

Java Programming

5-3: Deploying an Application

Practice Activities

Vocabulary:

Build application	The process of getting your program ready and available for any user to run the program independent of an IDE.
JNLP	Files that describes how your application/applet should be launched.
HTML	Files can be displayed in a web browser as web pages.
Two Tier Architecture	A java system with a client and a server.
Three Tier Architecture	A java system with a client, a server, and a database.
Package	A collection of java classes organized together.
JAR file	A java version of zip files.

Try It/Solve It:

1. Take the following java code and place it into a package named **greeting** in Eclipse:

```
public class MyName {
    private String name = "Jane";

    public String getName() {
        return name;
    } //end method getName

    public void setName(String name) {
        this.name = name;
    } //end method setName

    public void sayHello() {
        System.out.println("Hi" + this.name);
    } //end method sayHello
} //end class MyName
```

2. Create a new package called converse in the same project and then create a class called Hello. You need to use the sayHello method from your MyName class.
 - a) What do you need to include in your java program to get this to work?
 - b) Show your completed code.

3. You want to be able to change the name and then display the new name from within the Hello class.
 - a) How could you call the setName in a program before displaying the new name (Bob) to screen without using an import statement for the package?
 - b) Show your completed code.

4. Using the following example create a runnable JAR file using Eclipse.

Example Steps

- a) In Eclipse create a new Java Project named dice.
- b) In this project create a package also named dice.
- c) In this package create a class named Random.
- d) Add the following code to the Random class:

```
package dice;

import javax.swing.JOptionPane;

public class Random {

    public static void main(String[] args) {
        int rollofDice;
        String output;

        // generate a random number between 1 and 6 inclusive
        rollofDice = (int) (Math.random() * 6) + 1;
        //create a String message for the output window
        output = "You rolled a " + rollofDice;

        // print message using a window
        JOptionPane.showMessageDialog(null, output, "Random Number Demo",
                                    JOptionPane.INFORMATION_MESSAGE);

        System.exit(0);
    } //end method main
} //end class Random
```

- Save and run your code.
 - Now create a runnable JAR file and test that it runs outside the IDE!
 - Detail the steps taken to create the runnable JAR file!
5. Open the **orclProductionline.pdf** document from Oracle iLearning and complete the project task that will review what you have learned so far in the Java Programming course.