P(1)

Name: Altaf Hussain I.D: 16630

program: Bs(SE) SecB

Q No: 1

ANS (a) :

Access modifiers are like entry gates for other classes ie coler can controle what information or data can be accesible by other class, java pride a number of access modifiers to help you set the level of access you wend for classes as well as, field, method, and Constructer in your class. A member has package or default accesable when no accemability modifier is specified.

P(2)

Name: Altaf Hussain

I.D: 16630

Private access modifier:-

The private access modifier is specified using the keyword private.

=> the method or data member declared as private are accessible only within the class in which they are declared

=> Any other class of same package will not be able to access these member.

=> top level classes are interface can not be declared as private because.

1) private only visible within the enclosing class.
2) protected means " only visible within the enclosing calces and enclosing clas.

P(3)

Altaf Hussain I.D : 16630

Default :-

When no access modifiers is specified for a class method or data member. it is said to be is having the default access modifier by default.

The data member, class, or methods which are not declared using any access modifier i.e. having default access modifier are accessible only within the same package.

Altaf Hussain I.D: 16630

ANS: (B):-

```
Class Animal
  {
    protected void display() {
    System.out.println("I am an Eagle")
  }
}

Class Birds extends Animal{
  public static void main(String[] args ){
Birds birds = new Birds ();
  birds.display();
}
```

output :-

I am an Animal.

Question :- 02

ANS (a) :-

Protected access modifier :-

(1) if the mode of access modifiers is protected then it can be accessible for the class itself and incase of Inheritence also accessible for its derive class.

(2) it is inherated for derive class with in range.

(3) it provide less security than private data members.

Public modifier:-

(1) if the mode of access specifer is public then we can access the public member with in the class and outside of the class. (anything or anywhere).

(2) The public member is inherited for derive Class

(3) it does not provide any security for its data member.

ANS → (b)

Programme of Public access modifier.

```
Public class beautiful {

Public void main(){System.out.
                    "I am beautiful"
println("[redacted]");}

}

import pack

class [redacted]ungly{

public static void main(String
                    args[]){

A obj = new A();

obj.main();

}}
```

output:-

I am beautiful

Program of protected access modifier.:

```
Public class multiplication {
protected int multiply Two number(int a, int b) {
    return a*b;
}
}
```

```
import ab package
Class Test extend multiplication {
public static void main( String args[] ) {
    Test obj = new Test();
System.out.println(object.multiply Two number
    (3, 11);
}
}
```

I.D:16630

Altaf Hussain

Question No:03

Ans (a) :

inheritance

inheritance is important pillor of oop. it is the machanisms in java by which one class is allow to inherit the features.

features

Super class: The class whose features are inherited is know as super class (or a base class or a parent class).

Sub class:

The class that inherite the other class is know as sub class (or a derived class, extended class, or child class).

Altaf Hussain I.D : 16630

Types of Inheritance

1) Single inheritance :-

In single inheritance, subclass inherit the features of one superclass. In image below, the class A serves as a base class for the derived class B.

2) Multilevel inheritance :-

In multilevel inheritance, a derived class will be inheriting a base class and as well as the derived class also act as the base class to other class.

3) Hierarchical inheritance :-

In Hierachical Inheritence, one class serves as a superclass (base class) for more than one sub class

PC(11)

Altaf Hussain I.D: 16630

ANS (b) :-

Program of Inheritance of class employee

```
import person.person;
public class Employee extends person
    {// Employee is a child class
Public double salary;
public Employee() {
    super();
}
public Employee(string name, int id,
        String address, double salary)
    { super(name, id, address);
This.salary = salary;
}
```

Output :-

To Create an employee 3 press:
Enter the employee name: Abbass
Enter his salary : 500
Enter his i.d : 200.
Employee { Name: Abbas, id: 200, salary: 500}

ANS:-(a) Q No: 04

Polymorphism

Polymorphism is one of the OOPs feature that allows us to perform a single action in different ways. for example let say we have class Animal that has a method sound(). since these is generic class so we can't give it a implementation like; Roar, Meow, oink, etc. we had a genric message.

Why Polymorphism is needed :-

Polymorphism is the capability of a method to do different things based on the object that it is acting upon. In other words, polymorphism allows you define one interface and multiple implementation. As we have the above example that we seen the method sound().

PC(13)

Altaf Hussain I.D : 16630

Q No : 04 (b)

ANS → (b) :-

```
Class plant {
    void print
system.out.println (" plant class");
}
Class subclass1 extands plant, {
Void plants ()
{
    System.out.println(" subclass");
}
}
Class Subclass extends plants {
    void print ()
{
System.out.println(" sub class 1 ");
}
}
Class Testpolymorphism3 {
public static void main(string[] args)
{
plant a;
a = new subclass1();
a.print();
a = new subclass2();
a.print();
}
}
```

output

Sub Class 1

Sub class 2

Altaf Hussain

I.D :- 16630

Q No5 :- (a)

ANS :-

Abstraction :-

abstraction is one of the key element of good software design.
it help encapsulate behavior.
it help decouple software element.

Importance of abstraction in OOPs.

abstraction make the application extendable in much easier way it makes refactoring much easier.

1) Storing in DB based :-

The DB tables are shared in order to evenly distribute the items.
sharding is done by a hash key modulo function.
These function work on a string.

P(15)

Altaf Hussain

I.d : 16630

Using the key in Maps :-

The second usage for the triplet is mapping the items for fast lookup. so when Not using abstraction the map will not work clearly.

Altaf Hossain

I.D 816630

CQ No:: 5 (b)

ANS ::

Programme of abstraction

```
abstraction class Vehicle
{
    abstraction void start();
}
class Car extends Vehicle
{
    void start()
    {
        System.out.println("car starts with key");
    }
}
Class Scooter Extends Vehicle
{
    void start()
    {
        System.out.println("scooter starts with kick"
    }
    Public static void main(String[] args)
    {
        Car c = new Car();
        c.start();
        Scooter s = new Scooter();
        s.start();
```

P(17)

Altaf Hussain I.D: 16630

output

Car start with key
Scooter start with kick