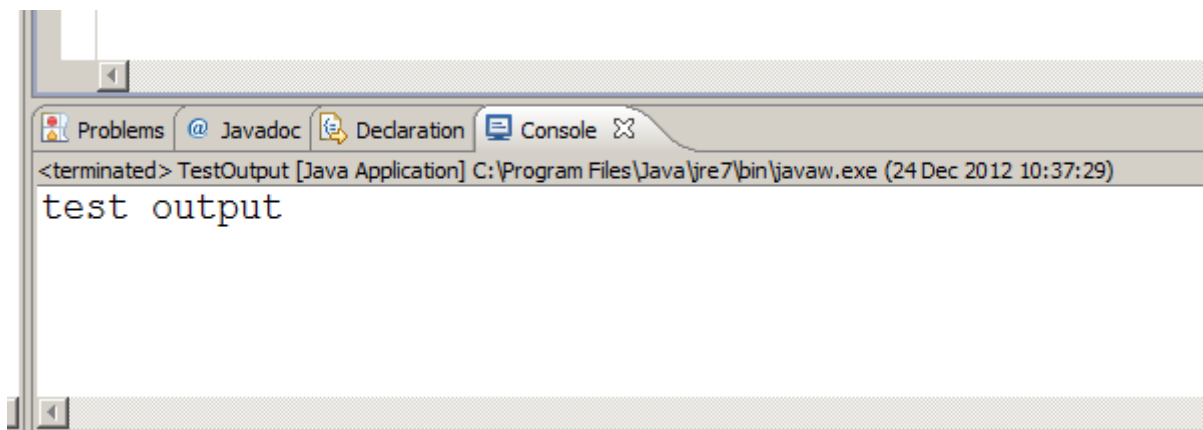
The following default code will be created

```
public class TestOutput {  
  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
    }  
}
```

Here is a simple line to check the code will run:

```
public class TestOutput {  
  
    public static void main(String[] args) {  
        System.out.println("test output");  
  
    }  
  
}
```

The output will show up in the console window



This code demonstrates output of some variables.

```
int num = 22;  
  
String word = "looking good";  
System.out.println("test output " + word + " " + num);
```

Here is a Java method. Note that it is outside of main but still inside the Java class.

```
public static void main(String[] args) {  
    int num = 22;  
    System.out.println("num is " + num);  
    num = doubleMe(num);  
    System.out.println("num is now " + num);  
}  
  
private static int doubleMe(int number)  
{  
    number = number*2;  
    return number;  
}
```

Since Java 5.0 the Scanner class allows keyboard entry to be captured. Java.util is required to use this class so must be imported. The following code contains no error checking if data is entered that is not a valid int data type.

```
import java.util.Scanner;
public class TestOutput {

    public static void main(String[] args) {
        int num = 0;
        System.out.println("Please enter a number");
        Scanner s = new Scanner(System.in);
        num = s.nextShort();
        System.out.println("num is " + num);
        num = doubleMe(num);
        System.out.println("num is now " + num);
    }
    private static int doubleMe(int number)
    {
        number = number*2;
        return number;
    }
}
```