**Topic 1: JVM Concepts and Language Basics**

**Assignment 1:**

Write a java program to display “Welcome to Java Programming” and then print your name on a separate line.

public class Name {  
  
 public static void main(String args[])  
 {  
 System.*out*.print("Welcome to Java Programming\nMeghana Prabhakar Karekar" );  
  
 }  
  
}

o/p



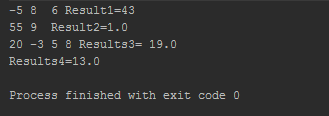
**Assignment 2:**

Write a Java program to print the result of the following operations. Declare variables and initialize them with given values a. -5 + 8 \* 6 b. (55+9) % 9 c. 20 + -3\*5 / 8 d. 5 + 15 / 3 \* 2 - 8 % 3

public class Print\_result {  
 public static void main(String args[])  
 {  
 int result1;  
 double result2,result3,result4;;  
 int a=-5,b=8,c=6,d=55,e=9,f=20,g=-3,h=5,i=15,j=3,k=2,l=8;  
 result1=a+b\*c;  
 result2=(d+e)%e;  
 result3=f+g\*h/b;  
 result4=h+i/j\*k-b%j;  
  
 System.*out*.println(a+" "+b+" "+" "+c+" " +"Result1="+result1);  
 System.*out*.println(d+" "+e+" "+" "+"Result2="+result2);  
 System.*out*.println(f+" "+g+" "+h+" "+b+" "+"Results3= "+result3);

System.*out*.println("Results4="+result4);

}

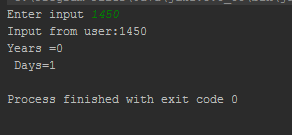
  
}

**Assignment 3:**

Write a Java program to convert minutes into a number of years and days.

import java.util.Scanner;  
  
public class Minutes\_days {  
 public static void main(String args[]) {  
  
 double minutes\_in\_year=60\*24\*365;  
 Scanner input=new Scanner(System.*in*);  
  
 System.*out*.print("Enter input ");  
 double minutes= input.nextDouble();  
  
 long years=(long)(minutes/minutes\_in\_year);  
 int days=(int)(minutes/60/24)%365;  
 System.*out*.println("Input from user:"+(int)minutes+"\nYears ="+years+"\n Days="+days);  
 }  
  
}

o/p

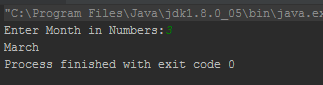


**Assignment 4:**

Write a program to print month in words, based on input month in numbers.(using switch case)

import java.util.Scanner;  
  
public class Month {  
 public static void main(String args[])  
 {  
 double month=0;  
  
 Scanner input=new Scanner(System.*in*);  
 System.*out*.print("Enter Month in Numbers:");  
 month=input.nextDouble();  
  
 switch((int) month)  
 {  
 case 1:  
 System.*out*.print("January");  
 break;  
 case 2:  
 System.*out*.print("February");  
 break;  
 case 3:  
 System.*out*.print("March");  
 break;  
 case 4:  
 System.*out*.print("April");  
 break;  
 case 5:  
 System.*out*.print("May");  
 break;  
 case 6:  
 System.*out*.print("June");  
 break;  
 case 7:  
 System.*out*.print("July");  
 break;  
 case 8:  
 System.*out*.print("August");  
 break;  
 case 9:  
 System.*out*.print("September");  
 break;  
 case 10:  
 System.*out*.print("October");  
 break;  
 case 11:  
 System.*out*.print("November");  
 break;  
 case 12:  
 System.*out*.print("December");  
 break;  
  
 }  
  
  
 }  
}

Output:

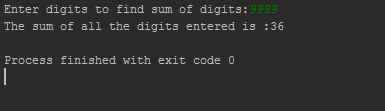


**Assignment 5:**

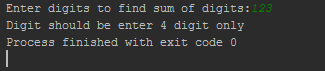
Write a program that will accept a 4 digit number(assume that the user enters only 4 digit nos.) and print the sum of all the 4 digits. For ex : If the number passed is 3629, the program should print “The sum of all the digits entered is 20”

import java.util.Scanner;  
  
public class Sum\_of\_digit {  
 public static void main(String args[])  
 {  
 int sum=0,digit=0,results=0;  
 Scanner input=new Scanner(System.*in*);  
 System.*out*.print("Enter digits to find sum of digits:");  
 digit= (int) input.nextDouble();  
  
 if(digit>999 && digit<=9999)  
 {  
 while (digit != 0) {  
 sum = sum + digit % 10;  
 digit = digit / 10;  
  
 }  
 System.*out*.println("Sum of Digit:" + sum);  
 }  
 else {  
 System.*out*.print("Digit should be enter 4 digit only");  
 }  
  
  
 }  
}

Output:



If input is 3 digit:

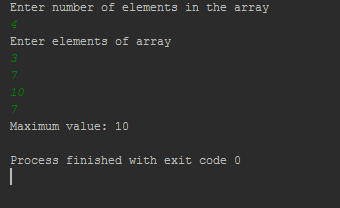


**Assignment 6:**

Write a program to find greatest number in an array

import java.util.Scanner;  
  
public class Largest\_Number {  
 public static void main(String args[])  
 {  
 int n,max;  
 Scanner s=new Scanner(System.*in*);  
 System.*out*.println("Enter number of elements in the array");  
 n=s.nextInt();  
 int array\_numbers[]=new int[n];  
 System.*out*.println("Enter elements of array");  
 for(int i=0;i<n;i++)  
 {  
 array\_numbers[i]=s.nextInt();  
  
 }  
 max=array\_numbers[0];  
 for(int i=0;i<n;i++)  
 {  
 if(max <array\_numbers[i])  
 {  
 max=array\_numbers[i];  
 }  
  
 }  
 System.*out*.println("Maximum value: "+max);  
 }  
}

Output:



**Topic 2: Object Oriented Concepts**

**Assignment 1:**

Write a program to create a class Book with the following

- attributes: -isbn, title, author, price

- methods :

i. Initialize the data members through parameterized constructor

ii. displaydeta ils() to display the details of the book

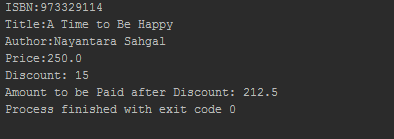
iii. discountedprice() : pass the discount percent, calculate the discount on price and find the amount to be paid after discount

- task :

Create an object book, initialize the book and display the details along with the discounted price

class Book  
{  
 long isbn;float price;  
 String title, author;  
  
 Book(long isbn, String title, String author, float price) {  
 this.isbn = isbn;  
 this.title = title;  
 this.author = author;  
 this.price = price;  
  
  
 }  
  
 void displaydetails() {  
 System.*out*.println("ISBN:" + isbn);  
 System.*out*.println("Title:" + title);  
 System.*out*.println("Author:" + author);  
 System.*out*.println("Price:" + price);  
 }  
  
 void discountedprice(int discount) {  
 float Discounted\_price = price-(price \* discount/100);  
 System.*out*.print("Discount: " + discount + "\nAmount to be Paid after Discount: " +Discounted\_price);  
  
 }  
  
}  
  
public class Book\_details  
{  
 public static void main(String args[])  
 {  
 Book mybook=new Book(973329114,"A Time to Be Happy","Nayantara Sahgal",250);  
 mybook.displaydetails();  
 mybook.discountedprice(15);  
 }  
 }

Output



**Assignment 3:**

Write a program to create a class Book with the following data members: isbn, title and price. Inherit the class Book to two derived classes : Magazine and Novel with the following data members:

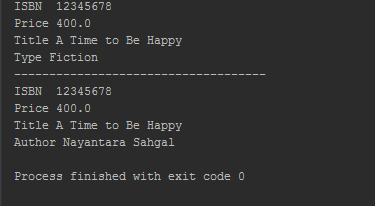
Magazine: type

Novel : author

Populate the details using constructors.

Create a magazine and Novel and display the details.

class Book {  
 long isbn=12345678;  
 float price=400;  
 String title="A Time to Be Happy";  
 Book()  
  
 {  
 System.*out*.println("ISBN "+isbn);  
 System.*out*.println("Price "+price);  
 System.*out*.println("Title "+title);  
  
  
 }  
  
  
}  
  
class Magazine extends Book  
{  
 String type;  
 Magazine(String type)  
 {  
  
 System.*out*.println("Type "+type);  
 }  
  
}  
class Novel extends Book  
{  
 String author;  
 Novel(String author)  
 {  
   
 System.*out*.println("Author "+author);  
 }  
  
}  
 public class TestInheritance {  
 public static void main(String args[]) {  
  
 Magazine m=new Magazine("Fiction");  
 System.*out*.println("------------------------------------");  
  
 Novel n=new Novel("Nayantara Sahgal");  
  
  
  
 }  
 }



**Assignment 5:**

Create an abstract class Instrument which is having the abstract function play.

Create three more sub classes from Instrument which is Piano, Flute, Guitar.

Override the play method inside all three classes printing a message

“Piano is playing tan tan tan tan ” for Piano class

“Flute is playing toot toot toot toot” for Flute class

“Guitar is playing tin tin tin ” for Guitar class

You must not allow the user to declare an object of Instrument class.

Create an array of 10 Instruments.

Assign different type of instrument to Instrument reference.

Check for the polymorphic behavior of play method.

Use the instanceof operator to print that which object stored at which index of instrument array.

abstract class instrument  
{  
 abstract public void play();  
}  
  
class piano extends instrument  
{  
 public void play()  
 { System.*out*.println("Piano is playing tan tan tan tan");  
 }  
}  
  
class flute extends instrument  
{  
 public void play()  
 { System.*out*.println("Flute is playing toot toot toot toot");  
 }  
}  
  
class guitar extends instrument  
{  
 public void play()  
 {  
 System.*out*.println("Guitar is playing tin tin tin tin");  
 }  
}  
public class inst  
{ public static void main(String [] args)  
 { instrument ins[]=new instrument[10];  
 for(int i=0;i<10;i++)  
 { if(i==1 || i==5 || i==9)  
 ins[i]=new piano();  
 else if(i==3 || i==4 || i==7)  
 ins[i]=new flute();  
 else ins[i]=new guitar();  
 ins[i].play();  
 if(ins[i] instanceof piano)  
 System.*out*.println("InstanceOf Piano");  
 else if(ins[i] instanceof flute)  
 System.*out*.println("InstanceOf Flute");  
 else   
 System.*out*.println("InstanceOf Guitar");   
 System.*out*.println();  
 }  
 }  
}

Output

Guitar is playing tin tin tin tin

InstanceOf Guitar

Piano is playing tan tan tan tan

InstanceOf Piano

Guitar is playing tin tin tin tin

InstanceOf Guitar

Flute is playing toot toot toot toot

InstanceOf Flute

Flute is playing toot toot toot toot

InstanceOf Flute

Piano is playing tan tan tan tan

InstanceOf Piano

Guitar is playing tin tin tin tin

InstanceOf Guitar

Flute is playing toot toot toot toot

InstanceOf Flute

Guitar is playing tin tin tin tin

InstanceOf Guitar

Piano is playing tan tan tan tan

InstanceOf Piano

Process finished with exit code 0

**Assignment 6:**

Write an interface called Playable, with a method

void play();

Let this interface be placed in a package called music.

Write a class called Veena which implements Playable interface. Let this class be placed in a package music.string

Write a class called Saxophone which implements Playable interface. Let this class be placed in a package music.wind Sensitivity: Internal & Restricted

Write another class Test in a package called live. Then,

a. Create an instance of Veena and call play() method

b. Create an instance of Saxophone and call play() method

c. Place the above instances in a variable of type Playable and then call play()

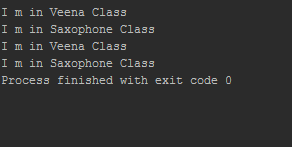
package live;  
  
import music.string.Veena;  
import music.wind.Saxophone;  
import music.Playable;  
  
public class Test {  
 public static void main(String args[])  
 {  
 Veena v=new Veena();  
 v.play();  
 Saxophone s=new Saxophone();  
 s.play();  
  
 Playable playable\_veena, playable\_s;  
 playable\_veena = new Veena();  
 playable\_s = new Saxophone();  
  
 playable\_veena.play();  
 playable\_s.play();  
 }  
}

package music;  
  
public interface Playable {  
 void play();  
 }

package music.string;  
import music.Playable;  
  
public class Veena implements Playable  
{  
  
 public void play() {  
  
 System.*out*.print("\nI m in Veena Class");  
 }  
}

package music.wind;  
  
import music.Playable;  
  
public class Saxophone implements Playable {  
  
 public void play() {  
 System.*out*.print("\nI m in Saxophone Class");  
 }  
}

Output:



**Topic 3: Exceptions, String Concepts**

**Assignment 1:**

Write a program to accept name and age of a person from the command prompt(passed as arguments when you execute the class) and ensure that the age entered is >=18 and < 60. Display proper error messages. The program must exit gracefully after displaying the error message in case the arguments passed are not proper. (Hint : Create a user defined exception class for handling errors.)

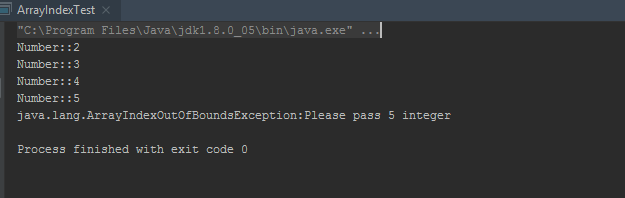
class InvalidAgeException extends Exception{  
 @Override  
 public String toString() {  
 return ("Age must be >=18 and <60");  
 }  
}  
  
public class Age\_exception {  
 public static void main(String[] args) {  
 String name = args[0];  
 int age = Integer.*parseInt*(args[1]);  
 if(age>=18 && age<60)  
 System.*out*.println(name+" has proper Age");  
 else  
 try {  
 throw new InvalidAgeException();  
 }  
 catch(InvalidAgeException ex) {  
 ex.printStackTrace();  
 }  
 }

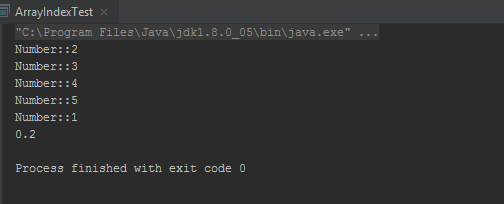
}

**Assignment 3:**

Write a program to accept 5 integers passed as arguments while executing the class. Find the average of these 5 nos. Use ArrayIndexOutofBounds exception to handle situation where the user might have entered less than 5 integers.

public class ArrayIndexTest  
{  
 public static void main(String args[])  
{  
 int num;  
 int a[]=new int[args.length];  
 double avg=0;  
  
 for(int i=0;i<args.length;i++)  
 {  
 num=Integer.*parseInt*(args[i]);  
 a[i]=num;  
 avg=num;  
 System.*out*.println("Number::"+num);  
  
  
 }  
 try {  
 if(a.length!=5)  
 {throw new ArrayIndexOutOfBoundsException();}  
 else  
 {  
 avg=avg/a.length;  
 System.*out*.println(avg);  
  
 }  
  
 }  
 catch(ArrayIndexOutOfBoundsException e)  
 {  
 System.*out*.println(e+":Please pass 5 integer");  
 }  
}  
}





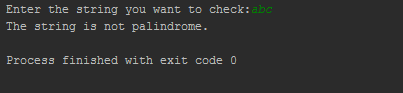
**Assignment 4:**

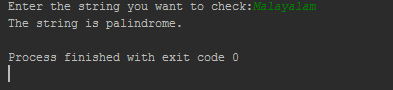
Write a program to check whether the given string is a palindrome or not.

[Hint :You have to extract each character from the beginning and end of the String and compare it with each other. String x=”Malayalam”; char c= x.charAt(i) where i is the index]

import java.util.Scanner;  
  
public class Palindrome  
{  
 public static void main(String args[])  
 {  
 String a, b = "";  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.print("Enter the string you want to check:");  
 a = s.nextLine();  
 int n = a.length();  
 for(int i = n - 1; i >= 0; i--)  
 {  
 b = b + a.charAt(i);  
 }  
 if(a.equalsIgnoreCase(b))  
 {  
 System.*out*.println("The string is palindrome.");  
 }  
 else  
 {  
 System.*out*.println("The string is not palindrome.");  
 }  
 }  
}

Output:



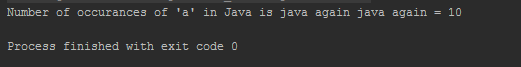


**Assignment 5:**

Write a program to check the no.of occurrences of a given character within the given string without using any loop. [Hint: String str=”How was your day today”; char c=’a’; no.of occurrences of a is=3]

class Occurence  
{  
 public static void main(String[] args)  
 {  
 String str = "Java is java again java again";  
 int count = str.length() - str.replace("a", "").length();  
 System.*out*.println("Number of occurances of 'a' in "+str+" = "+count);  
 }  
}

Output:



**Topic 4: Threads, Collection Framework, Garbage Collection**

**Assignment 1:**

Write a Java Program, where one thread prints a number ( Generate a random number using Math.random) and another thread prints the factorial of that given number. Both the outputs should alternate each other.

Eg: Number : 2

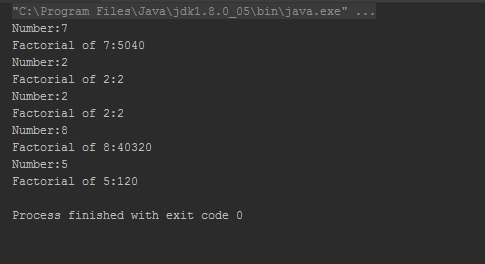
Factorial of 2 : 2

Number : 5

Factorial of 5 : 120

The program can quit after executing 5 times.

public class Multithread {  
 public static void main(String args[]) {  
  
 for(int n=1;n<6;n++)  
 {  
 int random=(int)(Math.*random*()\*10);  
 System.*out*.println("Number:"+random);  
  
 int fact=1;  
 if(random==0)  
 {  
 System.*out*.println("Factorial of :"+random+ ":1 ");  
 }  
 else {  
 for (int i = 1; i <= random; i++)  
 {  
 fact=fact\*i;  
 }  
 System.*out*.println("Factorial of "+random+":"+fact);  
 }  
 }  
 }  
  
 }



**Assignment 2:**

Write a Java Program which will print the current time on the console every 2 seconds. After doing this activity for 20 seconds the program quits.

import java.util.Date;  
import java.util.Timer;  
import java.util.TimerTask;  
  
public class MyThread extends TimerTask {  
  
 @Override  
 public void run() {  
 System.*out*.println("Timer task started at:"+new Date());  
 completeTask();  
 //System.out.println("Timer task finished at:"+new Date());  
 }  
  
 private void completeTask() {  
 try {  
 //assuming it takes 20 secs to complete the task  
 Thread.*sleep*(2000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
  
 public static void main(String args[]){  
 TimerTask timerTask = new MyThread();  
 //running timer task as daemon thread  
 Timer timer = new Timer(true);  
 timer.scheduleAtFixedRate(timerTask, 0, 2\*100);  
 System.*out*.println("TimerTask started");  
 //cancel after sometime  
 try {  
 Thread.*sleep*(20000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
  
 timer.cancel();  
 System.*out*.println("TimerTask cancelled");  
 try {  
 Thread.*sleep*(30000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
  
}

Output:

TimerTask started

Timer task started at:Fri Feb 21 21:04:45 IST 2020

Timer task started at:Fri Feb 21 21:04:47 IST 2020

Timer task started at:Fri Feb 21 21:04:49 IST 2020

Timer task started at:Fri Feb 21 21:04:51 IST 2020

Timer task started at:Fri Feb 21 21:04:53 IST 2020

Timer task started at:Fri Feb 21 21:04:55 IST 2020

Timer task started at:Fri Feb 21 21:04:57 IST 2020

Timer task started at:Fri Feb 21 21:04:59 IST 2020

Timer task started at:Fri Feb 21 21:05:01 IST 2020

Timer task started at:Fri Feb 21 21:05:03 IST 2020

TimerTask cancelled

Process finished with exit code 0

**Assignment 3:**

Create an Employee class with the related attributes and behaviours. Create one more class EmployeeDB which has the following methods.

a. boolean addEmployee(Employee e)

b. boolean deleteEmployee(int eCode)

c. String showPaySlip(int eCode)

d. Employee[] listAll()

Use an ArrayList which will be used to store the emplyees and use enumeration/iterator to process the employees.

public class Employee {  
 private int EmpId;  
 private String EmpName;  
 private String EmpEmail;  
 private char EmpGender;  
 private float EmpSalary;  
  
 public Employee() {}  
  
 public Employee(int empId, String empName, String empEmail, char empGender, float empSalary) {  
 super();  
 EmpId = empId;  
 EmpName = empName;  
 EmpEmail = empEmail;  
 EmpGender = empGender;  
 EmpSalary = empSalary;  
 }  
  
 public String GetEmployeeDetails() {  
 return "Employee [EmpId=" + EmpId + ", EmpName=" + EmpName + ", EmpEmail=" + EmpEmail  
 + ", EmpGender=" + EmpGender + ", EmpSalary=" + EmpSalary + "]";  
 }  
  
 public int getEmpId() {  
 return EmpId;  
 }  
  
 public void setEmpId(int empId) {  
 EmpId = empId;  
 }  
  
 public String getEmpName() {  
 return EmpName;  
 }  
  
 public void setEmpName(String empName) {  
 EmpName = empName;  
 }  
  
 public String getEmpEmail() {  
 return EmpEmail;  
 }  
  
 public void setEmpEmail(String empEmail) {  
 EmpEmail = empEmail;  
 }  
  
 public char getEmpGender() {  
 return EmpGender;  
 }  
  
 public void setEmpGender(char empGender) {  
 EmpGender = empGender;  
 }  
  
 public float getEmpSalary() {  
 return EmpSalary;  
 }  
  
 public void setEmpSalary(float empSalary) {  
 EmpSalary = empSalary;  
 }  
  
  
}

import java.util.Iterator;  
import java.util.List;  
import java.util.ArrayList;  
  
public class EmployeeDB {  
 List<Employee> employeeDb = new ArrayList<Employee>();  
  
  
 public boolean addEmployee(Employee e) {  
 return employeeDb.add(e);  
 }  
  
 public boolean deleteEmployee(int empId) {  
 boolean isRemoved = false;  
  
 Iterator<Employee> it = employeeDb.iterator();  
  
 while (it.hasNext()) {  
 Employee emp = it.next();  
 if (emp.getEmpId() == empId) {  
 isRemoved = true;  
 it.remove();  
 }  
 }  
  
 return isRemoved;  
 }  
  
 public String showPaySlip(int empId) {  
 String paySlip = "Invalid employee id";  
  
 for (Employee e : employeeDb) {  
 if (e.getEmpId() == empId) {  
 paySlip = "Pay slip for employee id " + empId + " is " +  
 e.getEmpSalary();  
 }  
 }  
  
 return paySlip;  
 }  
  
 public Employee[] listAll() {  
 Employee[] empArray = new Employee[employeeDb.size()];  
 for (int i = 0; i < employeeDb.size(); i++)  
 empArray[i] = employeeDb.get(i);  
 return empArray;  
 }  
  
}

public class MainTest {  
  
 public static void main(String[] args) {  
 EmployeeDB empDb = new EmployeeDB();  
  
 Employee emp1 = new Employee(101, "Bob", "bob@w3epic.com", 'M', 25000);  
 Employee emp2 = new Employee(102, "Alice", "alice@w3epic.com", 'F', 30000);  
 Employee emp3 = new Employee(103, "John", "john@w3epic.com", 'M', 20000);  
 Employee emp4 = new Employee(104, "Ram", "ram@w3epic.com", 'M', 50000);  
  
 empDb.addEmployee(emp1);  
 empDb.addEmployee(emp2);  
 empDb.addEmployee(emp3);  
 empDb.addEmployee(emp4);  
  
 for (Employee emp : empDb.listAll())  
 System.*out*.println(emp.GetEmployeeDetails());  
  
 System.*out*.println();  
 empDb.deleteEmployee(102);  
  
 for (Employee emp : empDb.listAll())  
 System.*out*.println(emp.GetEmployeeDetails());  
  
 System.*out*.println();  
  
 System.*out*.println(empDb.showPaySlip(103));  
 }  
  
}

Output :

Employee [EmpId=101, EmpName=Bob, EmpEmail=bob@w3epic.com, EmpGender=M, EmpSalary=25000.0]

Employee [EmpId=102, EmpName=Alice, EmpEmail=alice@w3epic.com, EmpGender=F, EmpSalary=30000.0]

Employee [EmpId=103, EmpName=John, EmpEmail=john@w3epic.com, EmpGender=M, EmpSalary=20000.0]

Employee [EmpId=104, EmpName=Ram, EmpEmail=ram@w3epic.com, EmpGender=M, EmpSalary=50000.0]

Employee [EmpId=101, EmpName=Bob, EmpEmail=bob@w3epic.com, EmpGender=M, EmpSalary=25000.0]

Employee [EmpId=103, EmpName=John, EmpEmail=john@w3epic.com, EmpGender=M, EmpSalary=20000.0]

Employee [EmpId=104, EmpName=Ram, EmpEmail=ram@w3epic.com, EmpGender=M, EmpSalary=50000.0]

Pay slip for employee id 103 is 20000.0

Process finished with exit code 0

**Assignment 5:**

Write a program to store a group of employee names into a HashSet, retrieve the elements one by one using an Iterator.

import java.util.HashSet;  
import java.util.Iterator;  
  
public class HashMap\_Assignment {  
  
 public static void main(String[] args) {  
 HashSet<String> set = new HashSet();  
  
 set.add("Bob");  
 set.add("Alice");  
 set.add("John");  
 set.add("Richard");  
  
 Iterator<String> it = set.iterator();  
 while (it.hasNext())  
 System.*out*.println(it.next());  
  
 }  
  
}

Output :

Bob

Alice

John

Richard

Process finished with exit code 0

**Assignment 4:**

Write a program creates a HashMap to store name and phone number (Telephone book). When name is give, we can get back the corresponding phone number.

import java.util.HashMap;  
import java.util.Scanner;  
  
public class Hash\_Map {  
 public static void main(String[] args) {  
 HashMap< String, String> TelephoneBook = new HashMap < String, String>();  
  
 TelephoneBook.put("SARA", "12345678");  
 TelephoneBook.put("TARA", "6787348730");  
 TelephoneBook.put("LARA", "678645730");  
 TelephoneBook.put("MEG", "670967548730");  
 TelephoneBook.put("CIA", "8765748730");  
  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Enter Name=");  
 String name=sc.next().toUpperCase();  
  
 if(TelephoneBook.containsKey(name)) {  
 System.*out*.println(name +"= "+TelephoneBook.get(name));  
 }  
 }  
}

Output

Enter Name=Meg

MEG= 670967548730

Process finished with exit code 0

**Topic 5: Command Line Args, System Properties, Packaging**

**Assignment 1:**

Create a package called test package;

Define a class called foundation inside the test package;

Inside the class, you need to define 4 integer variables;

Var1 as private;

Var2 as default;

Var3 as protected;

Var4 as public;

Import this class and packages in another class.

Try to access all 4 variables of the foundation class and see what variables are accessible and what are not accessible.

package testpackage;  
  
public class Foundation {  
 private int Var1;  
 int Var2;  
 protected int Var3;  
 public int Var4;  
}

import testpackage.\*;  
  
public class Modifiers\_access {  
 public static void main(String[] args) {  
 Foundation foundation = new Foundation();  
  
 foundation.Var4 = 5;  
  
 System.*out*.println("Only Variable 4 is accessible::"+foundation.Var4);  
 }  
}

output:

Only Variable 4 is accessible::5

Process finished with exit code 0

**Assignment 2:**

Write a Program to accept two Strings Wipro Bangalore as command line arguments and print the output “Wipro Technologies Bangalore” If the command line is “ABC Mumbai”, then it should print “ABC Technologies Mumbai” .

public class Command\_line {  
 public static void main(String args[])  
 {  
 String x = args[0];  
 String y = args[1];  
  
 System.*out*.println(x+" Technologies "+y.toString());  
  
 }  
}

Output :

ABC Technologies Mumbai

Process finished with exit code 0

Wipro Technologies Bangalore

Process finished with exit code 0