

iQRA National University

Name: Tufail shehZad

ID: 15824

subject: programming

instructor: Muhammad Ayub

program: BS (CS)

final assignment

Q 1^(a) :: You must have seen public private and protected keyword while practising Java program. These are called access modifiers.

An Access modifier restricts the access of a class constructor, data member and method in another class in Java. we have four Access modifiers:

- (1) default
- (2) private
- (3) protected
- (4) public

⇒ Default Access modifier

when we do not mention any access modifier it is called default access modifier.

⇒ private Access modifier

The scope of private is limited to the class only.

(1) private data member and method are only accessible within the class.

(2) class and interface cannot be declared as private.

(3) if a class has private constructor then you cannot create the object of that class from outside of the class.

private access modifier
example in Java

```
class ABC {  
    private double num = 100  
    private int square(int a) {  
        return a*a;  
    }  
}  
  
public class Example {  
    public static void main  
        (String args[]) {
```

```
ABC obj = new ABC()
```

```
System.out.println(obj.num)
```

```
System.out.println
```

```
(obj.square(10));
```

```
}
```

```
}
```

Output:

Compile - Time error

Q1 (b) :

// Java program to
illustrate default i

Package p1:

// class Greek is having
default Acces
class Greek

void display()
}

}

// Java program to
illustrate error wh:

// using class from diff
rent package

```
package p2;
```

```
import p1.*;
```

```
// this class is having  
default access class  
Creek New
```

```
{
```

```
public static void  
main (String ar)
```

```
{
```

```
// accessing class  
Creek from
```

```
Creek obj = new Creek()
```

```
obj.display();
```

```
}
```

```
}
```

output:

Compile Time error

Q2 :: public access

Modifier is a specified using the keyword public

The public Access Modifier has the widest scope among all other access modifiers.

⇒ protected access modifier.

Variable, Method and constructors which are declared protected in a superclass can be accessed only by the subclasses in other package or any

Class within the
package of the protected

protected access modifier
Applied to class
and interface.

Q2(b) :: package ABC package

```
public class Addition {
```

```
protected int Add Two  
number (int a int b)
```

```
}
```

```
}
```

```
}
```

```
package XYZ package
```

```
import ABC package
```

```
class Test extends Addition
```

```
{
```

```
public static void main  
(String args []) {
```

```
    Test obj = new Test ();
```

```
    System.out.println
```

```
(obj.addTwoNumbers
```

```
);
```

```
}
```

output :

Q 30: inheritance in Java is a mechanism in which one object acquires all the properties and behaviors of a parent object. it is an important part of oops (object oriented programming system)

The idea behind inheritance in Java is that you can

create new classes that are built upon existing classes.

Q 3 (B) ::

```
class Animal {  
    void eat () System.out.  
    println (" eating .....");  
}
```

```
class Dog extends  
Animal {
```

```
    void bark () System.out.  
    println (" barking .....");  
}
```

```
}
```

```
class TestInheritance {
```

```
    public static void main (String  
    args [] ) {
```

```
Dog d = new Dog();
```

```
d.bark();
```

```
d.eat();
```

```
} }
```

out put :

barking

eating

Q4 :: Polymorphism is the ability of an object to take on many forms. The most common use of polymorphism in OOP occurs when a parent class reference is used to refer to a child class object.

Any Java object that can pass more than one

is - A test is considered to be polymorphism in Java.

All Java objects are polymorphic since

any object will pass the is - A test for their type and

for the class object

it is important to know that the only possible way to access an object is through a reference variable. A reference variable can be of only one type once type once declared.

The type of a reference variable cannot be changed.

Q4 (B) Example of Java
Polymorphism

```
class person {  
    void walk () {
```

```
        System.out.println("Running slow");
```

```
    }
```

```
}
```

```
class Employee extends  
    person {
```

```
    void walk () {
```

```
        System.out.println("Running fast");
```

```
    }
```

```
public static void main
```

```
(String arg[]) {
```

```
    Person p = new Employee()
```

```
// up casting
```

```
    p.walk();
```

```
}
```

```
}
```

Q5: Abstraction is one of the key concepts of object-oriented programming (OOP) language. Its main

goal is to handle complexity by hiding unnecessary

that enables the user to implement more complex logic on top of the provided abstraction

without understanding or even thinking about all the hidden complexity.

That's a very generic concept that's not limited to object-oriented programming. You can find

it everywhere in
the real world.

Q5 (b) = 00

```
// abstract parent class  
abstract class Animal {
```

```
    // abstract method
```

```
    public abstract void sound();
```

```
}
```

```
// Dog class extends  
Animal class
```

```
public class Dog extends  
Animal {
```

```
    public void sound() {
```

```
        System.out.println("woof");
```

```
    }
```

```
public static void main  
(String args[]) {
```

```
Animal obj = new Dog();
```

```
obj.sound();
```

```
}
```

```
}
```

out put woof