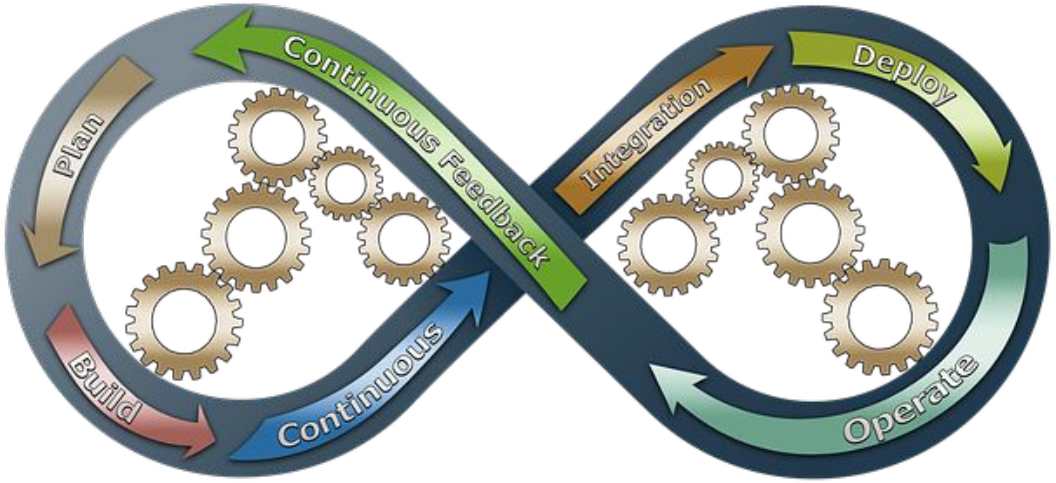


Session 3.5

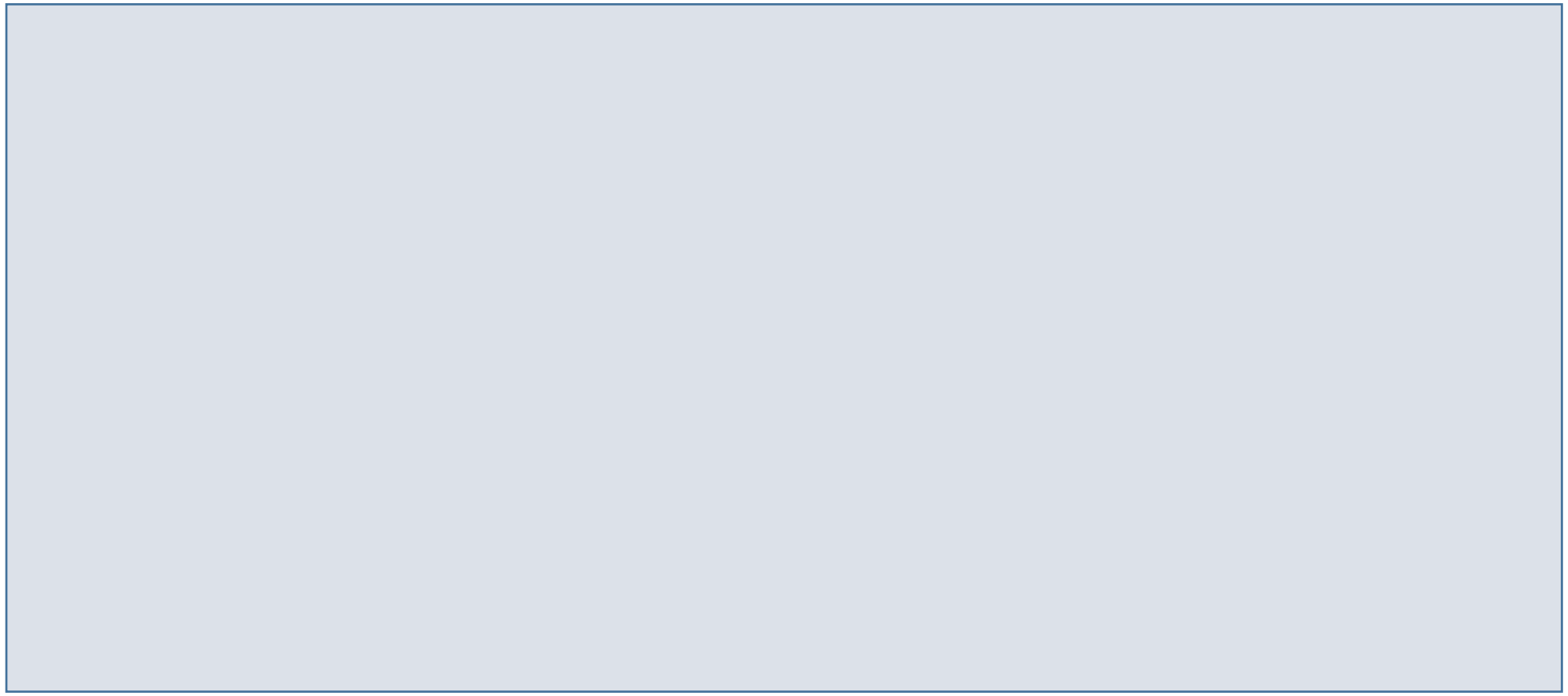
Access Control

AN INITIATIVE BY

UNICAL ACADEMY

© Unical Systems 2021

Introduction



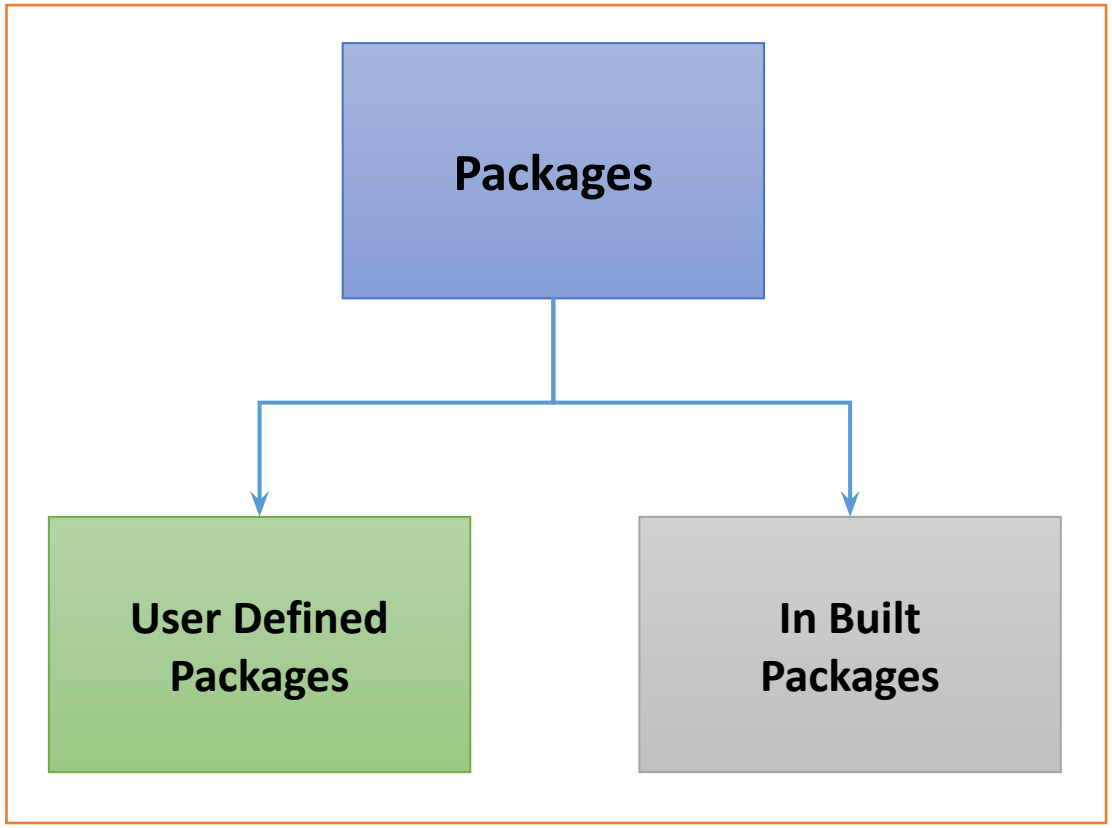
Let’s go!!!

© Unical Systems 2021

Access Control

UNICAL ACADEMY

Packages



Built-in Packages:

Package	Description
java.lang	Contains language support classes (e.g classes which defines primitive data types, math operations). This package is automatically imported.
java.io	Contains classes for supporting input / output operations.
java.util	Contains utility classes which implement data structures like Linked List, Dictionary and support ; for Date / Time operations.
java.applet	Contains classes for creating Applets
java.awt	Contain classes for implementing the components for graphical user interfaces (like buttons , menus etc.,)
java.net	Contain classes for supporting networking operations

Built-in Package Declaration Example:

```
// import all the classes from util package
import java.util.*;
// import the Vector class from util package
import java.util.vector;
// import the ArrayList class from util package
import java.util.ArrayList;
```

© Unical Systems 2021

UNICAL ACADEMY

User Defined Package declaration

1. Creating a Package

Choose a name for the package and include a *package* command as the first statement in the source file.

```
package MyPackage;
```

2. Including a Class in Package

Declare the package name as the first statement of program. Then include the class as part of the package.

```
package MyPackage;
public class Calculator {
     public int add(int a, int b) {
       return a+b;
      }
    public static void main(String args[]) {
       Calculator obj = new Calculator();
      System.out.println(obj.add(10, 20));
       }
 }
```

3. Use package in another program

To use the class Calculator, import the package MyPackage.

```
import MyPackage.Calculator;
public class Demo{
      public static void main(String args[]){
         Calculator obj = new Calculator();
         System.out.println(obj.add(100, 200));
      }
}
```

4. To use all the classes of package

```
Import MyPackage.*;
```

© Unical Systems 2021

UNICAL ACADEMY

5. Creating a class inside package while importing another package

- package declaration
- package import

```
//Declaring a package
package anotherpackage;
//importing a package
import MyPackage.Calculator;
public class Example{
        public static void main(String args[]){
              Calculator obj = new Calculator();
              System.out.println(obj.add(100, 200));
          }
}
```

© Unical Systems 2021

UNICAL ACADEMY

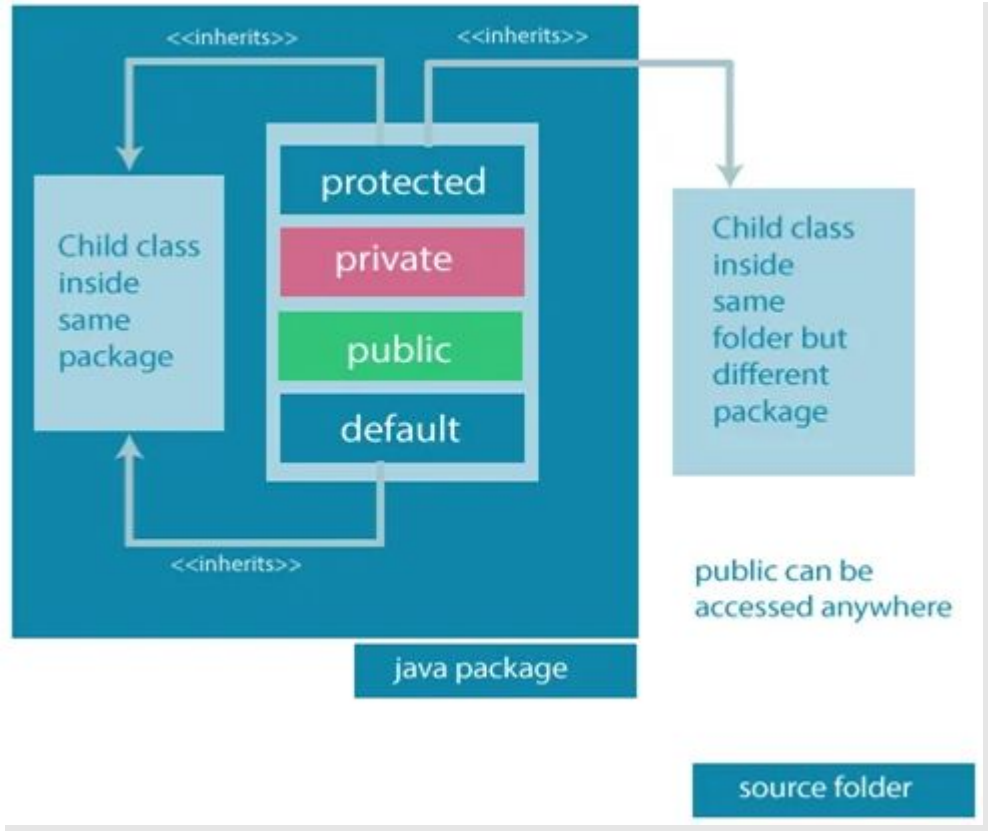
Default Access Modifier Example:

```
package MyPackage;
class Student {
    void display(){
     System.out.println("This is a Class");
     }
}
```

Private Access Modifier Example:

```
class Student {
      private String name;
    // getter method
     public String getName() {
         return this.name;
     } // setter method
     public void setName(String name) {
         this.name= name;
      }
}
public class Main {
          public static void main(String[] main){
             Data d = new Data();
             // access the private variable using the getter and setter
            d.setName("This is a Class ");
            System.out.println(d.getName());
          }
}
```

Accessibility of all Access Modifiers in Java



© Unical Systems 2021

UNICAL ACADEMY

Protected Access Modifier Example:

```
class Selenium {
      // protected method
       protected void display() {
         System.out.println("I am a student");
         }
 }
class Student extends Selenium {
       public static void main(String[] args) {
           // create an object of Student class
                Student student = new Student();
                // access protected method
                student.display();
             }
   }
```

Public Access Modifier Example:

```
// Selenium.java file
// public class
public class Selenium {
     // public variable
     public int stdtCount;
        // public method
     public void display() {
           System.out.println("I am a student.");
           System.out.println("we are " + stdtCount + " students.");
       }
}
// Main.java
public class Main {
     public static void main( String[] args ) {
         // accessing the public class
          Selenium student = new Selenium();
         // accessing the public variable
         student.stdtCount = 15;
        // accessing the public method
          student.display();
       }
 }
```

© Unical Systems 2021

Simple Imports & Static Imports

UNICAL ACADEMY

Simple Import

- It allows the programmer to access classes of a package without package qualification.
- It provides accessibility to classes and interface

Static Import

- It allows the programmer to access the static members of a class without the class qualification
- It provides accessibility to static members of the class

Without Static Imports Example:

```
class Student{
      public static void main(String args[]) {
           double var1= Math.sqrt(5.0);
           double var2= Math.tan(30);
            System.out.println("Square of 5 is:"+ var1);
            System.out.println("Tan of 30 is:"+ var2);
          }
}
```

Using Static Imports Example:

```
import static java.lang.System.out;
import static java.lang.Math.*;
class Student{
       public static void main(String args[]) {
          //instead of Math.sqrt need to use only sqrt
           double var1= sqrt(5.0);
          //instead of Math.tan need to use only tan
          double var2= tan(30);
         //need not to use System in both the below statements
          out.println("Square of 5 is:"+var1);
          out.println("Tan of 30 is:"+var2);
        }
   }
```

© Unical Systems 2021

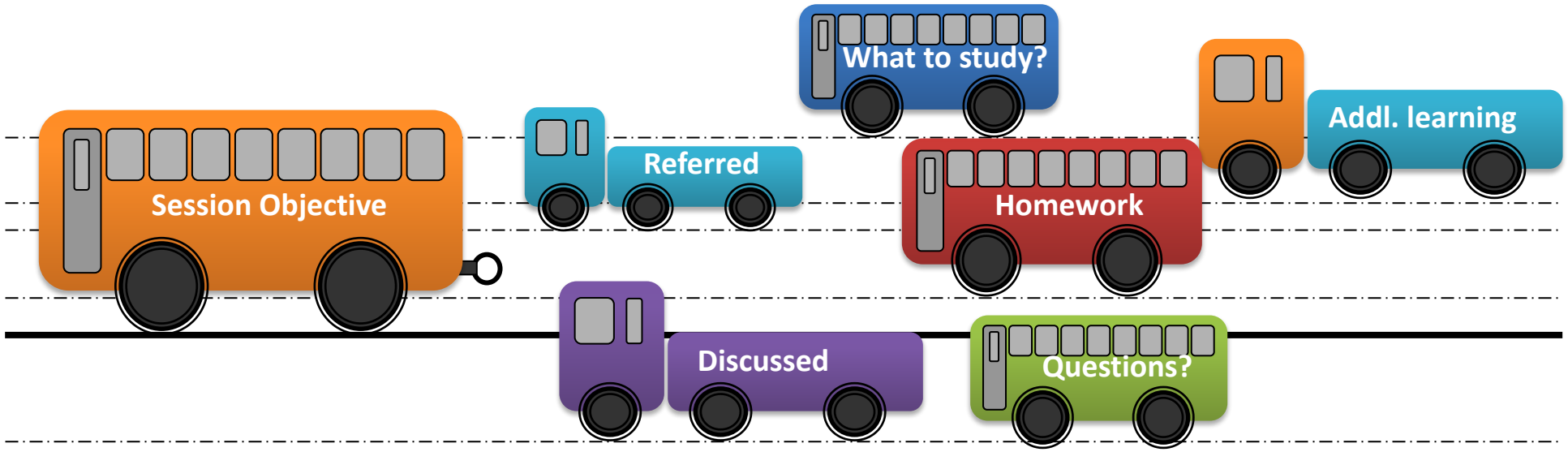
Java Programming Structure

UNICAL ACADEMY

Documentation Section	We can write a comment in this section. Comments are beneficial for the programmer because they help them understand the code.
Package Statement	We can create a package with any name. A package is a group of classes that are defined by a name. That is, if we want to declare many classes within one element, then we can declare it within a package. It is declared as: package package_name;
Import Statement	This indicates that if we want to use a class of another package, then we can do this by importing it directly into your program. Example: import calc.add;
Interface Section	Interfaces are like a class that includes a group of method declarations. It can be used when programmers want to implement multiple inheritances within a program.
Class Definition	A Java program may contain several class definitions. Classes are the main and essential elements of any Java program.
Main Method Class	Every Java stand-alone program requires the main method as the starting point of the program. There may be many classes in a Java program, and only one class defines the main method. Methods contain data type declaration and executable statements.

© Unical Systems 2021

Session Recap



© Unical Systems 2021