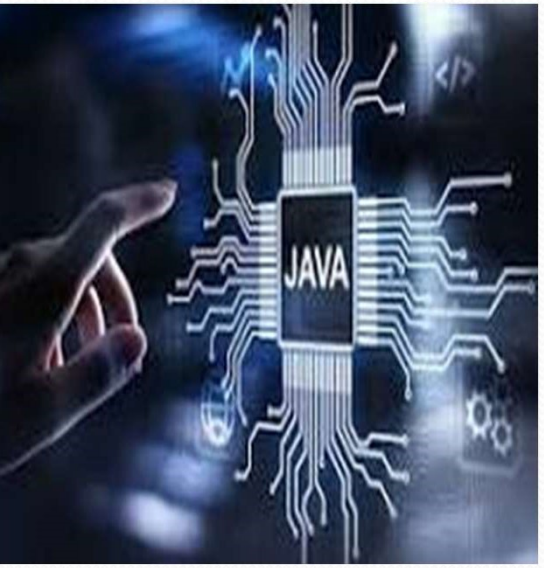
SNS COLLEGE OF TECHNOLOGY

An Autonomous Institution
Coimbatore-35

Department of Computer Science & Engineerin

23ITT202 – Object Oriented Programming

I B.E CSE/ II SEMESTER



UNIT II :Control Statements and Constructo

Topic : Access specifiers in Java

Topics for Discussion

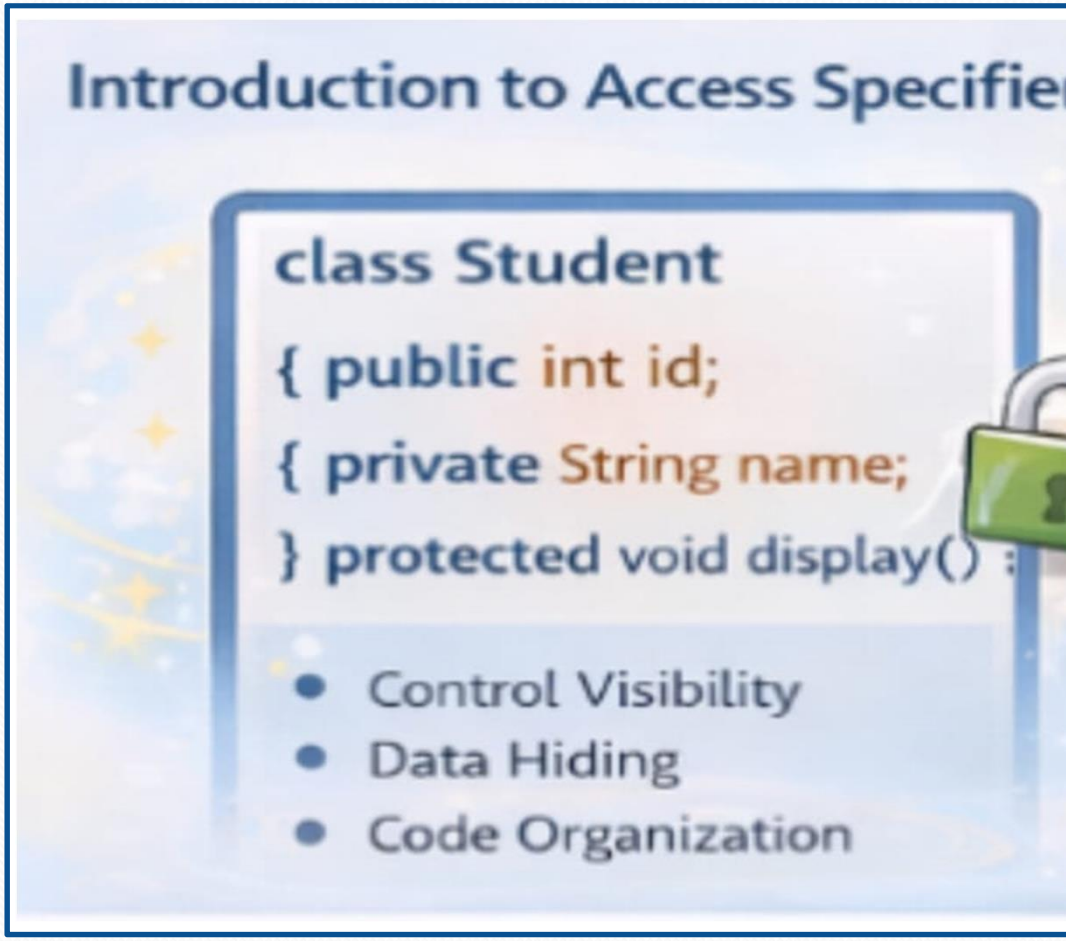
- ✓ Introduction to Access Specifiers.
- ✓ Need for Access Specifiers.
- ✓ Types of Access Specifiers.
- ✓ Public Access Specifier.
- ✓ Private Access Specifier.
- ✓ Protected Access Specifier.
- ✓ Default Access Specifier.

Introduction

Access Specifiers in Java are used to control the visibility of classes, variables, methods, and constructors.

They define who can access a particular member of a class.

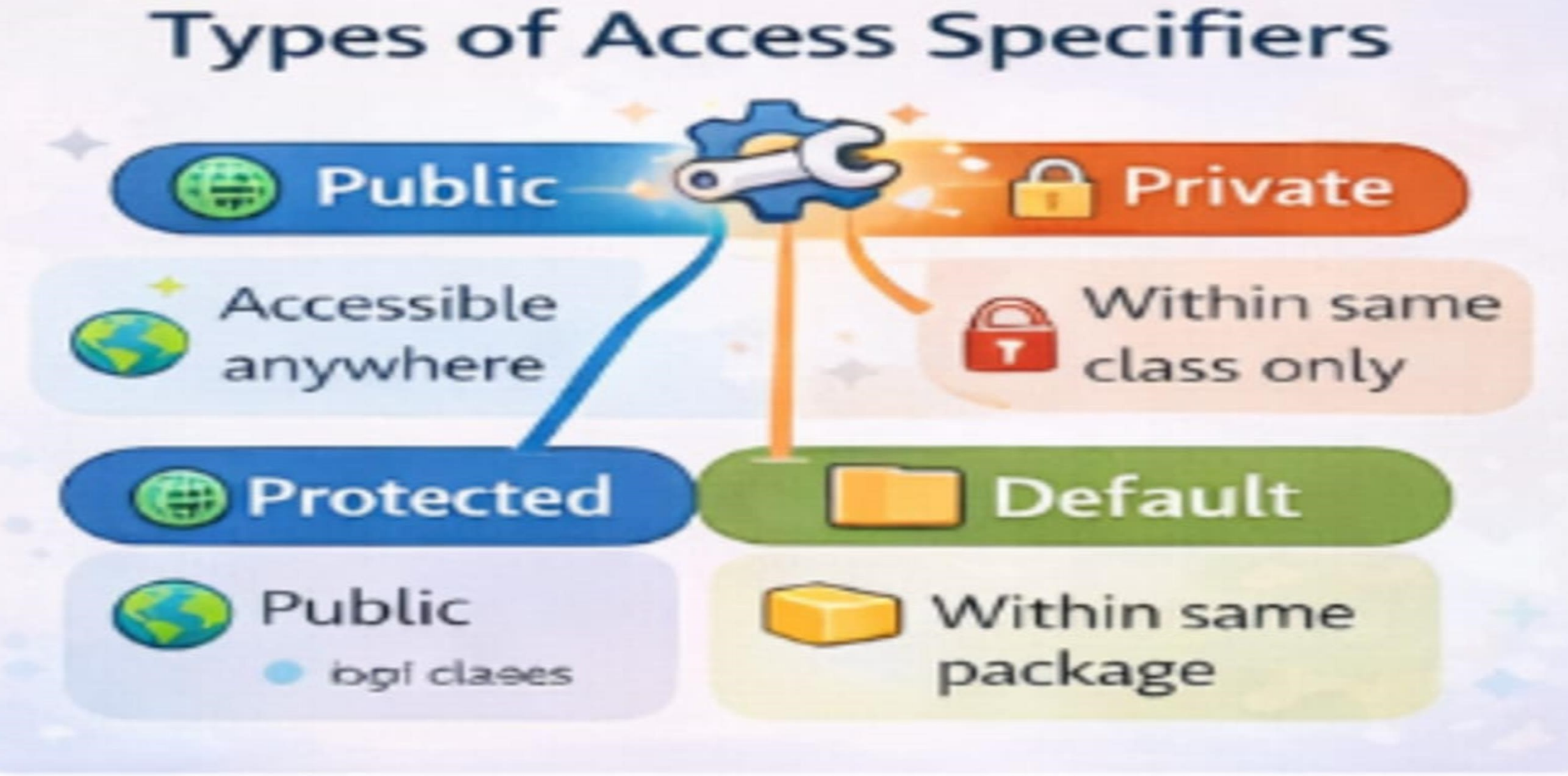
Access specifiers help in data hiding, security, and proper code organization.



Need For Access Specifiers

Why Access Specifiers are Needed?

- To protect data from unauthorized access.
- To implement encapsulation.
- To control access within packages and subclasses.
- To improve code security and maintainability

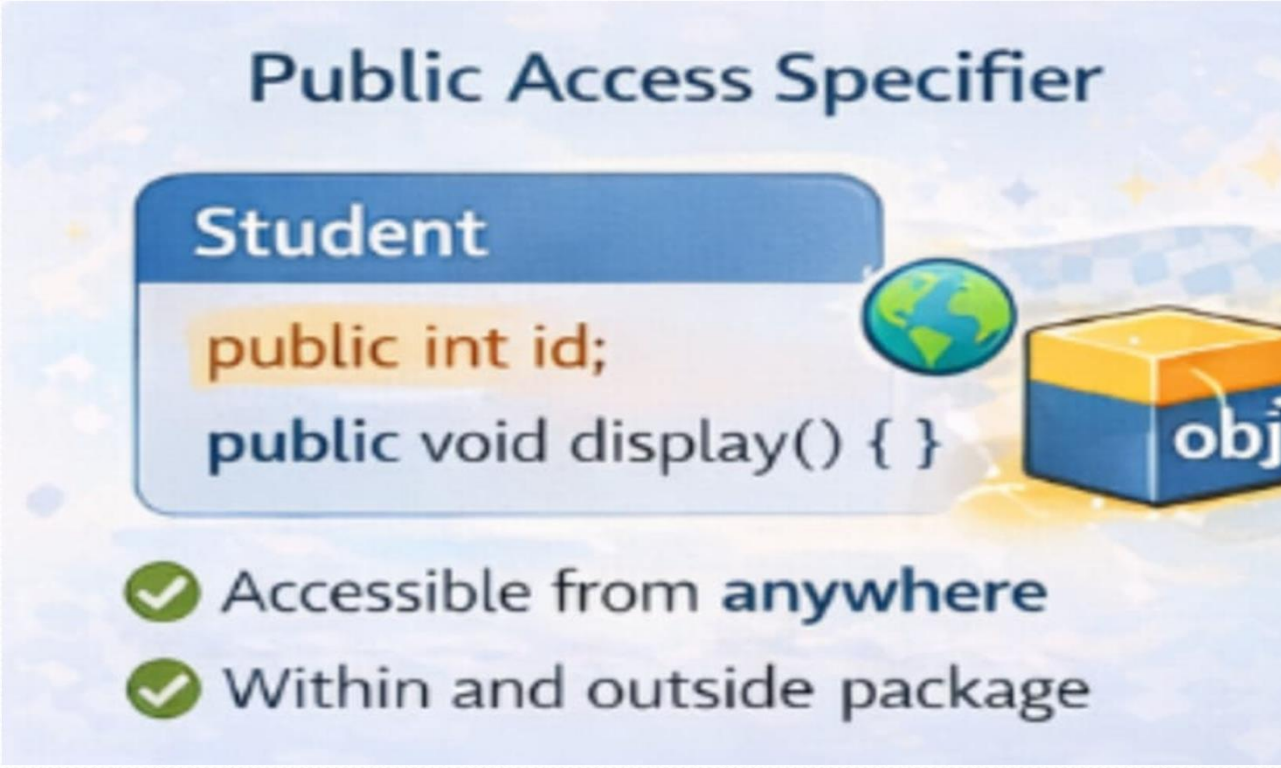

Types of Access Specifiers
Public
Accessible anywhere
Private
Within same class only
Protected
Public
Default
Within same package

Public Access Specifier

Members declared as public can be accessed from anywhere.

Accessible inside and outside the package.

```
public int id;
public void display() { }
```

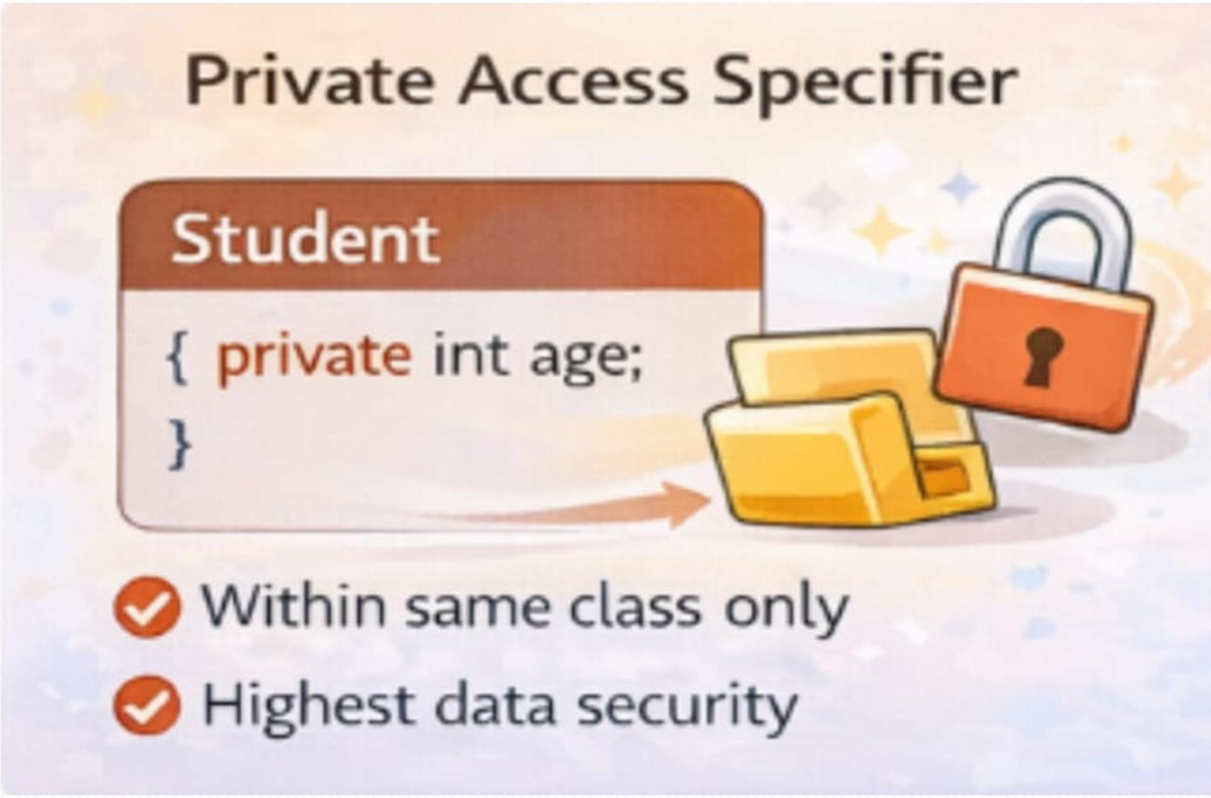


Private Access Specifier

Members declared as private can be accessed only within the same class.

Provides highest level of data security

private int age;

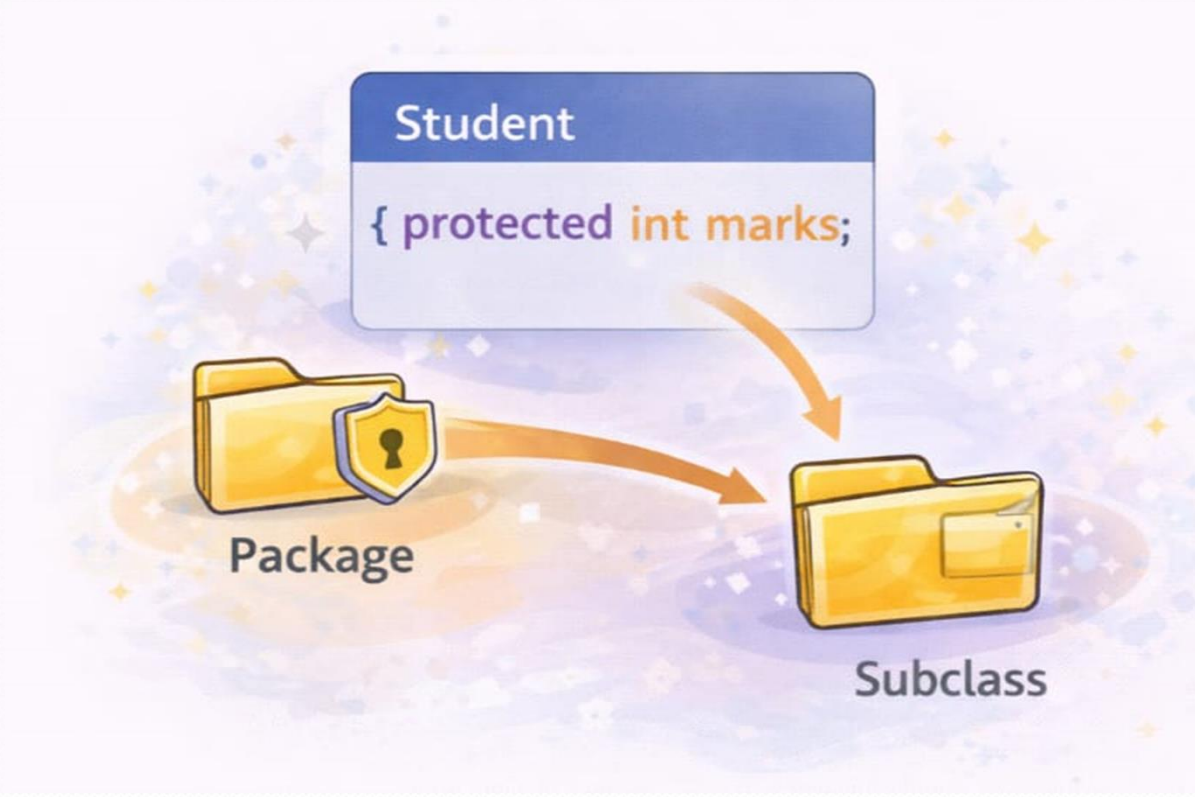


Protected Access Specifier

Accessible within the same package.

Also accessible in subclasses outside the package

protected int marks;

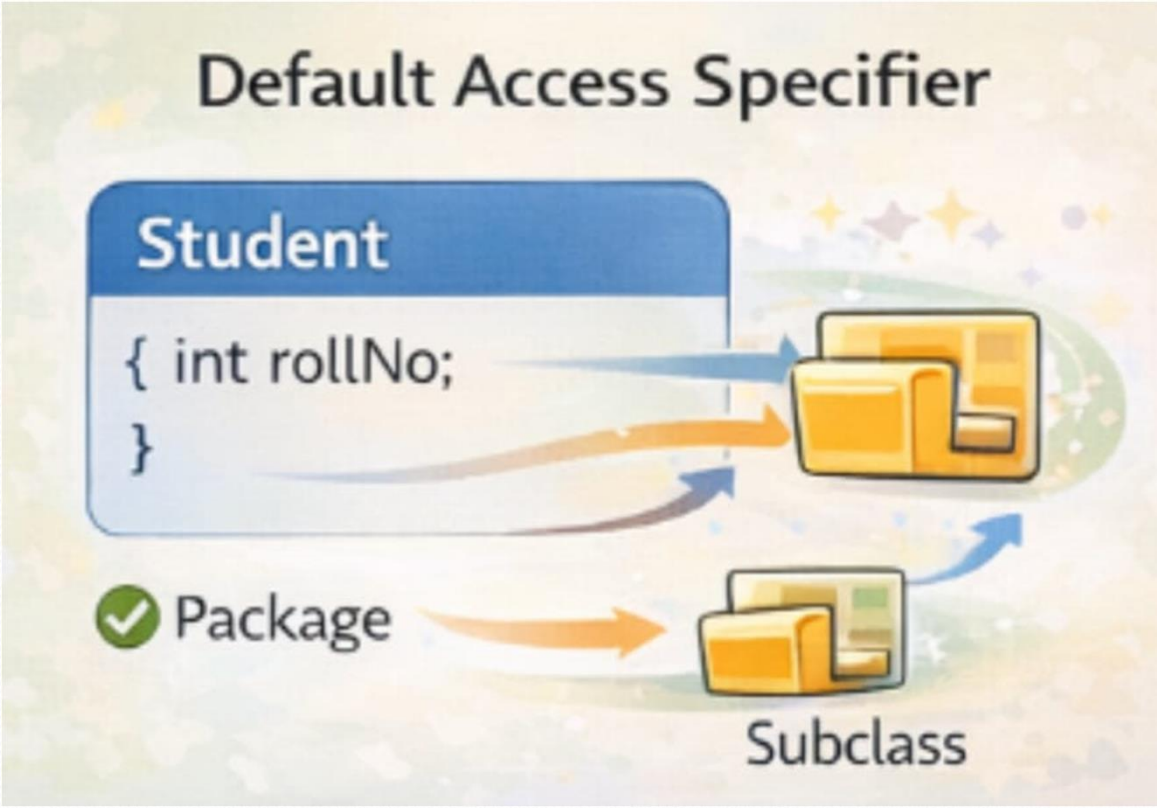


Default Access Specifier

If no access specifier is mentioned, it is default.

Accessible only within the same package

```
int rollNo;
```



Explanation of Access Specifiers

Private

- Accessible only inside the same class.
- Used for data hiding.
- Example: rollNumber
- Cannot be accessed directly outside the class.

Default (No Modifier)

- Accessible only within the same package.
- Example: department

Protected

- Accessible within the same package.
- Also accessible in subclasses (even in different packages).
- Example: college

Public

- Accessible from anywhere in the program.
- Example: name

Access Specifier	Within Class	Same Package	Subclass	Anywhere
private	✓	✗	✗	✗
default	✓	✓	✗	✗
protected	✓	✓	✓	✗
public	✓	✓	✓	✓

Program to Explain Access Specifiers

: AccessSpecifierDemo.java

```
Student {

rivate – accessible only within this class
e int rollNumber = 101;

Default – accessible within the same package
 department = "Computer Science";

Protected – accessible within same package
lso in subclasses (even in different packages)*/
cted String college = "SNS College";

Public – accessible from anywhere
 String name = "Arun";

blic method to access private variable
 void showRollNumber() {
m.out.println("Roll Number (Private): " +
umber);
```

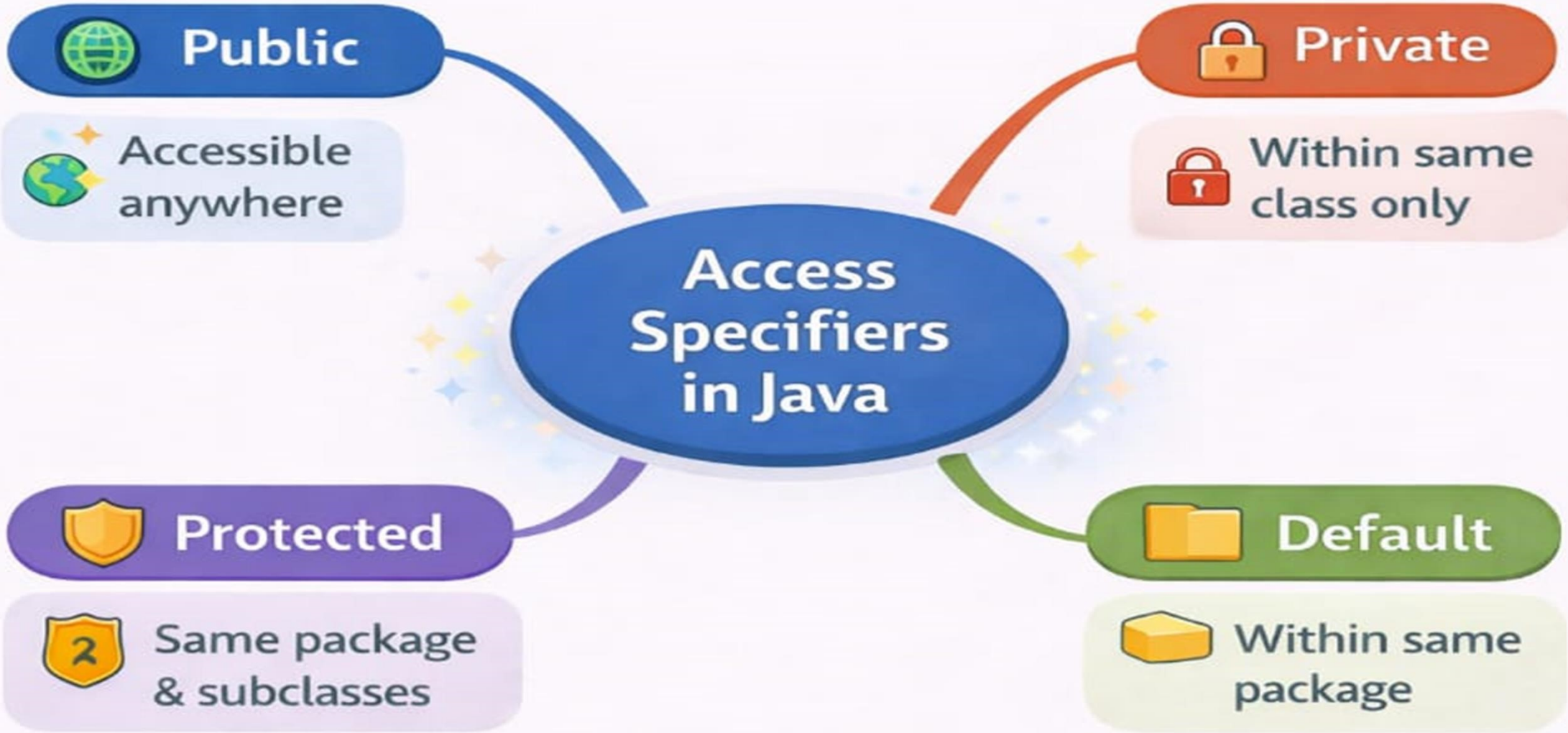
```
public class AccessSpecifierDemo {
public static void main(String[] args) {
Student s = new Student();
// Accessing public variable
System.out.println("Name (Public): " + s.name);
// Accessing protected variable (same package)
System.out.println("College (Protected): " + s.colleg
// Accessing default variable (same package)
System.out.println("Department (Default): " +
s.department);

// Cannot access private variable directly
// System.out.println(s.rollNumber); // ERROR
// Accessing private variable using public method
s.showRollNumber();
}}
```

Output:

```
Name (Public): Arun
College (Protected): SNS College
Department (Default): Computer Science
Roll Number (Private): 101
```

Mind Map


Public
Accessible anywhere
Private
Within same class only
Access Specifiers in Java
Protected
Same package & subclasses
Default
Within same package

Assessment

1. Private members are accessible only within the __.

Answer: class

2. Choose the correct answer.

Which access specifier allows access from anywhere?

a) Private

b) Protected

c) Default

d) Public

3. Define access specifier.

Answer : Access specifiers are keywords used to control the visibility of class members in Java.

References

- https://docs.oracle.com/javase/
- https://www.geeksforgeeks.org/access-modifiers-java/
- https://www.w3schools.com/java/java_modifiers.asp

Thank You 😊