

```
1 package Subclasses;
2
3 public class TestProtectedAccess {
4     // Main class to test the program
5     public static void main(String[] args) {
6         // Create an object of Dog class
7         Dog dog = new Dog("Buddy");
8
9         // Accessing protected variable and method via subclass
10        dog.dogInfo(); // This will access the protected members of Animal class
11
12        // Accessing protected member outside subclass is not allowed, e.g.,:
13        // System.out.println(dog.name); // Error: name has protected access in Animal
14        // dog.makeSound(); // Error: makeSound() has protected access in Animal
15    }
16
17
18
```

Outline

- Subclasses
  - TestProtectedAccess
    - main(String[]) : void

@ Javadoc Declaration Console

```
<terminated> TestProtectedAccess [Java Application] C:\Users\josej\Downloads\spring-tool-suite-4-4.26.0.RELEASE-e4.33.0-win32.win32.x86_64\sts-4.26.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32
Dog's name: Buddy
Buddy barks.
```

```
1 package Subclasses;
2
3 public class Vaccination {
4     public static void main(String[] args) {
5         // Creating a VaccinationSuccessful object with age and nationality
6         VaccinationSuccessful user1 = new VaccinationSuccessful(25, "Indian");
7
8         // Calling first dose method
9         user1.firstDose();
10
11        // Simulate the user taking the first dose
12        boolean firstDoseTaken = true; // Assume the first dose was taken
13
14        // Calling second dose method
15        user1.secondDose(firstDoseTaken);
16
17        // Calling booster dose method
18        user1.boosterDose();
19    }
20 }
21
22
```

Outline ×

- Subclasses
  - Vaccination
    - main(String[]) : void

@ Javadoc Declaration Console ×

```
<terminated> Vaccination [Java Application] C:\Users\josej\Downloads\spring-tool-suite-4-4.26.0.RELEASE-e4.33.0-win32.win32.x86_64\sts-4.26.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64
You are eligible for the first dose.
You need to pay 250 INR.
You are eligible for the second dose.
You are eligible for the booster dose.
```

Writable

Smart Insert

20 : 6 : 664



Search

ENG  
IN11:04  
26-11-2024

```
1 package Subclasses;
2
3 public class AnimalSoundTest {
4     // Main class to test the program
5     public static void main(String[] args) {
6         // Creating objects of Animal, Dog, and Cat
7         Animal animal = new Animal();
8         Dog dog = new Dog();
9         Cat cat = new Cat();
10
11         // Calling the makeSound() method for each object
12         System.out.println("Animal Sound:");
13         animal.makeSound(); // Calls Animal's makeSound
14
15         System.out.println("\nDog Sound:");
16         dog.makeSound(); // Calls Dog's overridden makeSound
17
18         System.out.println("\nCat Sound:");
19         cat.makeSound(); // Calls Cat's overridden makeSound
20     }
21 }
22
23
```

Outline ×

- Subclasses
  - AnimalSoundTest
    - main(String[]) : void

@ Javadoc Declaration Console ×

<terminated> AnimalSoundTest [Java Application] C:\Users\josej\Downloads\spring-tool-suite-4-4.26.0.RELEASE-e4.33.0-win32.win32.x86\_64\sts-4.26.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64.jre\bin\java.exe

Dog Sound:  
The dog barks.

Cat Sound:  
The cat meows.

Writable

Smart Insert

22 : 1 : 695



Search



ENG  
IN



10:55  
26-11-2024

```
HillStations.java  HillStationTest.java x
// main class to run the program
6 public static void main(String[] args) {
7     // Creating objects of HillStations (Superclass)
8     HillStations hillStation = new HillStations();
9
10    // Calling methods of HillStations (Superclass)
11    System.out.println("From HillStations class (Superclass):");
12    hillStation.location();
13    hillStation.famousFor();
14
15    // Creating objects of subclasses
16    HillStations manali = new Manali();
17    HillStations mussoorie = new Mussoorie();
18    HillStations gulmarg = new Gulmarg();
19
20    // Calling methods of each subclass (demonstrating polymorphism)
21    System.out.println("\nFrom Manali subclass:");
22    manali.location();
23    manali.famousFor();
24
25    System.out.println("\nFrom Mussoorie subclass:");
26    mussoorie.location();
27    mussoorie.famousFor();
28
29    System.out.println("\nFrom Gulmarg subclass:");
30    gulmarg.location();
31    gulmarg.famousFor();
32 }
33 }
34
35
36
```

Outline x

Subclasses  
✓ HillStationTest  
    main(String[]) : void

@ Javadoc Declaration Console x

&lt;terminated&gt; HillStationTest [Java Application] C:\Users\josej\Downloads\spring-tool-suite-4-4.26.0.RELEASE-e4.33.0-win32.win32.x86\_64\sts-4.26.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64.jre\bin\java.exe

From HillStations class (Superclass):

Location: Generic Hill Station

Famous for: Scenic Beauty

From Manali subclass:

Location: Manali, Himachal Pradesh

Famous for: Adventure Sports, Solang Valley

Writable

Smart Insert

35 : 1 : 1112