**Q1.Write a simple Banking System program by using OOPs concept where you can get account Holder name balance etc?\**

**Ans. import** java.util.Scanner;

**class** BankDetails {

**private** String accno;

**private** String name;

**private** String acc\_type;

**private** **long** balance;

    Scanner sc = **new** Scanner(System.in);

    //method to open new account

**public** **void** openAccount() {

        System.out.print("Enter Account No: ");

        accno = sc.next();

        System.out.print("Enter Account type: ");

        acc\_type = sc.next();

         System.out.print("Enter Name: ");

        name = sc.next();

        System.out.print("Enter Balance: ");

         balance = sc.nextLong();

    }

    //method to display account details

**public** **void** showAccount() {

        System.out.println("Name of account holder: " + name);

      System.out.println("Account no.: " + accno);

      System.out.println("Account type: " + acc\_type);          System.out.println("Balance: " + balance);

    }

Public class Account{

Public static int main(String [] args){

BankDetails obj=new BankDetails();

obj.openAccount();

Obj,showAccount();

}

}

**Q2. Write a Program where you inherit method from parent class and show method Overridden Concept?**

**Ans. class** Vehicle{

**void** run(){System.out.println("Vehicle is running");}

}

//Creating a child class

**class** Bike **extends** Vehicle{

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

  //creating an instance of child class

  Bike obj = **new** Bike();

  //calling the method with child class instance

  obj.run();

  }

}

**Q3.Write a program to show run time polymorphism in java?\**

**Ans . class** Bike{

**void** run(){System.out.println("running");}

}

**class** Splendor **extends** Bike{

**void** run(){System.out.println("running safely with 60km");}

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

    Bike b = **new** Splendor();//upcasting

    b.run();

  }

}

**Q4.Write a program to show Compile time polymorphism in java?\**

**Ans.** class **ClassMain**{

void **disp**(int number){

System.out.println ("method:" + number);

}

void **disp**(int number1, int number2){

System.out.println ("method:" + number1 + "," + number2);

}

double **disp**(double number) {

System.out.println("method:" + number);

return num;

}

}

class **CompileTimePolymorphismDemo**

{

public static void **main** (String args [])

{

ClassMain obj = new ClassMain();

double result;

obj.disp(40);

obj.disp(50, 30);

result = obj.disp(5.1);

System.out.println("Answer is:" + result);

}

}

**5. Achieve loose coupling in java by using OOPs  concept?**

**class** A

{

**void** foo()

{

  System.out.println("Inside the foo method of base class.");

}

}

// child or derived class

**class** B **extends** A

{

**void** foo()

{

  System.out.println("Inside the foo method of derived class.");

}

} **public** **class** CouplingExample

{

// main method

**public** **static** **void** main(String argvs[])

{

// creating an object of class B

B obj = **new** B();

obj.foo();

}

}

**Q6. What is the benefit of encapsulation in java?**

**Ans . Encapsulation use to limit the use of integer by using private keyword for instance variable .we can not access them directly in man method.**

**Q7.Is java a t 100% Object oriented Programming language? If no why ?**

**Ans .NO .java is not a 100% OOPS because of the existence of primitive data types, use of static and wrapper classes.**

**Q8.What are the advantages of abstraction in java?**

**Ans .Advantages-**

**1Abstrat class in java is avoids code duplication,**

**2.Enable code reusability.**

**Q9.What is an abstraction explained with an Example?**

**Ans . it denotes a Parents class in which included method don’t have any body.**

**Q10.What is the final class in Java?**

**Ans. Final is a class which is not inherited by child class. The final keyword not allow to do that.**