TOPIC: Protected Access Specifier

NAME : Arkapratim Ghosh
Roll No. : 13000121058
Registration No. : 211300100110045

•••

Paper Name: Object Oriented Programming
Paper Code: PCC-CS503
CSE, Sec-A, 5th Sem (2021-2025), CA-1

CONTENT

INTRODUCTION

- Access specifiers control visibility and accessibility of members in Java.
- protected is one of the four access specifiers.
- Members declared as protected have a specific accessibility level.
- Provides controlled access within the same package and to subclasses.
- Accessible within the same package and by subclasses outside the package.
- A key building block for object-oriented programming and design.
- protected offers intermediate access between default and private.

DATA ITEMS, MEMBERS, CONSTRUCTORS AS *protected*

```
public class FirstClass {
    protected String name;
    protected FirstClass(String name) {
        this.name = name;
    }
    protected String getName() {
        return name;
    }
}
```

- ➔ Java class named FirstClass is defined.
- ➔ contains a protected instance variable called name
- ➔ a protected constructor that takes a parameter named name to initialize the name variable
- ➔ protected method named getName() that allows external classes to retrieve the stored name value

ACCESSING *protected* FIELDS

➔ Access from the same package

```
public class SamePackageClass {
    public static void main(String[] args) {
        FirstClass first = new FirstClass("Arkapratim");
        System.out.println("FirstClass name is " +
first.getName());
        first.name = "APG"; // new name
    }
}
```

No errors

ACCESSING *protected* FIELDS (continued)

➔ Access from same package subclass

```
public class SamePackageSubClass extends FirstClass {

        public SamePackageSubClass(String name) {
            super(name);
            System.out.println("SamePackageSubClass name is " +
    this.getName());
            this.name = "APG"; // new name
        }
    }
```

No errors

ACCESSING *protected* FIELDS (continued)

➔ Access from different package

```
public class DifferentPackageClass {
    public static void main(String[] args) {
        FirstClass first = new FirstClass("Arkapratim");
        System.out.println("FirstClass name is "+
first.getName());
        first.name = "APG"; // new name
    }
}
```

- We get *compilation errors*

```
The constructor FirstClass(String) is not visible
The method getName() from the type FirstClass is not visible
The field FirstClass.name is not visible
```

ACCESSING *protected* FIELDS (continued)

➔ Accessing protected fields from different package using subclasses

```
public class SubClass extends FirstClass {
    public SubClass(String name) {
        super(name);
        System.out.println("SubClass name is " + this.getName());
        this.name = "APG"; // new name
    }
}
```

No errors

protected MEMBER ACCESS TABLE

Type / position of class	Access Granted ?
`Inside class`	*Yes*
`Same package class`	*Yes*
`Same package subclass`	*Yes*
`Other package class`	*No*
`Other package subclass`	*Yes*

REFERENCES

1. Accessing Protected Members in Java - GeeksforGeeks
2. Access modifiers in java - Javatpoint
3. java the complete reference, 7th edition -herbert schildt