System.out.println Statement

Definition

- System.out.println() is a built-in Java statement used to print output to the console.
- It helps developers see results, debug programs, or display information to users.
- System is a predefined class.
- out is a static member of the System class and is an instance of PrintStream.
- println() is a method of the PrintStream class that prints the output followed by a new line.

Structure Breakdown

System.out.println("Hello, Java!");

- **System** → A final class that provides access to system resources.
- **out** → A static member representing the standard output stream.
- **println()** → Method that prints data and moves to the next line.

Example

true

```
public class HelloWorld {
   public static void main(String[] args) {
      System.out.println("Welcome to Core Java!");
      System.out.println(123);  // Prints integer
      System.out.println(3.14);  // Prints decimal value
      System.out.println(true);  // Prints boolean value
    }
}
Output:
Welcome to Core Java!
123
3.14
```

Key Points

- ✓ System.out.print() → prints without new line.
- \checkmark System.out.println() → prints with a new line.
- \checkmark Used in debugging and displaying outputs in console applications.

Package in Java

Definition

- A package in Java is a group of related classes and interfaces.
- It helps organize code, avoid name conflicts, and control access.
- Packages are like folders/directories in a file system.

✓ Built-in Packages

Examples:

- java.lang → core classes like String, Math.
- java.util → utility classes like Scanner, ArrayList.
- java.io → input/output related classes.

User-defined Packages

• We can create our own packages to organize classes.

Creating a Package

Step 1: Create a folder structure representing the package.

Example: com.example.utilities

Step 2: Write a class inside the package.

package com.example.utilities;

```
public class MathHelper {
  public static int square(int number) {
    return number * number;
  }
}
```

Step 3: Import and use the package in another class.

import com.example.utilities.MathHelper;

```
public class TestPackage {
   public static void main(String[] args) {
     int result = MathHelper.square(5);
     System.out.println("Square of 5 is: " + result);
   }
}
Output:
Square of 5 is: 25
```

Why Use Packages?

- ✓ Code Organization: Keeps related classes together.
- ✓ Avoid Naming Conflicts: Same class names can exist in different packages.
- ✓ Access Protection: Control visibility with access specifiers (public, private, protected).
- √ Reusability: Easy to reuse classes across multiple projects.

Subpackage in Java

Definition

- A subpackage is a package nested inside another package.
- It helps further organize classes in a hierarchical way.

Example:

```
com.example.utilities.math
```

✓ Usage Example

Class in subpackage:

```
package com.example.utilities.math;
```

```
public class AdvancedMath {
  public static int cube(int number) {
    return number * number * number;
  }
}
```

Importing subpackage in main program:

```
import com.example.utilities.math.AdvancedMath;
```

```
public class TestSubpackage {
  public static void main(String[] args) {
    int result = AdvancedMath.cube(3);
    System.out.println("Cube of 3 is: " + result);
  }
}
```

Output:

Cube of 3 is: 27

✓ Key Notes on Packages & Subpackages

- ✓ Package names are usually in lowercase.
- ✓ Subpackages are created by adding further directories.
- ✓ Import statements are used to access classes from other packages.
- \checkmark Classes in subpackages are fully qualified by their package name.

Problem:

```
public class MindBendingExample2 {
  public static void main(String[] args) {
    System.out.print("Path: C:\\Users\\Admin\n");
    System.out.println("Quote: \"Believe in yourself\"");
    System.out.print("Tab\tseparated\tvalues\n");
    System.out.println("Line1\nLine2");
    System.out.print("This is ");
    System.out.print("a test.\n");
    System.out.println("And this is another line.");
}
```

Path: C:\Users\Admin

Quote: "Believe in yourself"

Tab separated values

Line1

Line2

This is a test.

And this is another line.