

JAVA – Access specifiers



```
class Car
   int mileage;
   int max_speed;
   public:
      void display();
```

Data Encapsulation - Wrapping the data and functions in one single unit



Abstraction - hiding irrelevant details from the user.

Access specifiers are the main pillar of implementing abstraction.

Java Package:

- Is a group of similar types of classes, interfaces and sub-packages
- built-in package and user-defined package
- built-in packages such as java, lang, awt, javax, swing, net, io, util, sql etc.(14 packages)

```
Simple example of java package
    package mypack;
    public class Simple
    {
          public static void main(String args[])
          {
                System.out.println("Welcome to package");
          }
     }
}
```



Access package from another package:

Three ways to access the package from outside the package.

import package.*;

import package.classname;

fully qualified name.

Using packagename.*

```
//save by B.java
package mypack;
import pack.*;
class B
{
    public static void main(String args[])
    {
        A obj = new A();
        obj.msg();
    }
}
```



Output: Hello

Using packagename.classname

```
//save by A.java
package pack;
public class A
       public void msg()
                   System.out.println("Hello");
//save by B.java
package mypack;
import pack.A;
class B
 public static void main(String args[])
                    A obj = new A();
                   obj.msg();
```

Output: Hello

Using fully qualified name

```
//save by A.java
package pack;
public class A
     public void msg()
              System.out.println("Hello");
//save by B.java
package mypack;
class B
  public static void main(String args[])
              pack.A obj = new pack.A(); //using fully qualified name
               obj.msg();
```

Output: Hello



All members of a class can be accessed:

Within the class - No restriction.

From outside the class - ?



- Defines the access control
- 1. Public
- 2. Private
- 3. Protected
- 4. Default

 Access specifiers define how a member's variables and member's functions of a class can be accessed from outside the class.



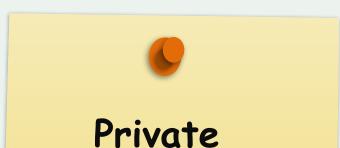
Private

Access Denied - Outsiders





```
class A
private int data;
int a;
public int b;
private void msg(){
        System.out.println("Hello java"); }
public class Simple
   public static void main(String args[]){
       A obj=new A();
        System.out.println(obj.data);
                 //Compile Time Error
        obj.msg();//Compile Time Error
        obj.data=40; // compile time error
        obj.a=10;
        obj.b=20;
```



TTIVUTE

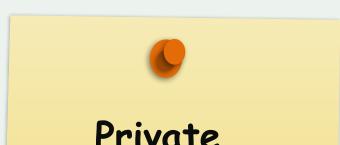
Access Denied - Outsiders

Access Specifiers





✓ Any object or function outside the class cannot access the private members.



Private

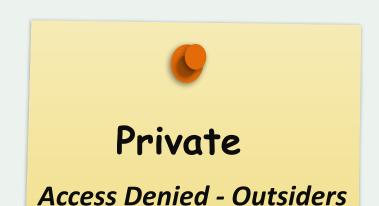
Access Denied - Outsiders





- ✓ Any object or function outside the class cannot access the private members.
- ✓ Can be accessed only by the functions inside the class.

```
public class A
    private String name;
     public String getName()
        return name;
    public void setName(String name)
        this.name = name;
 public class Main
   public static void main(Stringargs[])
      A obj=new A();
      obj.setName("Face");
                                                             Output:
      System.out.println(obj.getName());
                                                             Face
```





- ✓ Any object or function outside the class cannot access the private members.
- ✓ Can be accessed only by the functions inside the class.
- ✓ By default all the members of a class would be private.

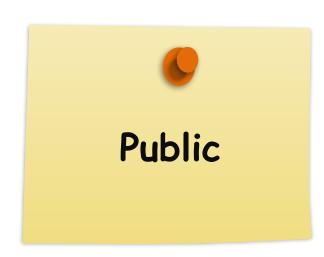




- ✓ A protected member variable or function is very similar to a private member
- ✓ But they can be accessed by any subclass (derived class) of that class.

```
//save by A.java
package pack;
public class A
   protected void msg()
        System.out.println("Hello");
//save by B.java
package mypack;
import pack.*;
class B extends A
    public static void main(String args[])
        B obj = new B();
        obj.msg();
```

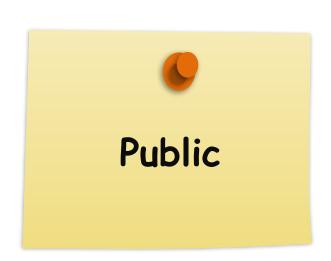
Output: Hello







```
class A
                                                          Output:
    public void msg();
                                                          Hello
       System.out.println("Hello");
class B
  public static void main(String args[])
         A \text{ obj} = \text{new } A();
          obj.msq();
```





- ✔ All the class members declared under public will be available to everyone.
- ✓ Public members can be accessed by other classes too.
- ✓ Can be accessed from anywhere in the program using the direct member access operator (.) with the object of that class.

Accessing Public Data Members:

```
class A
```

public void msg();

System.out.println("Hello");

class B

public static void main(String args[])

A obj = new A();obj.msq();

Accessed using the direct member access (.)

operator with the **object** of that class.









Accessing Default Data Members:

```
//save by A.java
                                       Access only the members of the same
package pack;
class A
                                       package.
    void msg()
        System.out.println("Hello");
//save by B.java
package mypack;
import pack.*;
class B
    public static void main(String args[])
        A obj = new A();//Compile Time Error
        obj.msq();//Compile Time Error
```

1. Which methods can access to private attributes of a class?



a. Only Static methods of the same classb. Only instances of the same classc. Only methods those defined in the same class



d. Only classes available in the same package.

2. What is the output of the following program?



```
class Area {
int width;
int length;
int area;
void area(int width, int length)
this.width = width;
this.length = length;
```

```
class Output {
public static void main(String args[])
{
Area obj = new Area();
obj.area(5, 6);
System.out.println(obj.length
+ " " + obj.width);
}
```

a. 0 0 b. 5 6 c. 6 5 d. 5 5



3. Which of the following statements are incorrect?



- a. Default constructor is called at the time of declaration of the object if a constructor has not been defined.
- b. Constructor can be parameterized.
- c. finalize() method is called when a object goes out of scope and is no longer needed.
- d. finalize() method must be declared protected.

4. What is the output of this program?



```
class Access Control {
class ControlAccess
                                public static void main(String args[])
public int x;
private int y;
                                access obj = new access();
void cal(int a, int b)
                                obj.cal(2, 3);
                                System.out.println(obj.x + " " + obj.y);
x = a + 1;
v =b:
```

a. 3 3 b. 2 3

c. Runtime Error

d. Compilation Error



5. The main method should be static for the reason



- a. It can be accessed easily by the class loader.
- b. It can be accessed by every method or variable without any hindrance.
- c. It can be executed without creating any instance of the class.
- a. None of the above

6. Find the output of following program?

```
class Access{
                                          class Access Control {
                                          public static void main(String args[])
public int x;
private int y;
void cal(int a, int b){
                                          access obj = new access();
                                          obj.cal(2, 3);
x = a + 1;
                                          System.out.println(obj.x);
y = b;
                                          obj.print();
void print() {
system.out.println(" " + y);
```

d. Compilation Error 🗸 c. Runtime Error a. 23 b. 33



```
8. What is the output of this program?

class Output {
 static void main(String args[])
```

```
int x , y = 1;
x = 10;
if (x != 10 \&\& x / 0 == 0)
System.out.println(y);
else
System.out.println(++y);
```

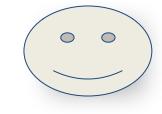
1 b. 2 c. Runtime Error

d. Compilation ror

9. Predict the output

```
public class InitDemo
static int i = demo();
Static
System.out.print(i);
InitDemo(){
System.out.print("hello1");
```

- Compilation error.
- Illegal Argument Exception is thrown at runtime.
- InsideDemo 10 Hello2
- Hello2 InsideDemo 10



public static void main

(String... args)

System.out.print("Hello2");

static int demo(){ System.out.print("InsideDemo");

return 10;

10. What will be the output for the below code?

```
public class Test{
  static{
  int a = 5;
  }
  public static void main(String[] args){
  System.out.println(a);
  }
}
  Compile with error
```

d. Runtime Exception

b. 5 c. 0

11. Find the output of the following program?

- volume = width*length*height; class Area int width;
- int length; int volume; area()
- width=5;
- length=6; void volume()
 - a. 0
 - b. 1
- - - c. 30

class Cons method {

Area obj = new Area();

obj.volume();

d. error 💊

public static void main(String args[])

System.out.println(obj.volume);

12. What is the return type of Constructors?

- a. int
- b. float
- c. void
- d. None of the above



```
14.Find the output of the following program?
```

int height;
int length;
}
class Mainclass {
public static void main(String args[])
{
Box obj = new Box();

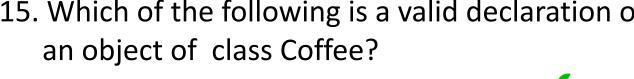
int width;

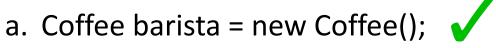
System.out.println(obj);

0 b. 1 c. Runtime error d. Garbage value



15. Which of the following is a valid declaration of an object of class Coffee?







d. new Coffee barista;







