**JAVA ASSIGNMENT DAY-2**

1.Write a Java program to create a class called "Person" with a name and age attribute. Create two instances of the "Person" class, set their attributes using the constructor, and print their name and age.

package com.selva;

public class Person {

String name;

int age;

Person(String name,int age){

this.name=name;

this.age=age;

}

public static void main(String[] args) {

// **TODO** Auto-generated method stub

Person obj=new Person("Gayathri",21);

System.***out***.println(obj.name);

System.***out***.println(obj.age);

}

}

2. Write a Java program to create a class called "Dog" with a name and breed attribute. Create two instances of the "Dog" class, set their attributes using the constructor and modify the attributes using the setter methods and print the updated values.

package com.selva;

public class Dog {

String name;

String breed;

Dog(String name,String breed){

this.name=name;

this.breed=breed;

}

void setName(String name) {

this.name=name;

}

void setBreed(String breed) {

this.breed=breed;

}

void display() {

System.***out***.println("Name:" +name +" Breed:" + breed);

}

public static void main(String[] args) {

// **TODO** Auto-generated method stub

Dog obj=new Dog("Tinku","Labrador");

obj.setName("Jack");

obj.setBreed("German Shepherd");

obj.display();

}

}

3. Write a Java program to create a class called "Rectangle" with width and height attributes. Calculate the area and perimeter of the rectangle.

package com.selva;

public class Rectangle {

int width;

int height;

void area(int width,int height) {

int a=width\*height;

System.***out***.println("Area : "+a);

}

void perimeter(int width,int height) {

int p=2\*(width+height);

System.***out***.println("Perimeter : "+p);

}

public static void main(String[] args) {

Rectangle obj=new Rectangle();

obj.area(20, 15);

obj.perimeter(20,15);

}

}

4. Write a Java program to create a class called "Circle" with a radius attribute. You can access and modify this attribute. Calculate the area and circumference of the circle.

package com.selva;

public class Circle {

double radius;

public Circle(double radius) {

this.radius = radius;

}

public void setRadius(double radius) {

this.radius = radius;

}

public double calculateArea() {

return Math.***PI*** \* radius \* radius;

}

public double calculateCircumference() {

return 2 \* Math.***PI*** \* radius;

}

public void printDetails() {

System.***out***.println("Circle Radius: " + radius);

System.***out***.println("Area: " + calculateArea());

System.***out***.println("Circumference: " + calculateCircumference());

}

public static void main(String[] args) {

Circle obj = new Circle(5.0);

System.***out***.println("Initial Details:");

obj.printDetails();

obj.setRadius(7.5);

System.***out***.println("\nUpdated Details:");

obj.printDetails();

}

}

5. Write a Java program to create a class called "Book" with attributes for title, author, and ISBN, and methods to add and remove books from a collection.

package com.selva;

import java.util.\*;

public class Book {

String title;

String author;

String ISBN;

HashMap<String,String> mp=new HashMap<>();

Book(String title,String author,String ISBN)

{

this.author=author;

this.ISBN=ISBN;

this.title=title;

}

void add()

{

if(!mp.containsKey(this.title))

{

mp.put(this.author,this.title+" "+this.ISBN);

System.***out***.println("Added Successfully");

System.***out***.println(this.author+" "+this.title+" "+this.ISBN);

}

else

System.***out***.println("Already exists");

}

public static void main(String[] args) {

Book b=new Book("The Arabian Nights","Antoine Galland","978-3-16-148410-0");

b.add();

}}

6. Write a Java program to create a class called "Employee" with a name, job title, and salary attributes, and methods to calculate and update salary.

package com.selva;

public class Employees {

double baseSalary;

double bonus;

double commission;

double overtimePay;

double allowances;

double deductions;

double salary;

Employees(double baseSalary, double bonus, double commission, double overtimePay, double allowances, double deductions) {

this.baseSalary = baseSalary;

this.bonus = bonus;

this.commission = commission;

this.overtimePay = overtimePay;

this.allowances = allowances;

this.deductions = deductions;

}

void calcSalary()

{

this.salary=(this.baseSalary+this.allowances+this.bonus+this.commission+this.overtimePay)-this.deductions;

}

void updateSalary(double baseSalary, double bonus, double commission, double overtimePay, double allowances, double deductions)

{

this.salary=(baseSalary+allowances+bonus+commission+overtimePay)-deductions;

}

void display()

{

System.***out***.println("Salary: "+this.salary);

}

public static void main(String[] args)

{

Employees e=new Employees(55500,32900,2929.23,4920.29,29202.29,59002.19);

System.***out***.println("Before: ");

e.calcSalary();

e.display();

System.***out***.println("After: ");

e.updateSalary(27289, 891.339, 82890.29, 28919.39, 29920.290, 9992.22);

e.display();

}}

7. Write a Java program to create a class called "Bank" with a collection of accounts and methods to add and remove accounts, and to deposit and withdraw money. Also define a class called "Account" to maintain account details of a particular customer.

package com.selva;

import java.util.\*;

class Account

{

String userName;

String userAccNo;

Account(String userName,String userAccNo)

{

this.userName=userName;

this.userAccNo=userAccNo;

}

}

public class Bank {

HashMap<String,Double> acc=new HashMap<>();

double balance=0.0;

void addAccount(String accNo)

{

if(!acc.containsKey(accNo))

{

acc.put(accNo,this.balance);

System.***out***.println("Account added");

}

else

System.***out***.println("Account already exists");

}

void deleteAccount(String accNo)

{

if(acc.containsKey(accNo))

acc.remove(accNo);

else

System.***out***.println("No account found");

}

void deposit(String accNo,double mon)

{

if(acc.containsKey(accNo)) {

System.***out***.println("Before Deposit: ");

this.getBalance(accNo);

acc.put(accNo, acc.get(accNo)+mon);

System.***out***.println("After Deposit: ");

this.getBalance(accNo);

}

else

{

System.***out***.println("Don't have account");

}

}

void withdraw(String accNo,double mon)

{

if(acc.containsKey(accNo) && acc.get(accNo)>mon) {

System.***out***.println("Before Withdraw: ");

this.getBalance(accNo);

acc.put(accNo, acc.get(accNo)-mon);

System.***out***.println("After Withdraw: ");

this.getBalance(accNo);

}

else

{

System.***out***.println("Balance: "+acc.get(accNo));

}

}

void getBalance(String accNo)

{

System.***out***.println("Account Number: "+accNo);

System.***out***.println("Balance : "+acc.get(accNo));

}

public static void main(String[] args) {

Account acc=new Account("Samuvel","289939");

Bank b=new Bank();

b.addAccount(acc.userAccNo);

b.deposit(acc.userAccNo, 40030);

b.withdraw(acc.userAccNo, 100);

}

}