

# Packages



By

Dr M. Senthilkumar  
Assistant Professor

Department of Computer Science

Government Arts and Science College, Avinashi - 641654

# Packages

- ✓ Collection of Classes and/ or Interfaces
- ✓ Organized in a Hierarchical / Tree like Structure

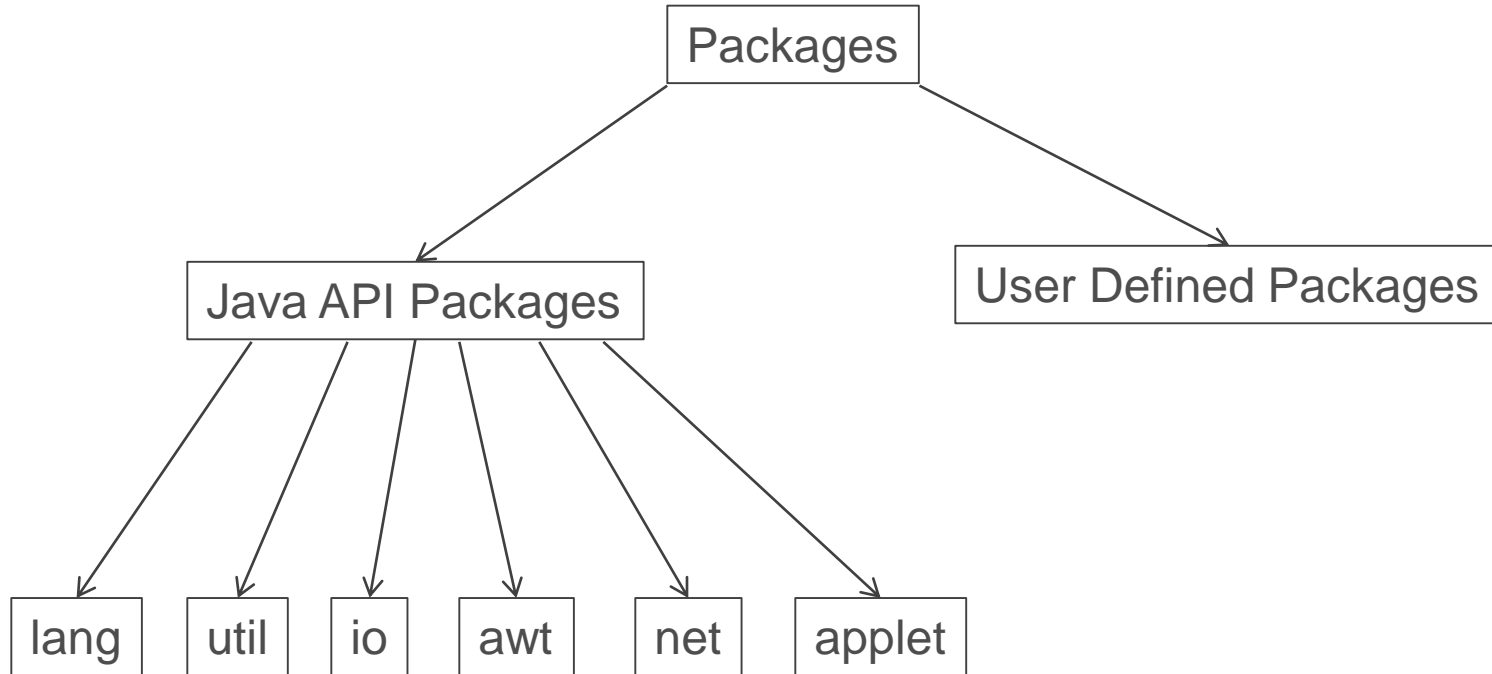
# Packages - Benefits

- ✓ Reusability: Package used by a program can be used by other program
- ✓ Two Classes of two different Packages can have same name

# Packages - Benefits

- ✓ Way to Hide Classes to prevent other Programs / Packages that are meant for internal use only
- ✓ A way to separate Design and Coding.  
First, Design Classes and their Relationship  
Second, Write Java Code to implement Methods  
Possible to change the Implementation at any time

# Types of Packages



# Java API

<code>java.lang</code>	Language support classes
	Automatically imported
	Java compilers uses these classes
	Classes for Primitive data types Strings Mathematical functions Threads Exceptions

# Java API

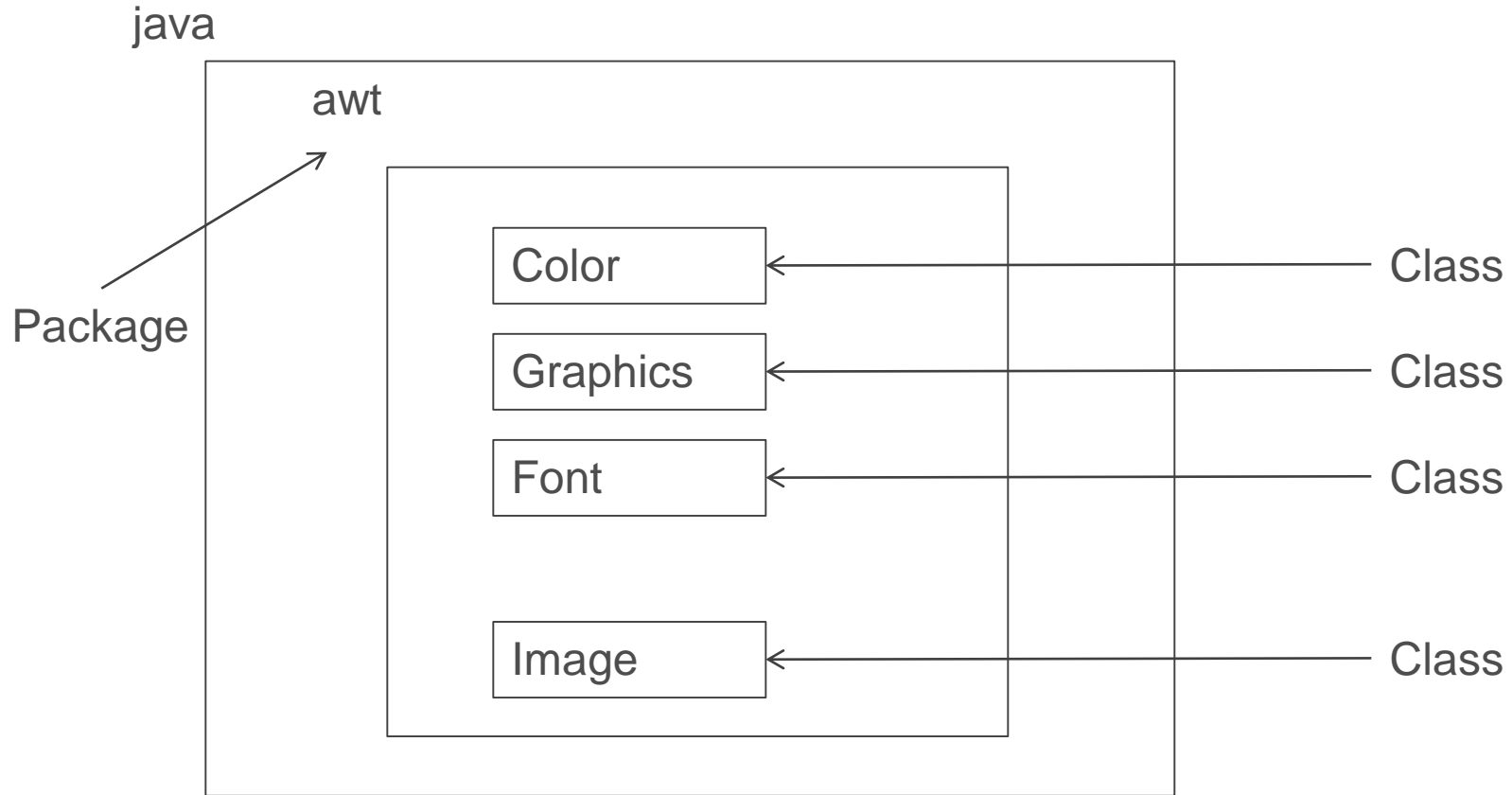
<b>java.lang</b>	<b>Utility classes</b>
	Classes for Vectors Hash Tables Random Numbers Date
<b>java.io</b>	<b>Input/ Output support classes</b>
	Classes for Input and Output Data

# Java API

<b>java.awt</b>	<b>Classes for implementing GUI</b>
	Classes for Buttons Lists Menus Windows
<b>java.net</b>	<b>Networking support classes</b>
	Classes for communicating Local computers and Internet Servers
<b>java.awt</b>	<b>Classes for creating and implementing applets</b>



# Hierarchical View - java.awt package



# Importing a Package

For a Specific Class in the Package

```
import packagename.classname;
```

For all the Classes in the Package

```
import packagename.*;
```

# Importing a Package

```
import java.awt.Color;
```

```
import java.awt.*;
```

# Naming Convention in a Package

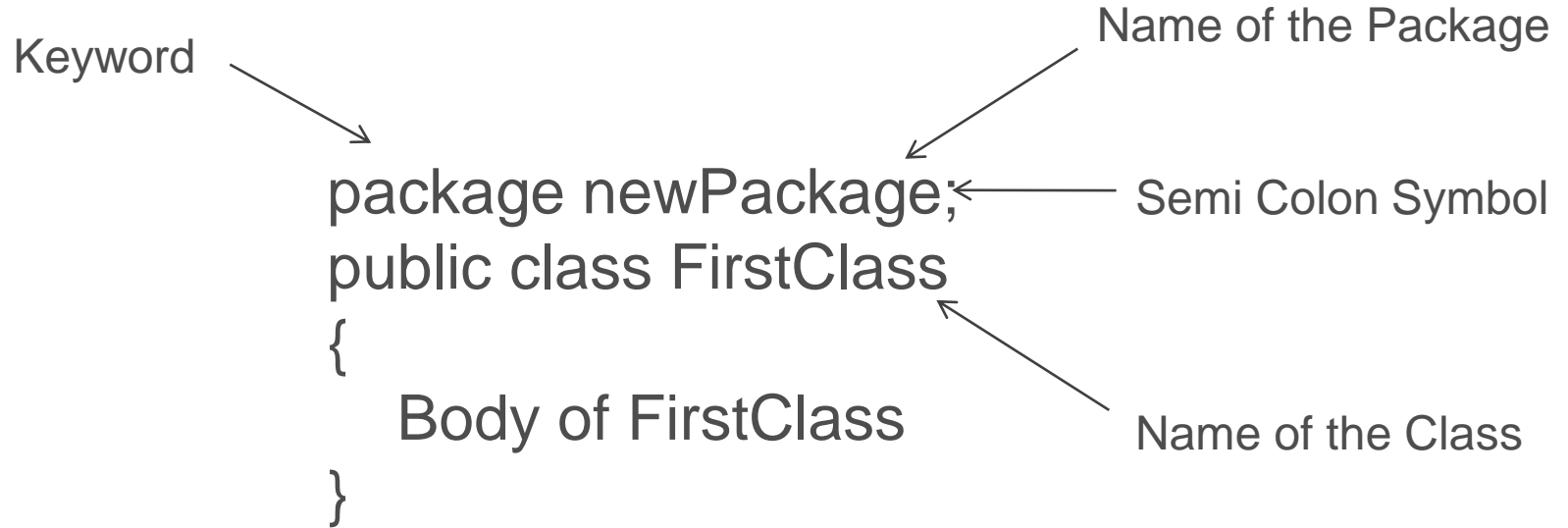
```
double y = java.lang.Math.sqrt(x);
```

Package Name

Class Name

Function Name

# Creating our own Packages



# Rules - Creating our own Packages

1. Declare a Package using  
package packageName;
2. Define a Class inside this package and declare it  
as public
3. Create a Sub-Directory under the Directory  
where main source files are stored  
Sub-Directory name = Package Name

# Rules - Creating our own Packages

4. Store the file as `classname.java` where `classname` is the name of the First Class
5. Compile `classname.java`
6. The `.class` file must be within Sub-Directory

# Example 1 – Creation of Package

```
package package1;

public class ClassA
{
    public void displayA( )
    {
        System.out.println("Inside Class A");
    }
}
```

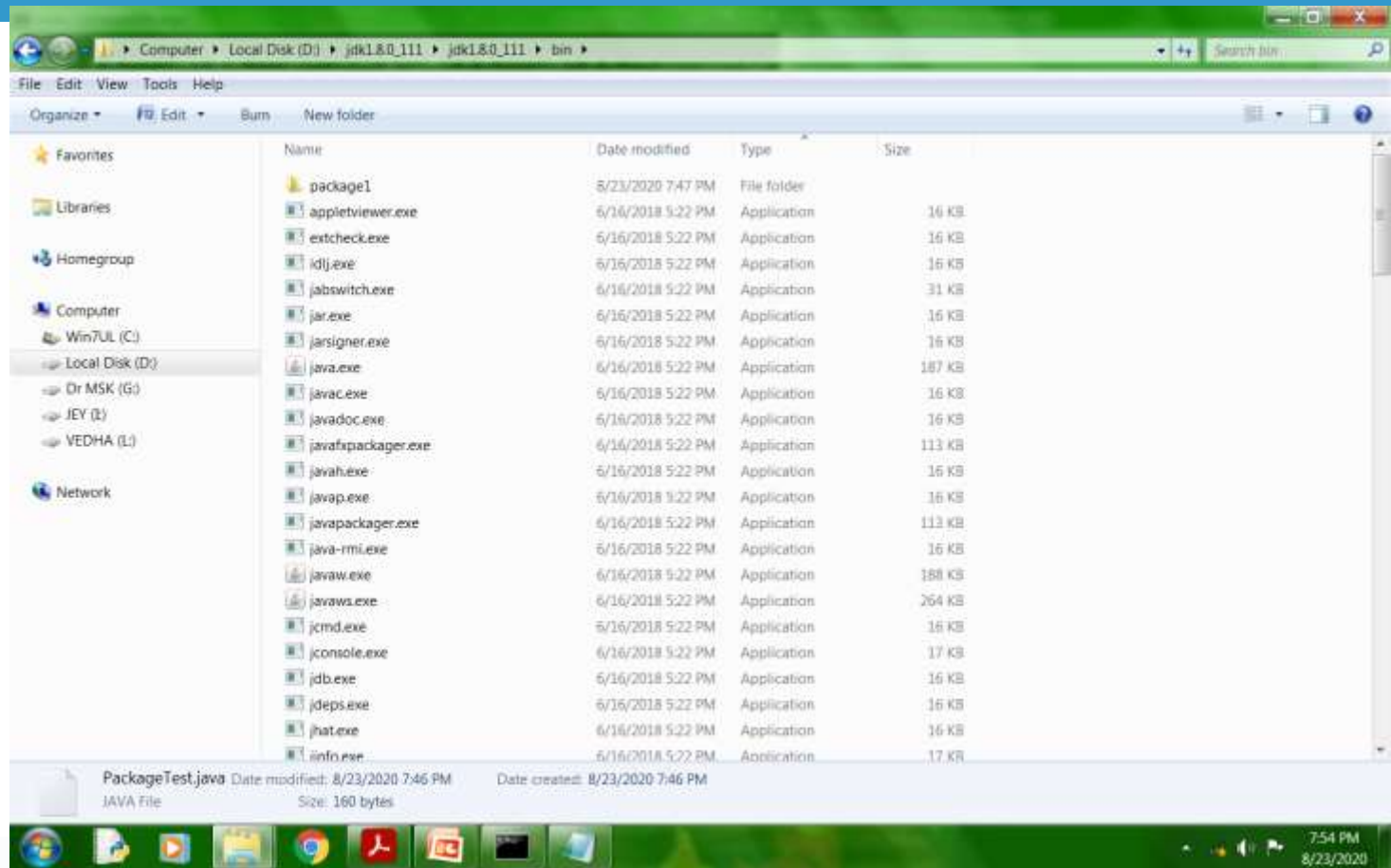
← Subdirectory Name: package1

← Package Name: package1

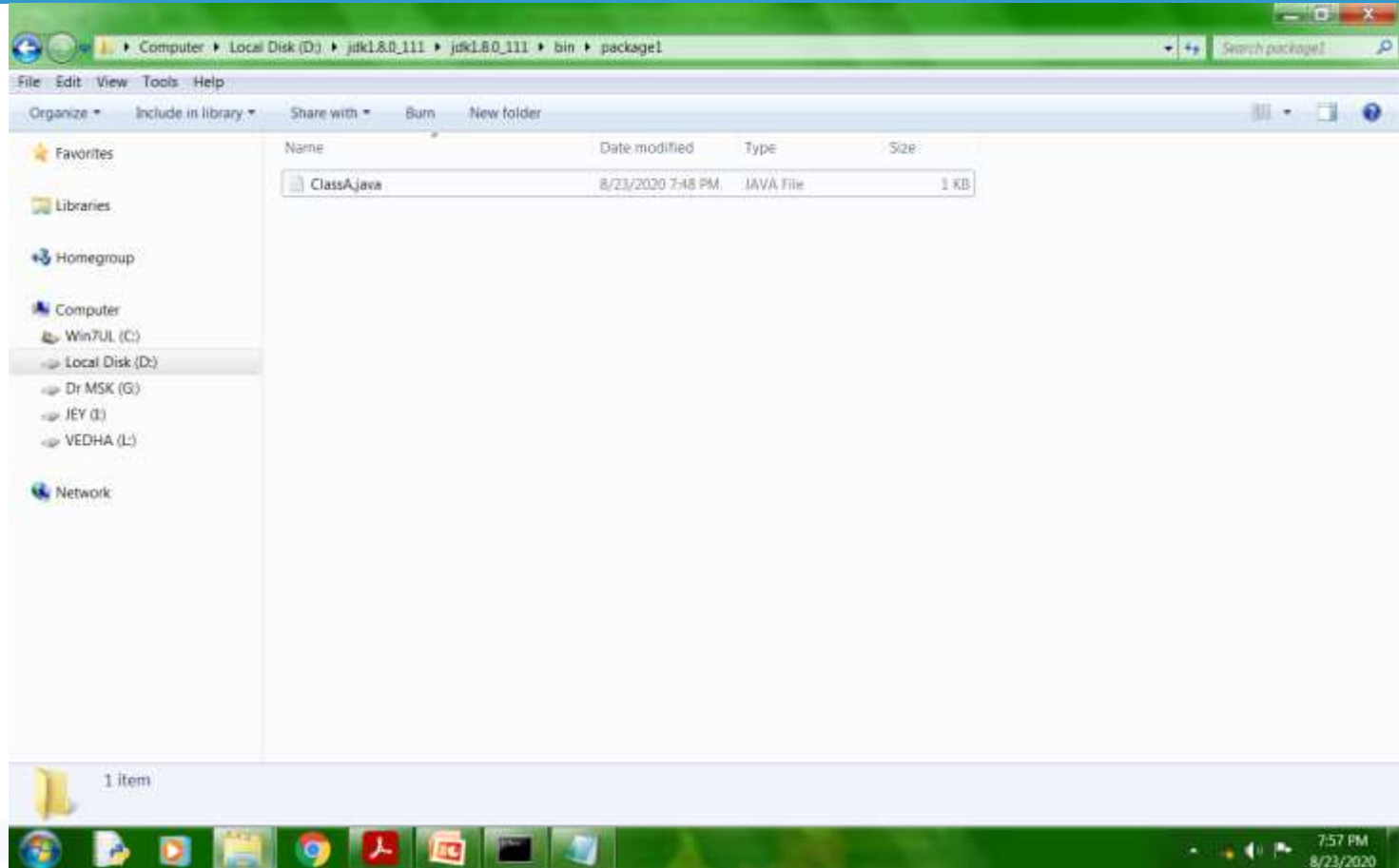
← Source File: ClassA.java



# Example 1 – Sub-Directory



# Example 1 – Create a Class under Package

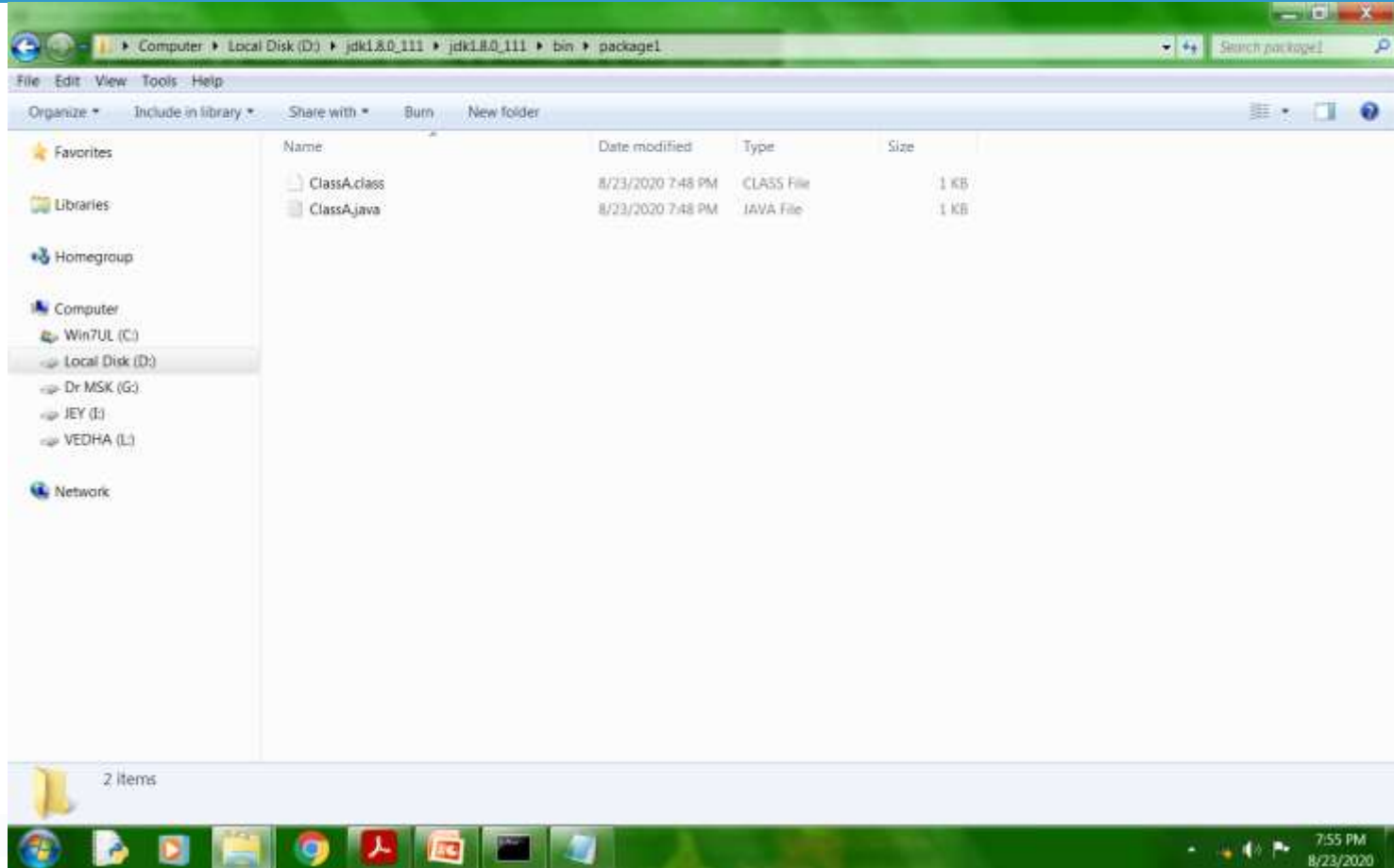


# Example 1 – Compile the Class of the Package

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>javac D:\jdk1.8.0_111\jdk1.8.0_111\bin\package1  
\ClassA.java
```

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>
```

# Example 1 – See .class file under Package



# Example 1 – Import the Created Package

```
import package1.ClassA;
class PackageTest
{
    public static void main(String args[ ])
    {
        ClassA ObjectA = new ClassA( );
        ObjectA.displayA( );
    }
}
```

# Example 1 – Compile and Run the Java Code

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>javac PackageTest.java
```

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>java PackageTest  
Inside Class A
```

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>
```

# Example 2 – Creation of Package

```
package package2;
public class ClassB
{
    protected int m = 10;
    public void displayB( )
    {
        System.out.println("Inside Class B");
        System.out.println("m = " + m);
    }
}
```

# Example 2 – Importing Package

```
import package1.ClassA;
import package2.*;
class PackageTest2
{
    public static void main(String args[ ])
    {
        ClassA ObjectA = new ClassA( );
        ClassB ObjectB = new ClassB( );
        ObjectA.displayA( );
        ObjectB.displayB( );
    }
}
```



# Example 2 - Output

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>javac D:\jdk1.8.0_111\jdk1.8.0_111\bin\package2\ClassB.java
```

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>javac PackageTest2.java
```

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>java PackageTest2
```

Inside Class A

Inside Class B

m = 10

```
D:\jdk1.8.0_111\jdk1.8.0_111\bin>
```

# Visibility Control

Access Modifier/ Access Location	public	protected	friendly (default)	private protected	private
Same Class	Yes	Yes	Yes	Yes	Yes
Subclass in Same Package	Yes	Yes	Yes	Yes	No
Other Classes in the Same Package	Yes	Yes	Yes	No	No
Subclass in Other Package	Yes	Yes	No	Yes	No
Non-Subclass in Other Package	Yes	No	No	No	No

# References

- ✓ Programming with Java – A Primer - E. Balagurusamy, 3rd Edition, TMH

Thank You