**CDAC Mumbai PG-DAC August 24**

**Assignment No- 5**

1. Create a base class BankAccount with methods like deposit() and withdraw(). Derive a class SavingsAccount that overrides the withdraw() method to impose a limit on the withdrawal amount. Write a program that demonstrates the use of overridden methods and proper access modifiers & return the details.

package que1.assignment\_5;

import java.util.Scanner;

public class BankAccount{

protected float Current\_balance;

protected float deposit;

protected float withdraw;

static final int Paswword =9905;

public void deposit()

{

this.Current\_balance=Current\_balance+deposit;

setCurrent\_balance(this.Current\_balance);

}

public void withdraw()

{

Current\_balance=Current\_balance-withdraw;

setCurrent\_balance(Current\_balance);

}

public float getCurrent\_balance() {

return Current\_balance;

}

public void setCurrent\_balance(float current\_balance) {

Current\_balance = current\_balance;

}

public float getDeposit() {

return deposit;

}

public void setDeposit(float deposit) {

this.deposit = deposit;

}

public float getWithdraw() {

return withdraw;

}

public void setWithdraw(float withdraw) {

this.withdraw = withdraw;

}

}

class SavingAccount extends BankAccount

{

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

//SavingAccount save = new SavingAccount();

public SavingAccount(){

}

public void deposit()

{

System.out.println("Enter Amount To Deposit: ");

setDeposit(sc.nextFloat());

if(getDeposit()>0) {

super.deposit();

System.out.println("Current Balance: "+getCurrent\_balance());

// this.Current\_balance=Current\_balance+deposit;

// setCurrent\_balance(this.Current\_balance);

}

}

public void withdraw()

{

System.out.println("Enter Amount To Withdraw: ");

setWithdraw(sc.nextFloat());

if(getWithdraw()<=100000 && getWithdraw()<=Current\_balance)

{

super.withdraw();

System.out.println("Current Balance: "+getCurrent\_balance());

// Current\_balance=Current\_balance-withdraw;

// setCurrent\_balance(Current\_balance);

}else

System.out.print("Insufficent Balance !!!!\nPlease Add Balance to Your Account\nCurrent Balance: "+getCurrent\_balance()+"\n\n\n");

}

public int security\_check()

{

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

return sc.nextInt();

}

public static int choise()

{

System.out.println("Select Operation: \n0.Exit\n1.Deposit\n2.Withdraw");

return sc.nextInt();

}

public void MasterMind()

{

if (security\_check()==SavingAccount.Paswword)

{

int ch;

while((ch=SavingAccount.choise())!=0) {

switch (ch) {

case 1:

deposit();

break;

case 2:

withdraw();

break;

default:

break;

}

}

}else

{

System.out.println("Invalid Password!!!");

}

}

}

package que1.assignment\_5;

public class mainclass {

public static void main(String[] args) {

SavingAccount SA = new SavingAccount();

for(int i=1;i<4;i++)

{

SA.MasterMind();

}

System.out.println("Attempts Are Over!!\nYou Can Try After 4 Hours.");

}

}

1. Create a base class Vehicle with attributes like make and year. Provide a constructor in Vehicle to initialize these attributes. Derive a class Car that has an additional attribute model and write a constructor that initializes make, year, and model. Write a program to create a Car object and display its details.

package Assignment\_5\_2;

import java.util.Scanner;

public class Car extends Vehicle{

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

protected String Car\_Name;

protected String Car\_Type;

public Car()

{

}

public Car(String car\_Name, String car\_Type,String model\_launch\_year,String origin\_company) {

super(model\_launch\_year, origin\_company);

Car\_Name = car\_Name;

Car\_Type = car\_Type;

}

public String getCar\_Name() {

return Car\_Name;

}

public void setCar\_Name(String car\_Name) {

Car\_Name = car\_Name;

}

public String getCar\_Type() {

return Car\_Type;

}

public void setCar\_Type(String car\_Type) {

Car\_Type = car\_Type;

}

public void accept\_record()

{

System.out.println("Enter Car Name : ");

sc.next();

setCar\_Name(sc.nextLine());

System.out.println("Enter Car Type : ");

setCar\_Type(sc.nextLine());

System.out.println("Enter Car Model Launch Year : ");

setModel\_launch\_year(sc.nextLine());

System.out.println("Enter Car Origin Company Name : ");

setOrigin\_company(sc.nextLine());

}

public void print\_record()

{

System.out.println("Enter Car Name : "+getCar\_Name());

System.out.println("Enter Car Type : "+getCar\_Type());

System.out.println("Enter Car Model Launch Year : "+getModel\_launch\_year());

System.out.println("Enter Car Origin Company Name : "+getOrigin\_company());

}

public static select choise()

{

System.out.println("0.Exit\n1.Accept\n2.Print\nEnter Coice: ");

return select.values()[sc.nextInt()];

}

}

package Assignment\_5\_2;

public class Vehicle {

protected String model\_launch\_year;

protected String origin\_company;

public Vehicle(){

}

public Vehicle(String model\_launch\_year, String origin\_company) {

this.model\_launch\_year = model\_launch\_year;

this.origin\_company = origin\_company;

}

public String getModel\_launch\_year() {

return model\_launch\_year;

}

public void setModel\_launch\_year(String model\_launch\_year) {

this.model\_launch\_year = model\_launch\_year;

}

public String getOrigin\_company() {

return origin\_company;

}

public void setOrigin\_company(String origin\_company) {

this.origin\_company = origin\_company;

}

}

package Assignment\_5\_2;

public class mainclass {

public static void main(String[] args) {

select ch;

Car c=new Car();

while((ch=Car.choise())!=select.Exit)

{

switch (ch) {

case Accept:

c.accept\_record();

break;

case Print:

c.print\_record();

break;

default:

break;

}

}

}

}

package Assignment\_5\_2;

public enum select {

Exit,Accept,Print

}

1. Create a base class Animal with attributes like name, and methods like eat() and sleep(). Create a subclass Dog that inherits from Animal and has an additional method bark(). Write a program to demonstrate the use of inheritance by creating objects of Animal and Dog and calling their methods.

package Assignment\_5\_3;

public abstract class Animal {

protected String animalName;

public abstract void animaltype();

public abstract void soundmake();

}

package Assignment\_5\_3;

import java.util.Scanner;

public class Animal\_Info {

public static void main(String[] args) {

Scanner sc =new Scanner (System.in);

System.out.println("Slect Aniamal\n1.Dog\n2.Lion");

select ch = select.values()[sc.nextInt()];

System.out.println(ch);

Animal A=null;

//While(ch!=select.Exit)

{

switch(ch) {

case Dog:

{

A = new Dog();//upcasting

} break;

case Lion:

{

A = new Lion(); //upcasting

}

break;

}

}

if(A!=null) {

A.animaltype();// it will call overridden methods

A.soundmake();

}

sc.close();

}

}

package Assignment\_5\_3;

public class Dog extends Animal{

@Override

public void animaltype()

{

System.out.println("Dog is a Pet type animal ");

}

@Override

public void soundmake()

{

System.out.println("Dog make sound like barking ");

}

}

package Assignment\_5\_3;

public class Lion extends Animal{

@Override

public void animaltype()

{

System.out.println("Lion is a Wild animal");

}

@Override

public void soundmake()

{

System.out.println("Lion make sound like roaring");

}

}

package Assignment\_5\_3;

public enum select {

Exit,Dog,Lion;

}

1. Build a class Student which contains details about the Student and compile and run its

instance.

package Assignment\_5\_4;

public class ClassRoom {

protected String stdName;

protected int age;

protected int rollNo;

public String getStdName() {

return stdName;

}

public void setStdName(String stdName) {

this.stdName = stdName;

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

public int getRollNo() {

return rollNo;

}

public void setRollNo(int rollNo) {

this.rollNo = rollNo;

}

}

package Assignment\_5\_4;

import java.util.Scanner;

public class Student extends ClassRoom {

Scanner sc=new Scanner (System.in);

public void Entry()

{

System.out.println("Enter The name of student");

setStdName(sc.nextLine());

System.out.println("Enter The age of student");

setAge(sc.nextInt());

System.out.println("Enter The Id of student");

setRollNo(sc.nextInt());

}

@Override

public String toString() {

String str= "Student [stdName=" + stdName + ", age=" + age + ", rollNo=" + rollNo + "]";

return str;

}

}

1. Write a Java program to create a base class Vehicle with methods startEngine() and stopEngine(). Create two subclasses Car and Motorcycle. Override the startEngine() and stopEngine() methods in each subclass to start and stop the engines differently.

package Assignment\_5\_5;

public class Car extends Vehicle {

@Override

public void startEngine()

{

System.out.println("Engine of Car Started");

}

public void stopEngine()

{

System.out.println("Engine of Car Started");

}

}

package Assignment\_5\_5;

public class Master {

public static void main(String[] args) {

Car C=new Car();

C.startEngine();

C.stopEngine();

Motorcycle M =new Motorcycle();

M.startEngine();

M.stopEngine();

}

}

package Assignment\_5\_5;

public class Motorcycle extends Vehicle {

@Override

public void startEngine()

{

System.out.println("Engine of MoterCycle Started");

}

public void stopEngine()

{

System.out.println("Engine of MoterCycle Started");

}

}

package Assignment\_5\_5;

public abstract class Vehicle {

public abstract void startEngine();

public abstract void stopEngine();

}