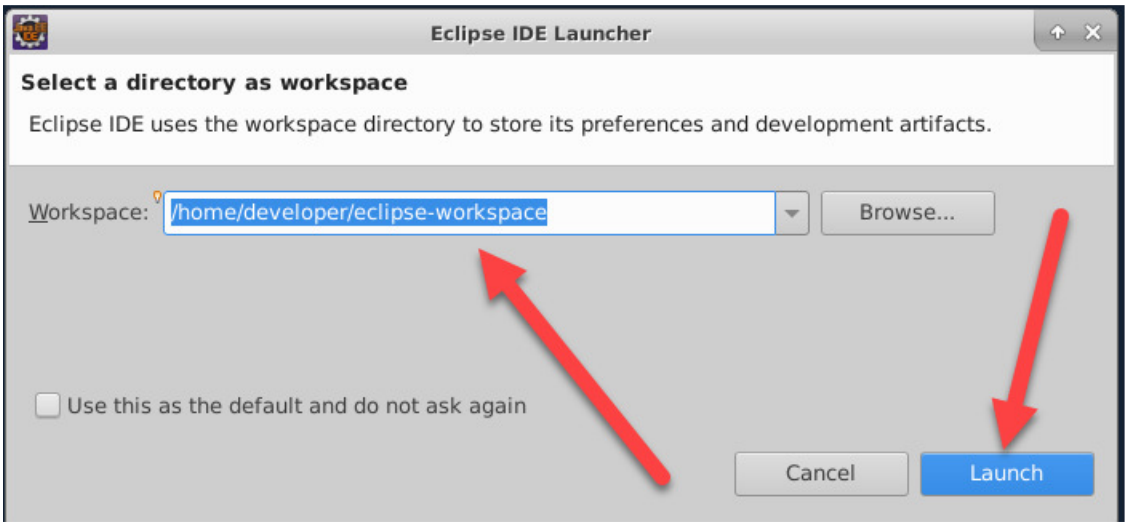
Start the lab

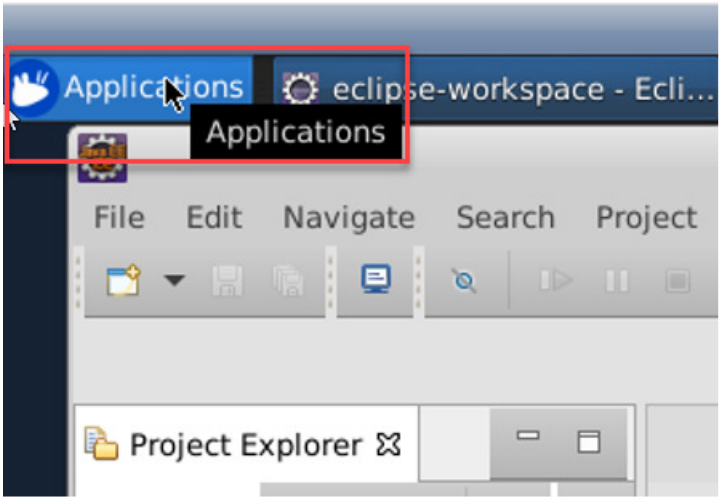
Lab 1: Java Console Applications

Let Eclipse start

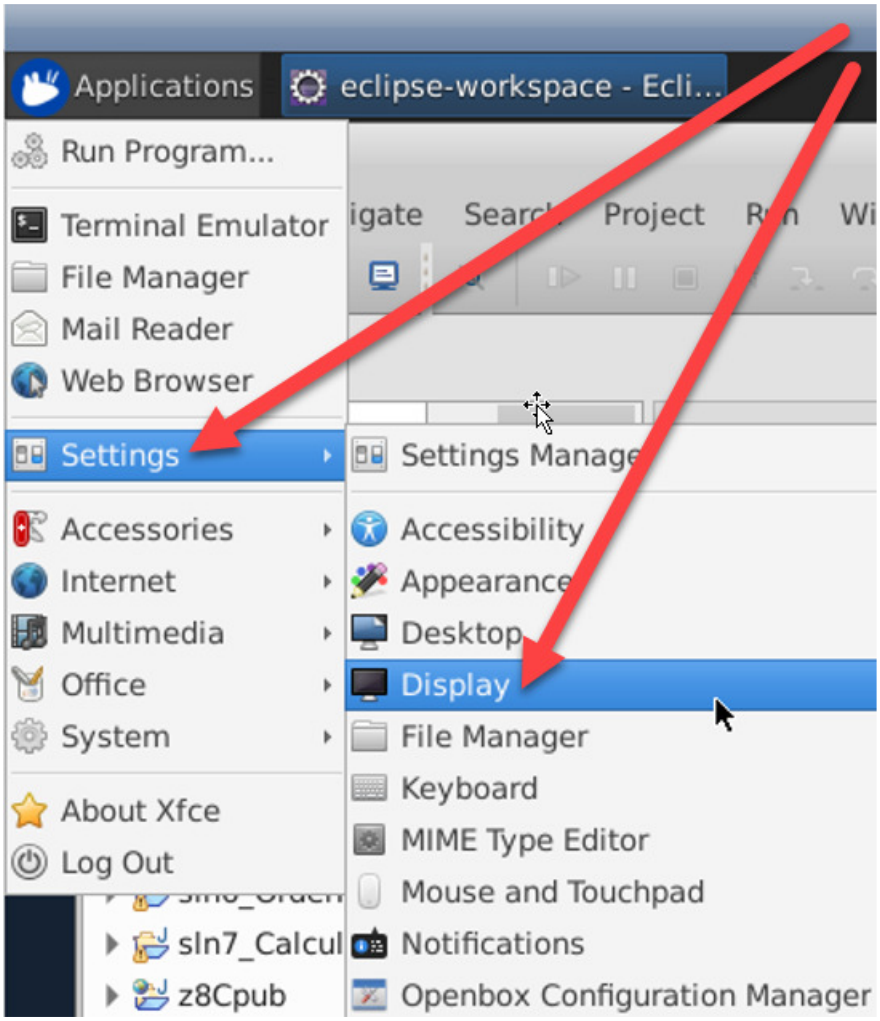
Leave the workspace unchanged, and click **Launch.**



In the upper left hand corner, click the **Applications** button.

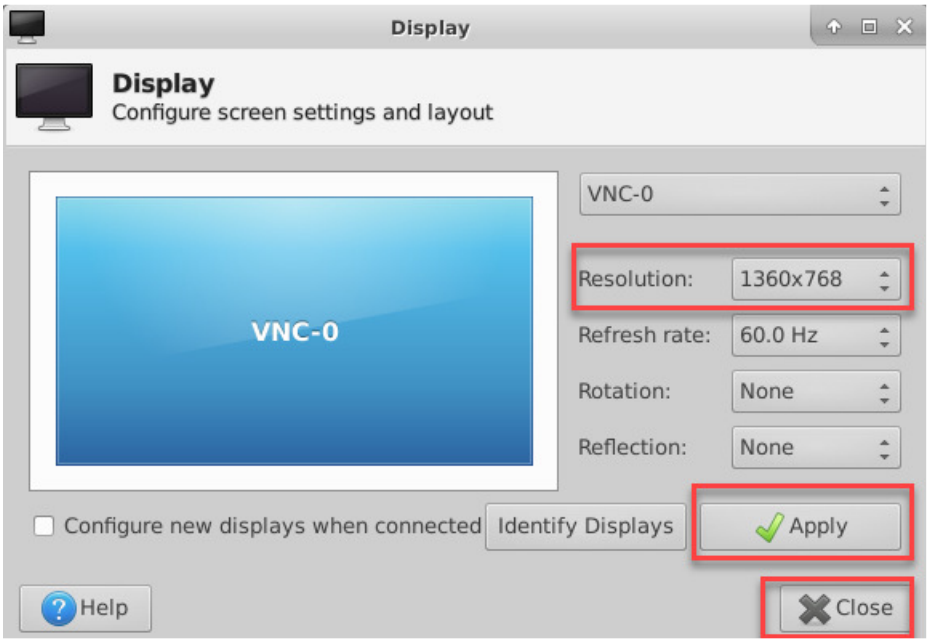


Click **Settings** -> **Display**

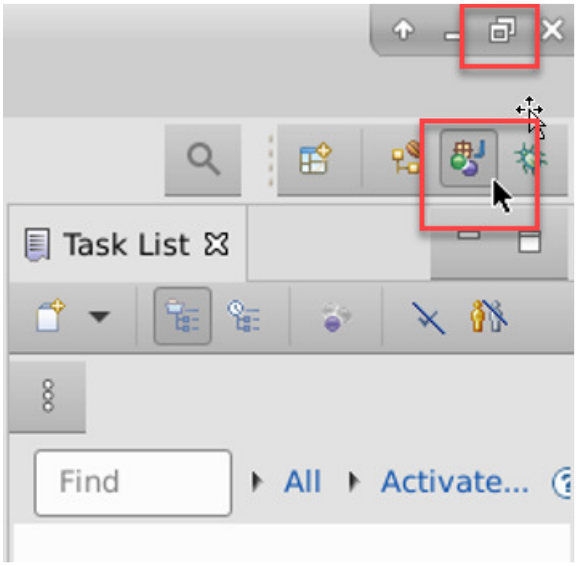


IMPORTANT: This example uses resolution **1360x768**. However, a different setting may work better for you.

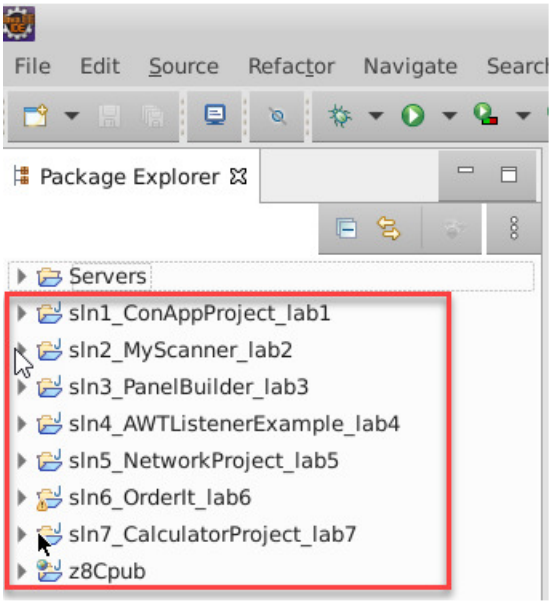
Click **Apply** and **Close**.



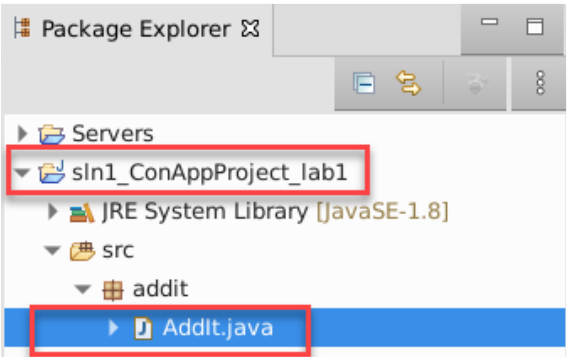
In the upper right corner, maximize **Eclipse**. Also, important, click the **Java Perspective button**. It must be the **Java perspective**.



Solution projects are listed in the Project Explorer. Key in all lines of code. However, if needed, use the solutions for troubleshooting.

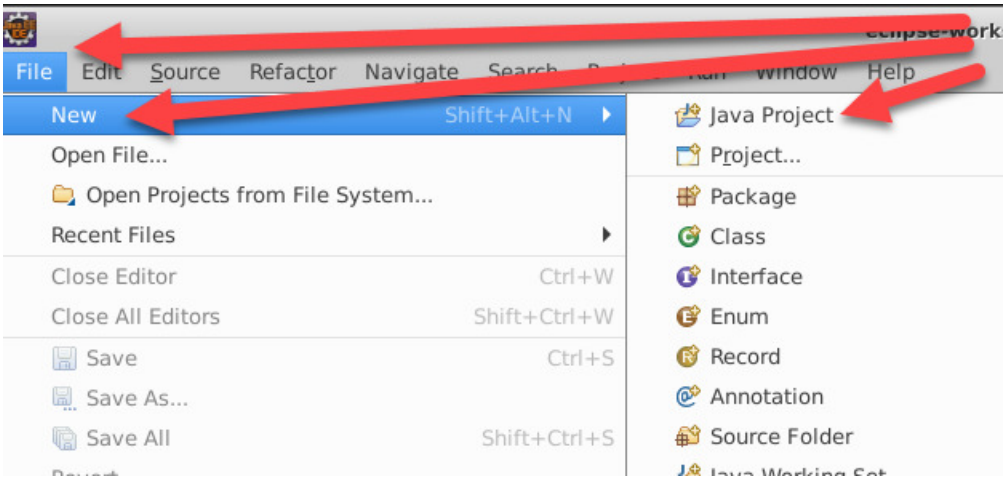


For this first lab, use solution project **sln1_ConAppProject_lab1** for reference. It is possible to copy and paste the Java code. However, avoid copy and paste as much as possible.

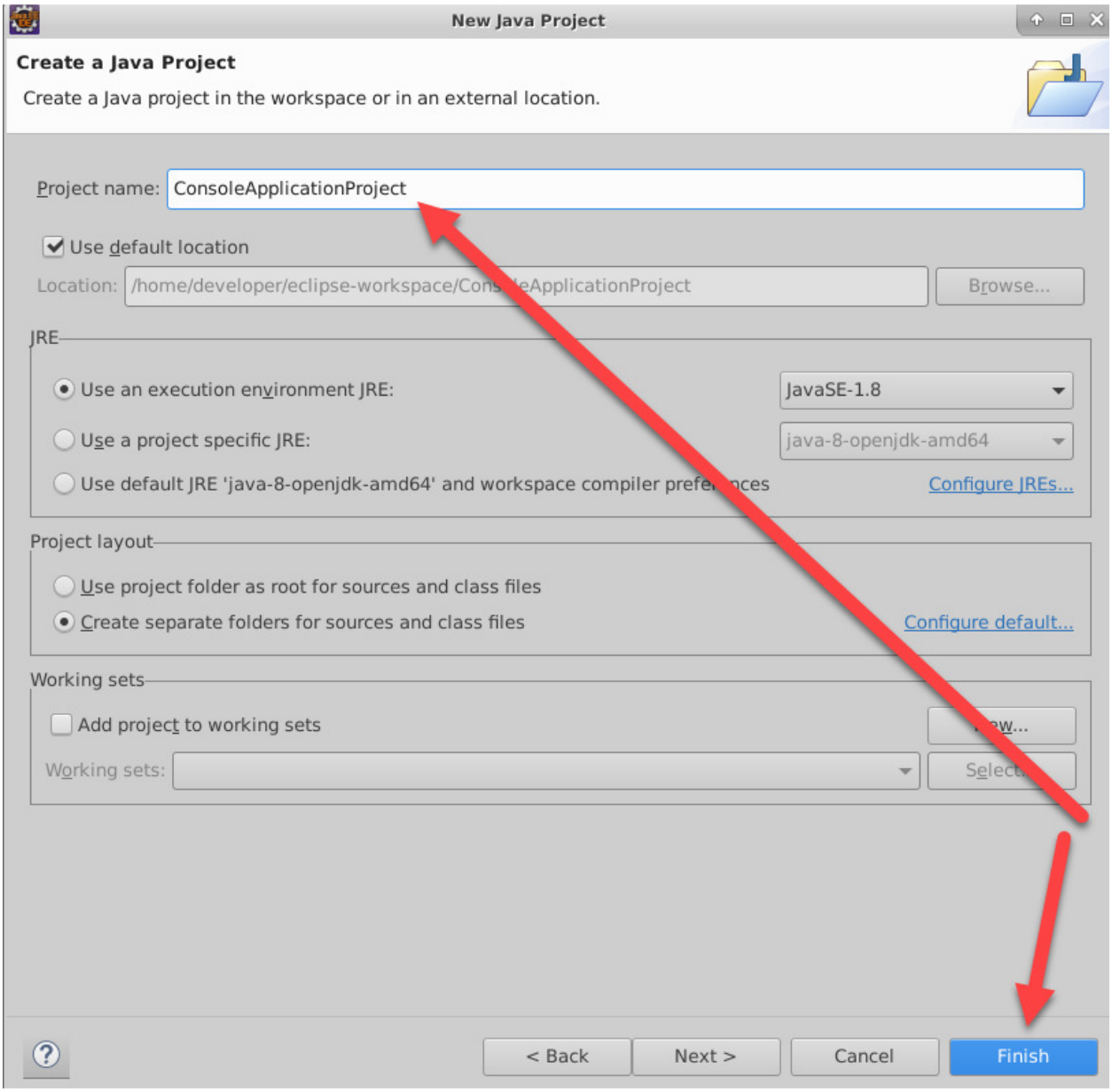


Create a new Java Project called **ConsoleApplicationProject**

Click the **File** menu -> **New** -> **Java Project**

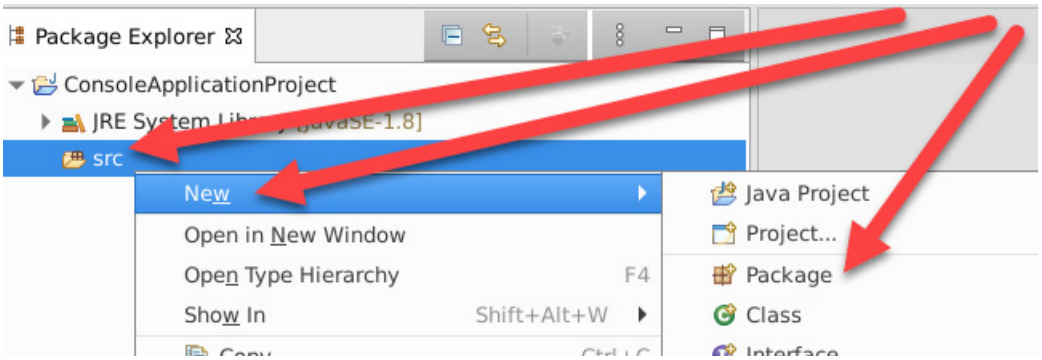


Name the project **ConsoleApplicationProject** and click **Finish**.



Create a package called addIt

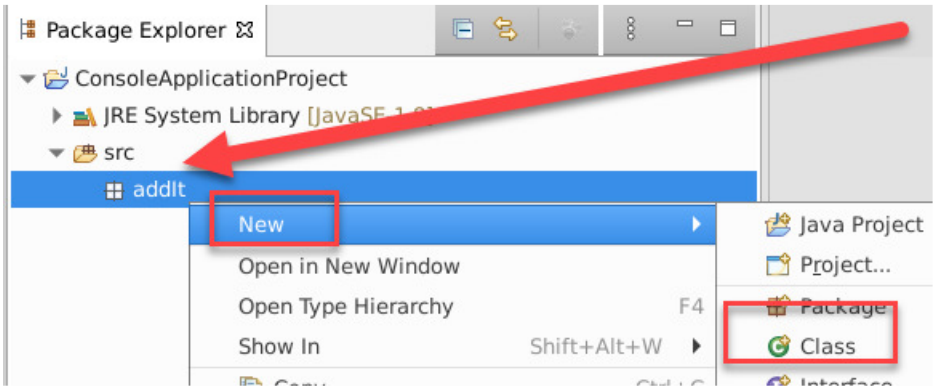
Expand the **ConsoleApplicationProject** and right click on the **src** folder, select **New** -> **Package**



When the Java Package dialog opens, enter:
Name: **addIt**

Click **Finish**

Right click on the **addIt** package and select, **New** -> **Class**

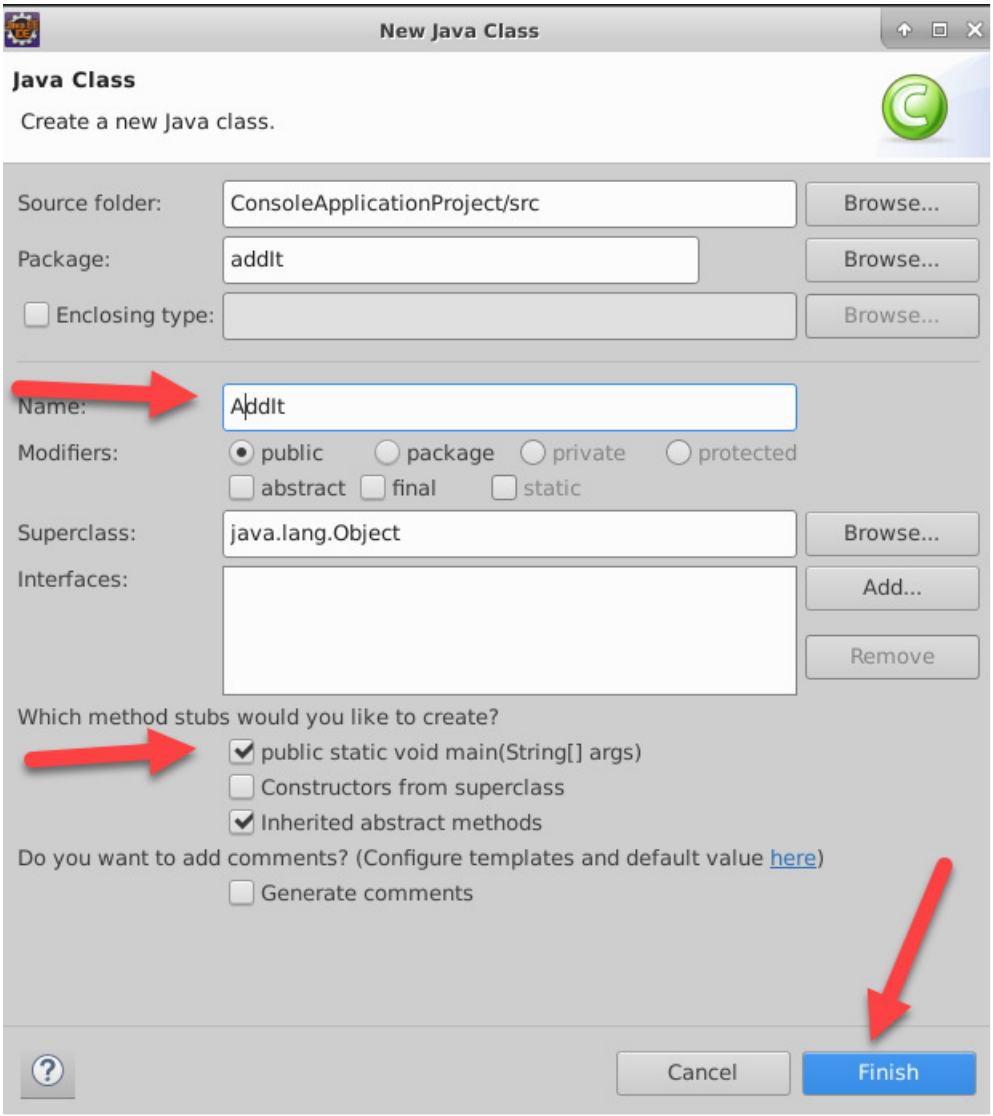


In the AddIt Java Class dialog, enter the following:

Name: **AddIt**

Check: **public static void main(String[] args)**

Click: **Finish**



Eclipse will open the AddIt.java class file.

AddIt.java

```
1  package addIt;
2
3  public class AddIt {
4
5      public static void main(String[] args) {
6          // TODO Auto-generated method stub
7
8      }
9
10 }
11
```

Add the libraries that support command line input

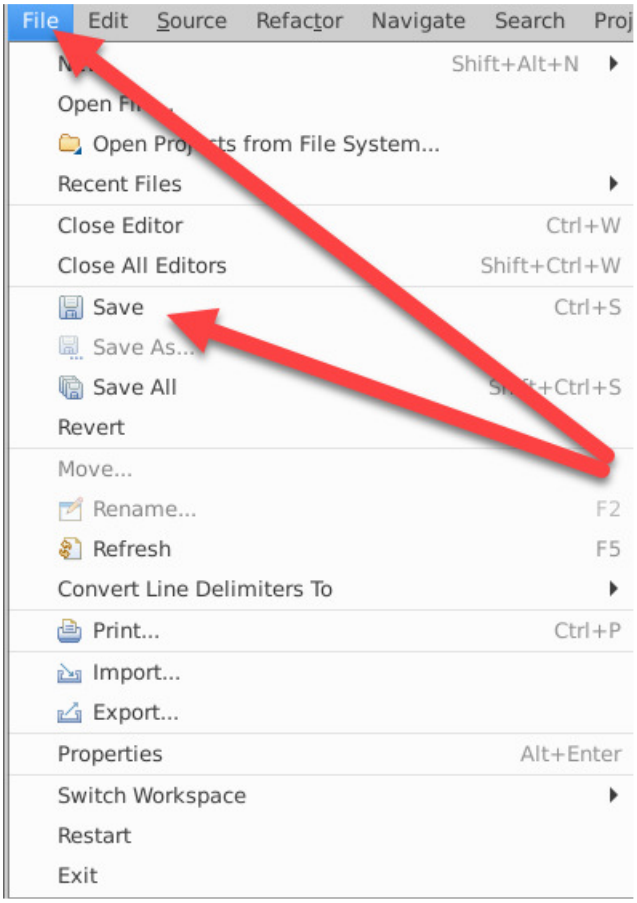
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

These libraries support command line input, and converting the input to data.

AddIt.java

```
1  package addIt;
2
3  import java.io.BufferedReader;
4  import java.io.IOException;
5  import java.io.InputStreamReader;
6
7  public class AddIt {
8
9      public static void main(String[] args) {
10         // TODO Auto-generated method stub
11
12     }
13
14 }
15
```

Click the **File** menu -> **Save**



Enter two methods that return the user's name and adds 2 numbers. Make sure you add the methods above the main method.

AddIt.java

```java
package addIt;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class AddIt {


    public int addEm(int numA, int numB) {

        int sum = numA + numB;
        return sum;
    }

    public String getName(String nm) {

        return "Greetings " + nm + "\n";
    }


```

It is time to instantiate class. We do so in the main method.

In the main method add the code below

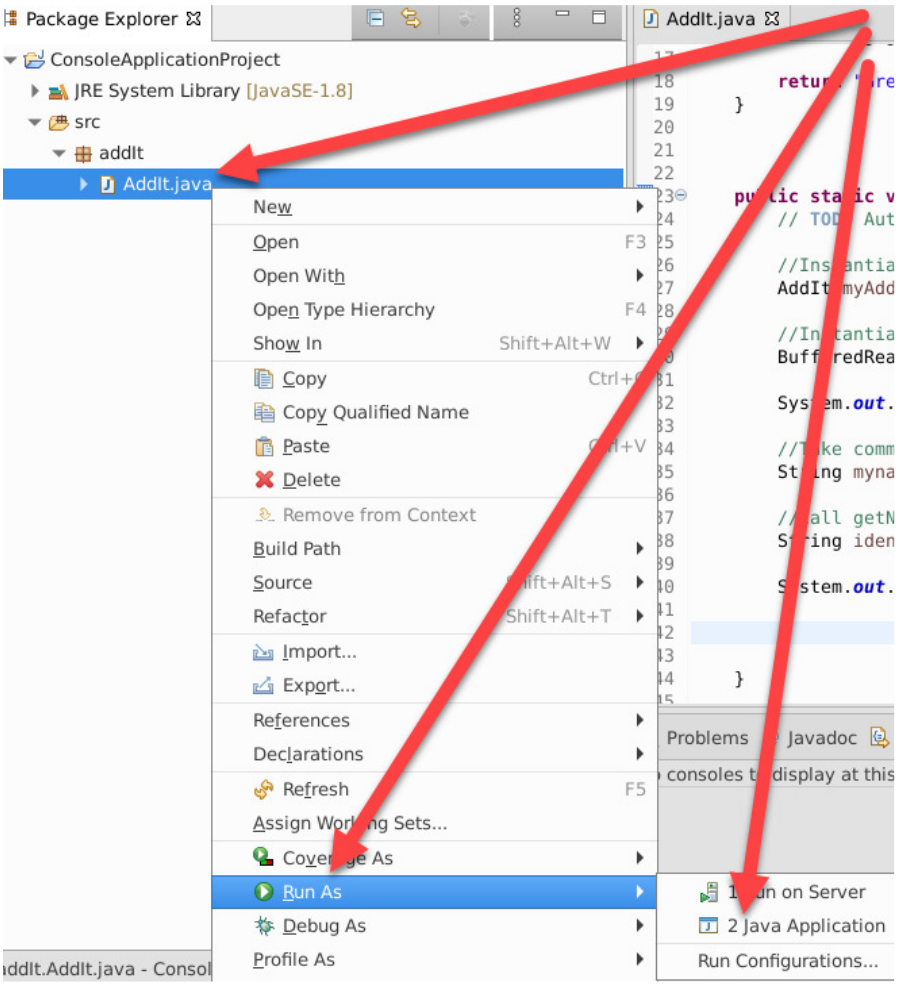
Make sure to edit the main method's header and add "throws IOException" at the end. Take input from the command line can introduce an error, IOException helps handle the errors.

```java
public static void main(String[] args) throws IOException {
    // TODO Auto-generated method stub

    //Instantiate the AddIt class and create the myAddIt object
    AddIt myAddIt = new AddIt();

    //Instantiate the BufferReader and create the readit object
    BufferedReader readIt = new BufferedReader(new InputStreamReader(System.in));

    System.out.println("Enter your name"); //instructions for the user

    //Take command line input and store it in a string
    String myname = readIt.readLine();

    //Call getName on myAddIt pass in the command line parameter and return the name
    String ident = myAddIt.getName(myname);

    System.out.println(ident);  //Output the result of getName to the console


}
```

Click **File** menu -> **Save**

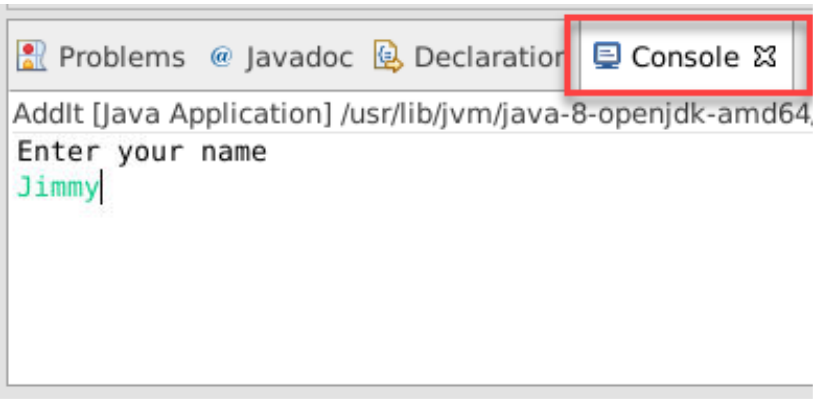
Run the application

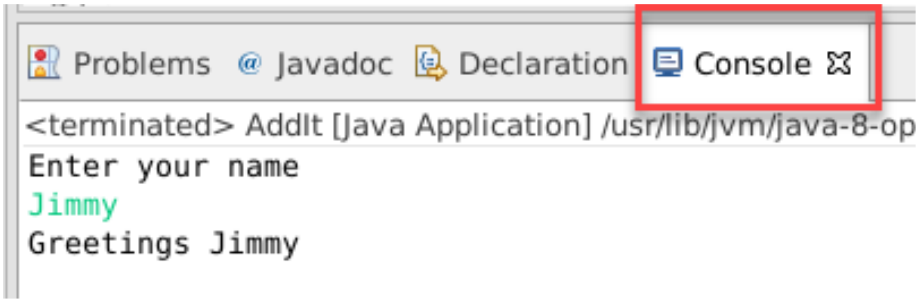
In the Package Explorer view, right click the AddIt.java file, click Run As -> Java Application



Click in the **Console** view

to focus it, and enter a name from the keyboard





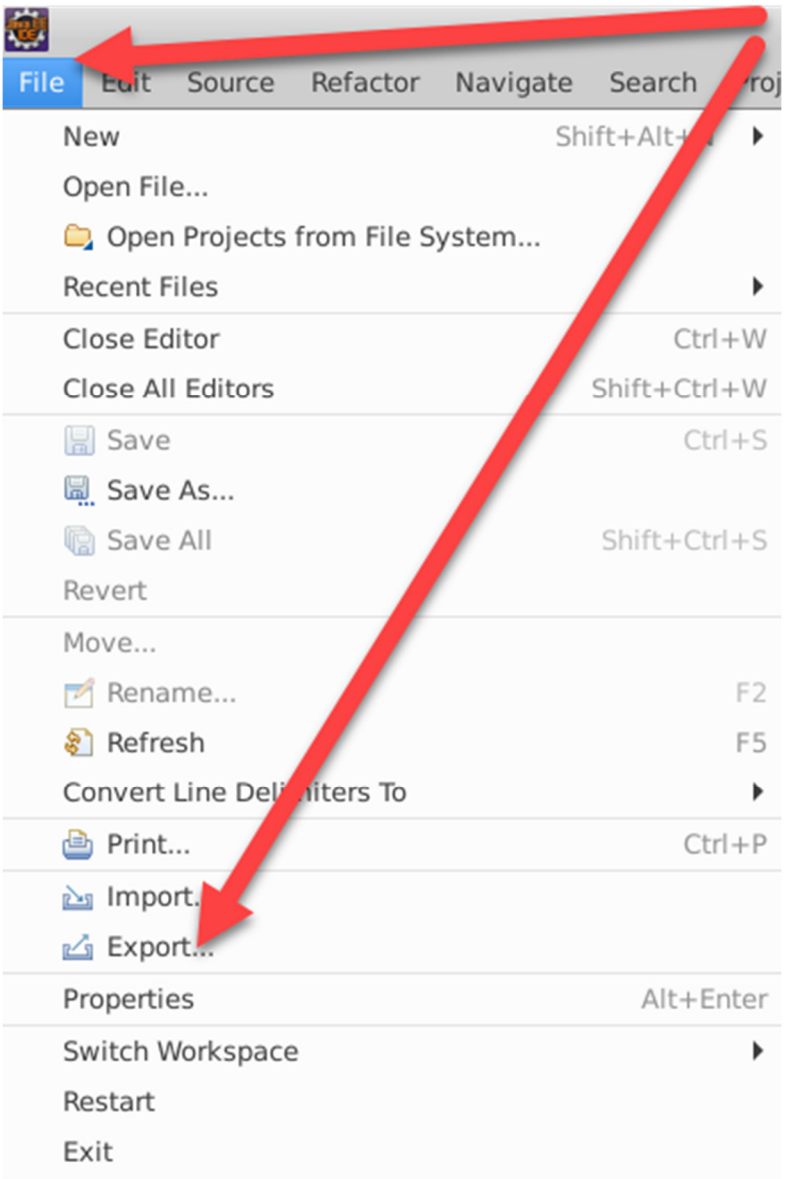
Click **File** -> **Save**

AddIt.java

```java
    public static void main(String[] args) throws IOException {
        // TODO Auto-generated method stub

        //Instantiate the AddIt class and create the myAddIt object
        AddIt myAddIt = new AddIt();

        //Instantiate the BufferReader and create the readit object
        BufferedReader readIt = new BufferedReader(new InputStreamReader(System.in));

        System.out.println("Enter your name"); //instructions for the user

        //Take command line input and store it in a string
        String myname = readIt.readLine();

        //Call getName on myAddIt pass in the command line parameter and return the name
        String ident = myAddIt.getName(myname);

        System.out.println(ident);  //Output the result of getName to the console

        //Get the numbers to add from the command line
        System.out.println("Enter the first number to add");
        String num = readIt.readLine();

        //The command line input comes in as a string. The parseInt method converts the string data to and int data.
        int inum0 = Integer.parseInt(num);

        //Get the numbers to add from the command line
        System.out.println("Enter the second number to add");
        num = readIt.readLine();   //num can be reused

        int inum1 = Integer.parseInt(num);

        //Call method addEm on object myAddIt to add ints inum0 and inum1
        int tot = myAddIt.addEm(inum0, inum1);

        //Output the sum to the console
        System.out.println("The sum of " + inum0 + " and " + inum1 + " is: " + tot);
    }
```

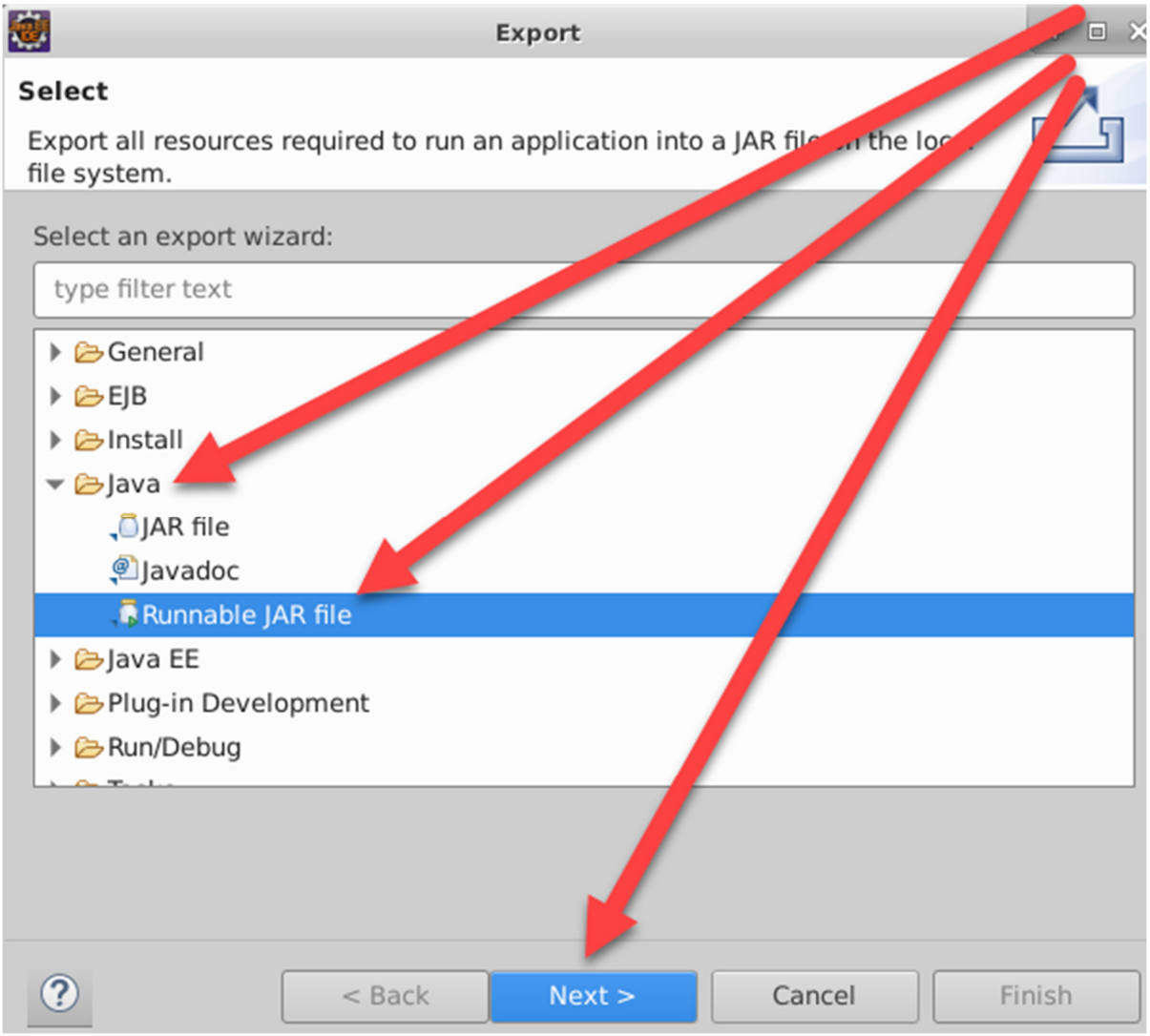
Export a jar file to run from the command line

Click **File** menu -> **Export**



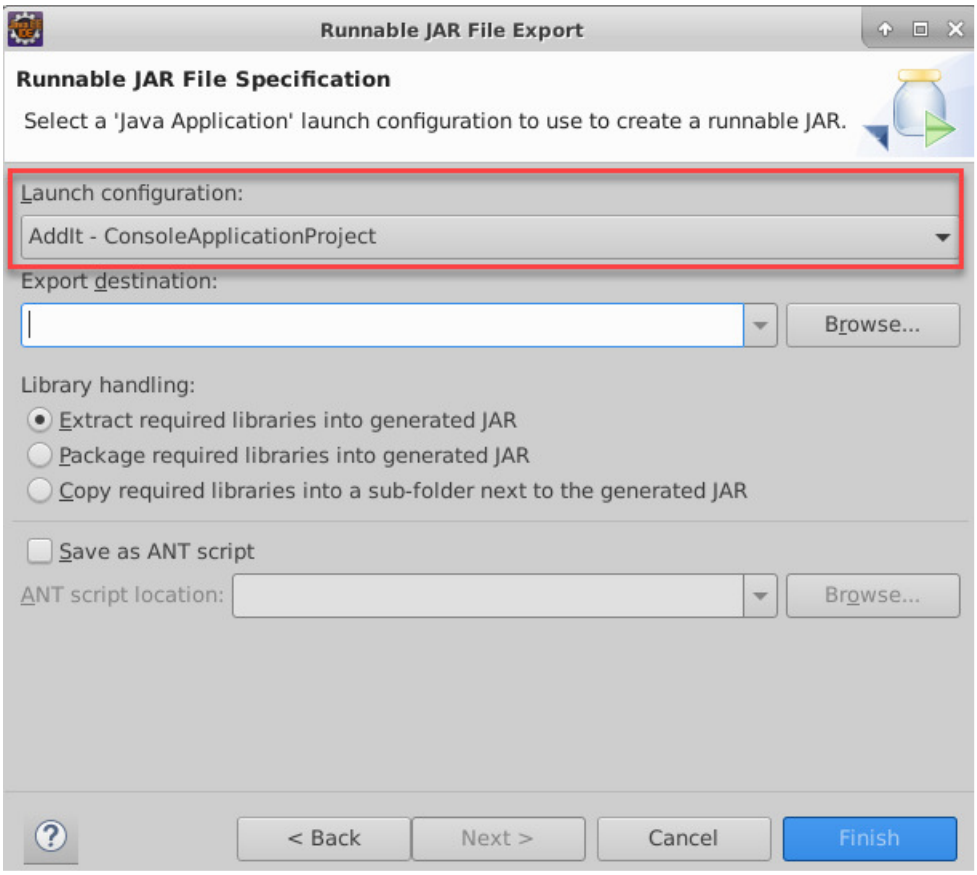
On the **Export** dialog

Click the **Java** folder and select **Runnable Jar file**, and **Next**



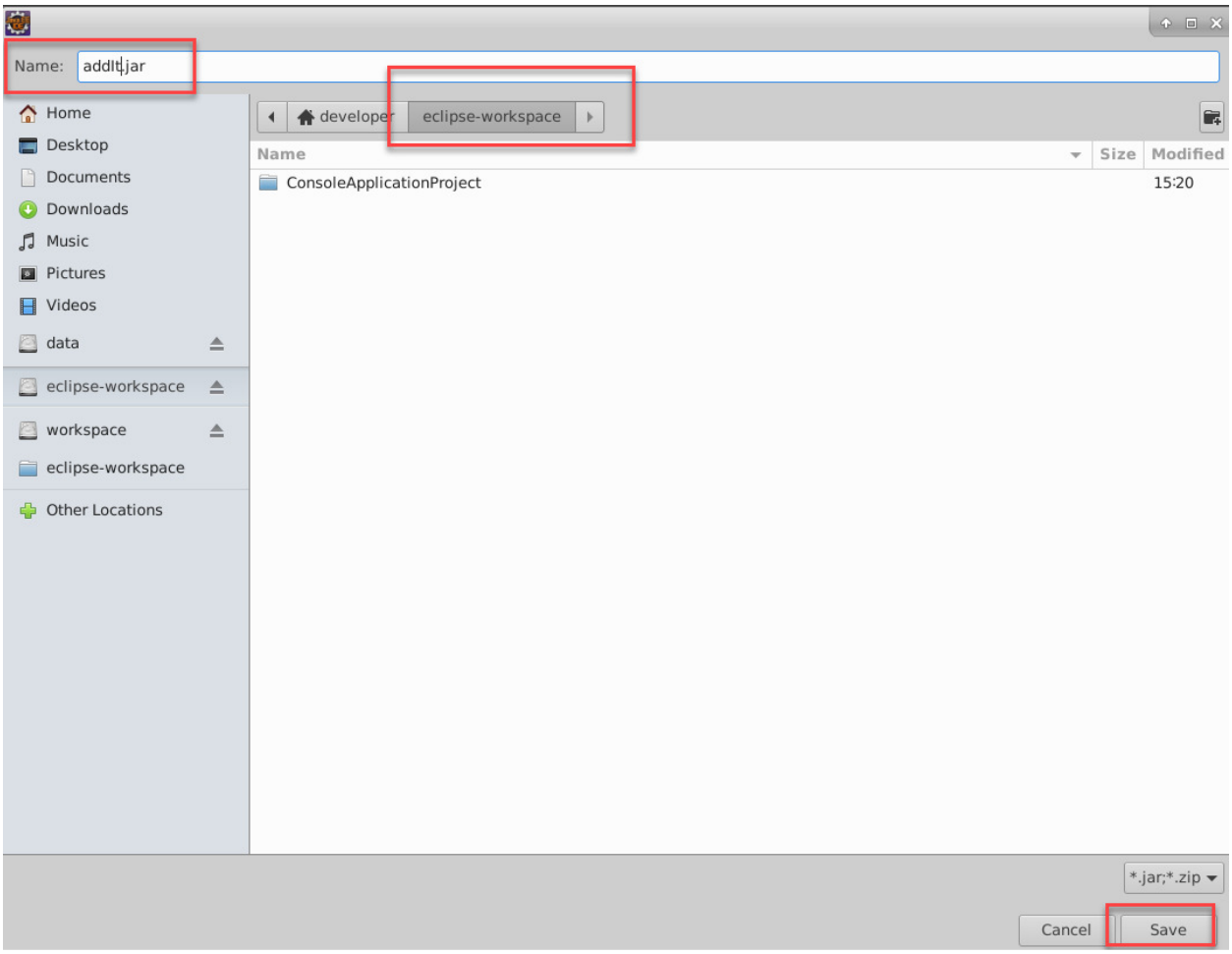
From **Launch configuration** dropdown, select: AddIt – ConsoleApplicationProject

Click **Browse** next to the **Export destination** field, to select the directory for the jar file.

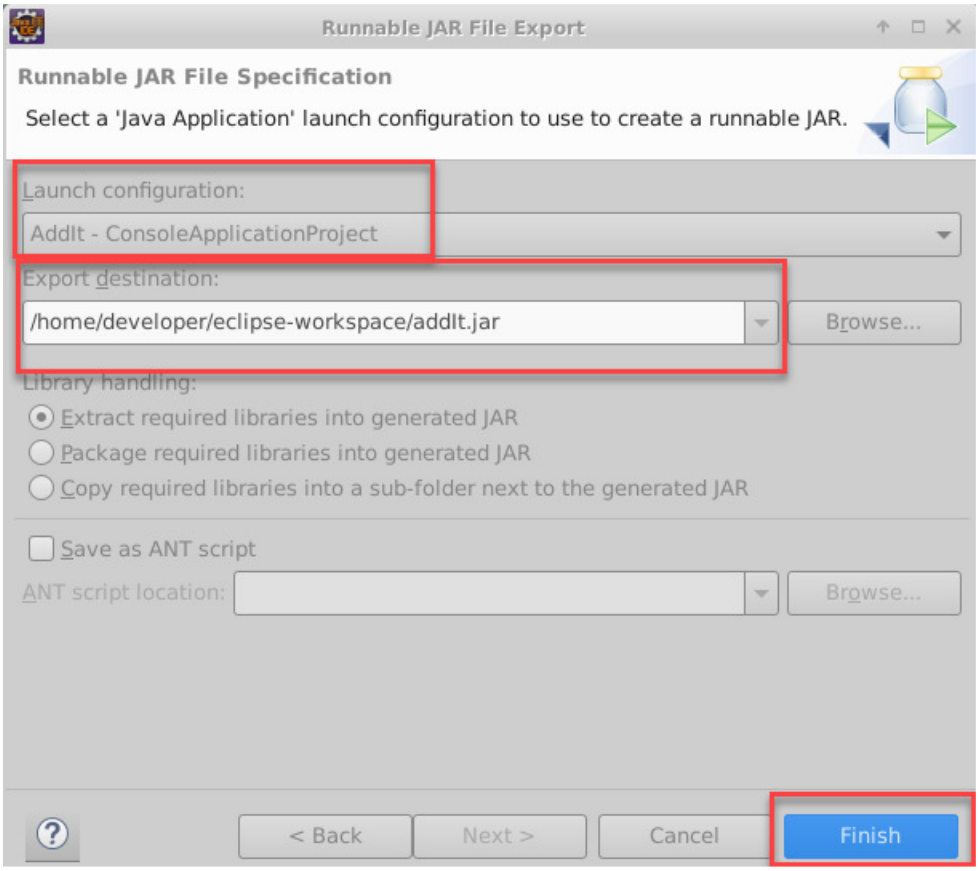


Name the jar file: **addIt.jar**

Click **eclipse-workspace** and **Save**.

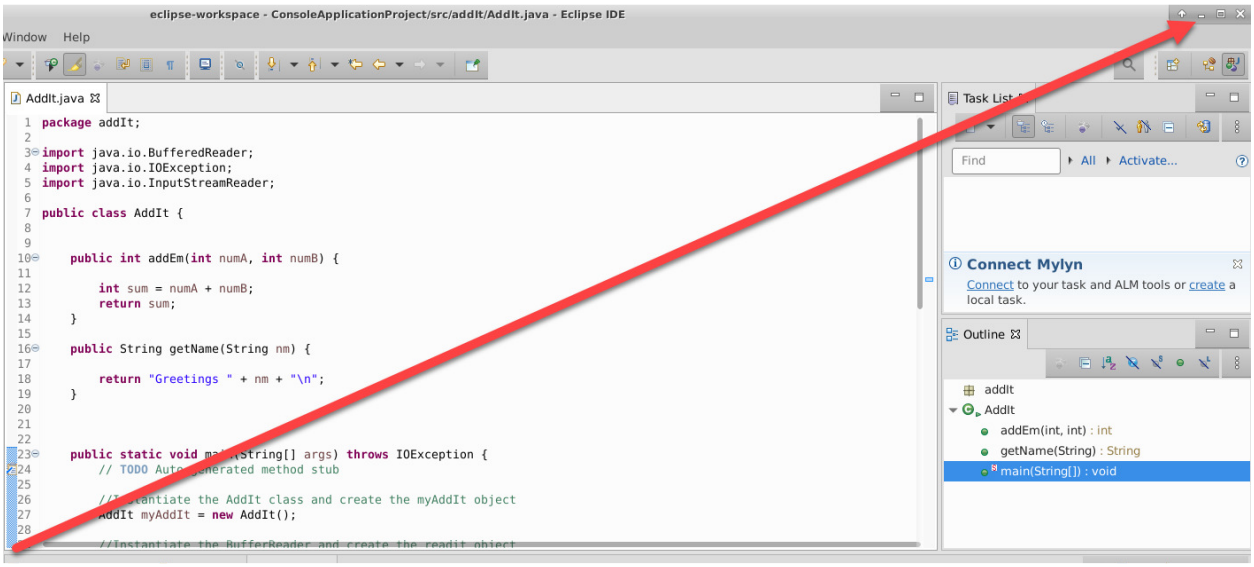


Make note of the **Export destination** directory and click **Finish**

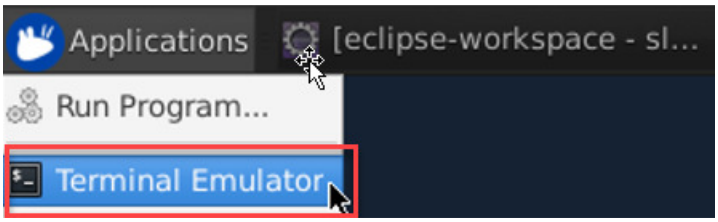


Run the jar from the command line.

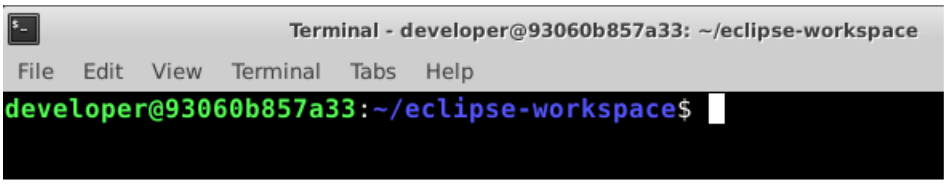
Minimize Eclipse. Click the minimize button in the upper right hand corner. This brings you back to the Desktop.



On the Desktop, click the Applications button, and **Terminal Emulator**.

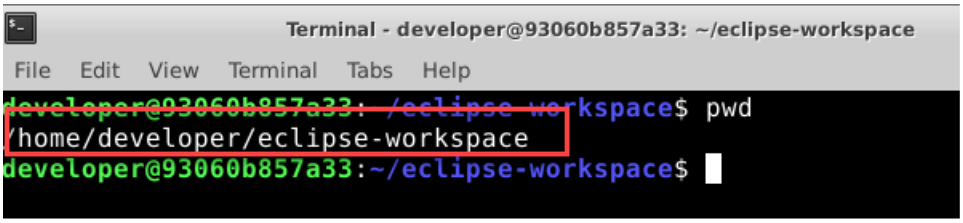


A terminal will open.

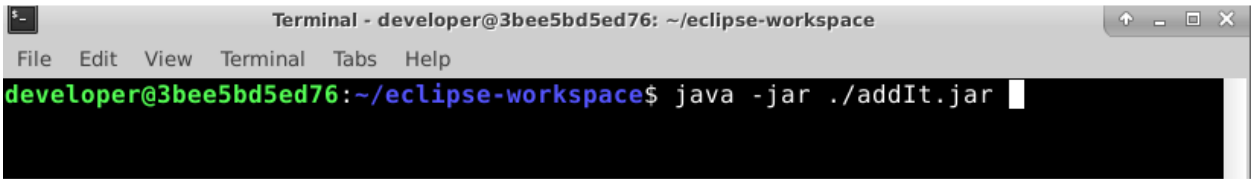


In the terminal, run command: **pwd**

Make sure the present working directory (**pwd**) is: **/home/developer/eclipse-workspace**



To run the jar file enter the command: java -jar ./addIt.jar



Enter your name, and 2 integers to add. Observe the result

```
developer@3bee5bd5ed76:~/eclipse-workspace$ java -jar ./addIt.jar
Enter your name
Jimmy
Greetings Jimmy

Enter the first number to add
17
Enter the second number to add
14
The sum of 17 and 14 is: 31
developer@3bee5bd5ed76:~/eclipse-workspace$
```