**Automation Assignment**

**Module: Ad. Selenium – Java for Web Driver**

**1)W.A.J. P to Take three numbers from the user and print the greatest number.**

import java. util. Scanner;

public class Gratest\_number\_1 {

public static void main (String [] args) {

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

System.***out***.println("Enter First Number: ");

int num1 = sc. nextInt ();

System.***out***.println("Enter Second Number: ");

int num2 = sc. nextInt ();

System.***out***.println("Enter Third Number: ");

int num3 = sc. nextInt ();

if (num1 >= num2 && num1 >= num3) {

System.***out***.println("Greatest number is: " + num1);

}

else if (num2 >= num1 && num2 >= num3) {

System.***out***.println("Greatest number is: " + num2);

}

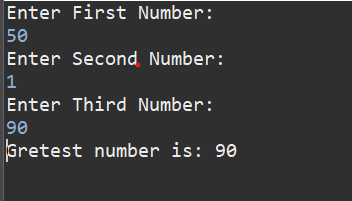
else {

System.***out***.println("Greatest number is: " + num3);

}

}

}



**2) W.A.J.P in Java to display the first 10 natural numbers using while loop.**

public class Ten\_Natural\_Numbers\_2 {

public static void main (String [] args) {

System.*out*.println("Natural number is: ");

int num =1;

while (num <= 10) {

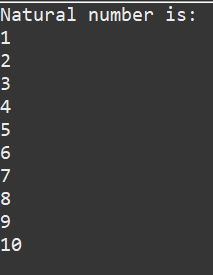
System.*out*.println(num);

num++;

}

}

}



**3)W.A.J. P to find factorial for Given Number.**

import java. util. Scanner;

public class Factorial\_3 {

public static void main (String [] args) {

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

System.*out*.println("Enter number: ");

int num = sc. nextInt ();

int fact = 1;

for (int i=1; i<=num; i++) {

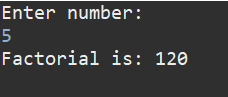
fact = fact \*i;

}

System.*out*.println("Factorial is: " + fact);

}

}



**4)W.A.J. P to check given number is Prime or not?**

import java. util. Scanner;

public class Prime\_4 {

public static void main (String [] args) {

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

System.*out*.println("Enter Number: ");

int num = sc. nextInt ();

int count = 0;

for (int i=1; i<=num; i++) {

if (num%i == 0) {

count++;

}

}

if (count == 2) {

System.*out*.println("Number is Prime");

}

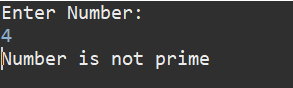
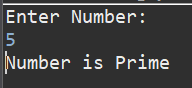
else {

System.*out*.println("Number is not prime");

}

}

}

**5)W.A.J. P to check given number is Armstrong or not?**

import java. util. Scanner;

public class Amstrong\_5 {

public static void main (String [] args) {

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

System.*out*.println("Enter Number: ");

int num = sc. nextInt ();

int sum =0;

int tmp = num;

while (num>0) {

int digit = num%10;

sum = sum + (digit \* digit \* digit);

num = num/10;

}

if (sum == tmp) {

System.*out*.println("Number is Amstrong");

}

else

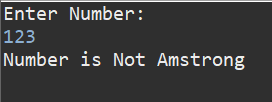
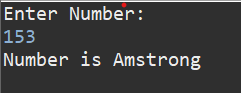
{

System.*out*.println("Number is Not Amstrong");

}

}

}

**6)W.A.J. P for create Fibonacci Series.**

import java. util. Scanner;

public class Fibonacci\_6 {

public static void main (String [] args) {

int a=0;

int b=1;

int c;

System.*out*.println("Fibonacci Series up to 10 times:");

System.*out*.print(a + " " + b + " ");

for (int i=2; i<=10; i++) {

c = a+b;

System.*out*.print(c + " ");

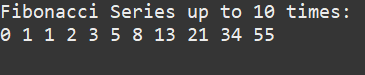
a=b;

b=c;

}

}

}



**7)W.A.J. P to Print pattern Given Below.**

**Pattern-1**

**1**

**12**

**123**

**1234**

**12345**

public class Pattern1\_7 {

public static void main (String [] args) {

for (int i=1; i<=5; i++) {

for (int j=1; j<=i; j++) {

System.*out*.print(j);

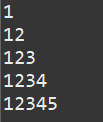
}

System.*out*.println();

}

}

}



**Pattern-2**

**1**

**22**

**333**

**4444**

**55555**

public class Pattern2\_7 {

public static void main (String [] args) {

for (int i=1; i<=5; i++) {

for (int j=1; j<=i; j++) {

System.*out*.print(i);

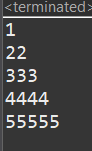
}

System.*out*.println();

}

}

}



**Pattern-3**

**1**

**01**

**101**

**01010**

**101010**

public class Pattern3\_7 {

public static void main (String [] args) {

for (int i = 1; i <= 5; i++) {

for (int j = 1; j <= i; j++) {

if ((i + j) % 2 == 0)

System.*out*.print("1");

else

System.*out*.print("0");

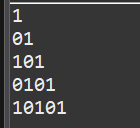
}

System.*out*.println();

}

}

}



**Pattern-4**

**1**

**2 2**

**3 3 3**

**4 4 4 4**

public class pattern4\_7 {

public static void main (String [] args) {

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

for (int j = 1; j <= 4 - i; j++) {

System.*out*.print(" ");

}

for (int j = 1; j <= i; j++) {

System.*out*.print(i + " ");

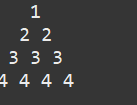
}

System.*out*.println();

}

}

}



**Pattern-5**

**\***

**\*\*\***

**\*\*\*\*\***

**\*\*\***

**\***

public class pattern5\_7 {

public static void main (String [] args) {

// Upper half

for (int i = 1; i <= 3; i++) {

for (int j = i; j < 3; j++) {

System.*out*.print(" ");

}

for (int j = 1; j <= (2 \* i - 1); j++) {

System.*out*.print("\*");

}

System.*out*.println();

}

for (int i = 3 - 1; i >= 1; i--) {

for (int j = 3; j > i; j--) {

System.*out*.print(" ");

}

for (int j = 1; j <= (2 \* i - 1); j++) {

System.*out*.print("\*");

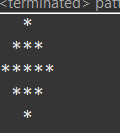
}

System.*out*.println();

}

}

}



**8)WAP to compute the sum of the first 100 prime numbers.**

public class first\_100\_primeNumbers\_8 {

public static void main (String [] args) {

int sum = 0;

System.*out*.println("1 to 100 Prime numbers: ");

for (int num = 2; num<=100; num++) {

int count =0;

for (int i=1; i<=num; i++) {

if (num % i == 0) {

count++;

}

}

if (count == 2) {

System.*out*.print(num + " ");

sum = sum + num;

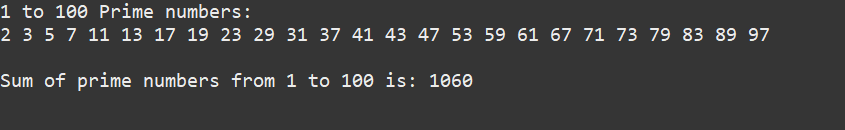
}

}

System.*out*.println("\n\nSum of prime numbers from 1 to 100 is: " + sum);

}

}



**9) WAP to sum values of an array.**

public class Sum\_value\_array\_9 {

public static void main (String [] args) {

int a [] = {10,20,30,40,50};

int sum =0;

for (int i=0; i<a. length; i++) {

sum = sum + a[i];

}

System.*out*.println("Sum of array is: " + sum);

}

}



**10) WAP to calculate the average value of array elements.**

public class Calculate\_averageArray\_10 {

public static void main (String [] args) {

int a [] = {10,20,30,40,50};

int sum =0;

for (int i=0; i<a. length; i++) {

sum = sum + a[i];

}

int avg = sum / a. length;

System.*out*.println("Average: " + avg);

}

}



**11)WAP to find the index of an array element.**

public class find\_index\_array\_11 {

public static void main (String [] args) {

int a [] = {10,20,30,40,50};

int index = 10;

for (int i=0; i<a. length; i++) {

if (a[i] == index) {

System.*out*.println(index + " is at index " + i);

}

}

}

}



**12)WAP to find the maximum and minimum value of an array.**

public class max\_min\_12 {

public static void main (String [] args) {

int a [] = {10,5,20,50,1};

int max = a [0];

int min = a [0];

for (int i=0; i<a. length; i++) {

if (a[i] > max) {

max = a[i];

}

if(a[i] < min) {

min = a[i];

}

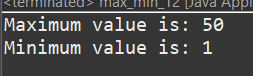
}

System.*out*.println("Maximum value is: " + max);

System.*out*.println("Minimum value is: " + min);

}

}



**13)WAP to Compare Two String.**

public class Compare\_two\_string\_13 {

public static void main (String [] args) {

String st1 = "Hello";

String st2 = "hello";

if (st1.equalsIgnoreCase(st2)) {

System.*out*.println("String are equal");

}

else

{

System.*out*.println("String are not equal");

}

}

}



**14)WAP to concatenate a given string to the end of another string.**

public class Concatintion\_two\_string\_14 {

public static void main (String [] args) {

String st1 = "Snehal ";

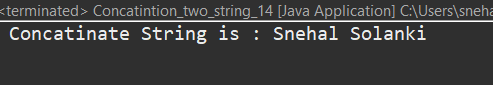
String st2 = "Solanki";

String res = st1 + st2;

System.*out*.println(" Concatenate String is : " + res);

}

}



**15)WAP to demonstrate try catch block.**

public class Try\_Catch\_15 {

public static void main (String [] args) {

try {

int a = 10;

int b= 0;

int res = a/b;

System.*out*.println("Result is: " +res);

} catch (ArithmeticException e) {

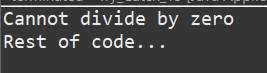
System.*out*.println("Cannot divide by zero");

}

System.*out*.println("Rest of code...");

}

}



**16) WAP to demonstrate multiple catch blocks**

public class Multiple\_catch\_16 {

public static void main (String [] args) {

int arr [] = new int [5];

try {

arr [5] = 10;

System.*out*.println("Hello ");

} catch (ArithmeticException e) {

System.*out*.println("Error " + e);

}

catch (Exception e) {

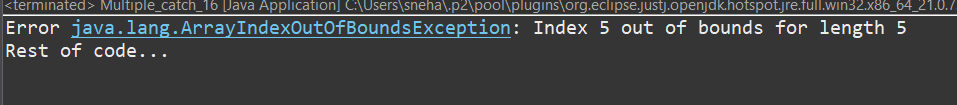
System.*out*.println("Error " + e);

}

System.*out*.println("Rest of code...");

}

}



**17) WAP to create one thread by implementing Runnable interface in Class.**

class Thread1 implements Runnable {

public void run () {

System.*out*.println("Runnable Thread is Running......");

}

public static void main (String [] args) {

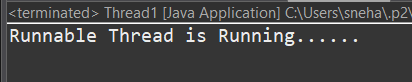
Thread1 t1 = new Thread1();

Thread m1 = new Thread(t1);

m1. start ();

}

}



**18) WAP to create one thread by extending Thread class in another Class.**

public class Thread2 extends Thread {

public void run () {

System.*out*.println("Extending Thread is Running....");

}

public static void main (String [] args) {

Thread2 t2 = new Thread2();

t2. start ();

}

}



**19) WAP to iterate through all elements in an array list.**

import java.util.\*;

public class Array\_listD1 {

public static void main (String [] args) {

ArrayList<String> st = new ArrayList<String> ();

st.add("Apple");

st.add("Banana");

st.add("Mango");

st.add("Orange");

//Using while

Iterator itr = st. iterator ();

while (itr. hasNext ()) {

System.*out*.println(itr. next ());

}

//Using for each loop

System.*out*.println("--------------");

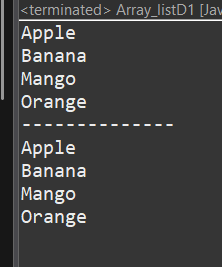
for (String s:st) {

System.*out*.println(s);

}

}

}



**20) WAP to update specific array element by given element.**

import java.util.\*;

public class Update\_Array {

public static void main (String [] args) {

ArrayList<String> st = new ArrayList<String>();

st.add("Red");

st.add("Yellow");

st.add("Pink");

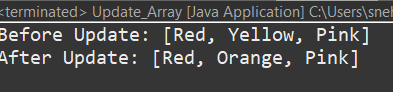
System.*out*.println("Before Update: " + st);

st.set (1, "Orange");

System.*out*.println("After Update: " + st);

}

}



**21) WAP to remove the third element from a array list.**

import java.util.\*;

public class List\_Remove {

public static void main (String [] args) {

ArrayList<String> st=new ArrayList<String> ();

st.add("Cow");

st.add("Buffalo");

st.add("Dog");

st.add("Sparrow");

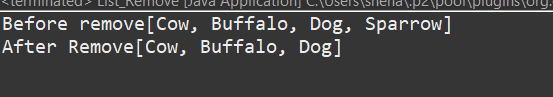
System.*out*.println("Before remove" + st);

st. remove (3);

System.*out*.println("After Remove" + st);

}

}



**22) WAP to Copy one array into another**

public class copy\_array {

public static void main (String [] args) {

int a [] = {1,2,3,4,5};

int b [] = new int [5];

for (int i=0; i<5; i++) {

b[i] = a[i];

}

System.***out***.println("Original Array: ");

for (int i=0; i<5; i++) {

System.***out***.println(a[i]);

}

System.***out***.println("Copied Array: ");

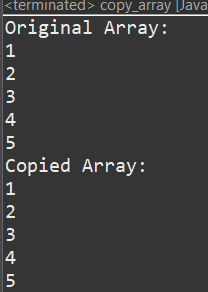
for (int i=0; i<5; i++) {

System.***out***.println(b[i]);

}

}

}



**23) WAP to reverse an array of integer values.**

public class Reverese\_array {

public static void main (String [] args) {

int a [] = {10,20,30,40,50};

int b [] = new int [5];

System.***out***.println("Original Array is: ");

for (int i=0; i<=4; i++) {

System.***out***.println(a[i]);

}

System.***out***.println("Reverse Array is: ");

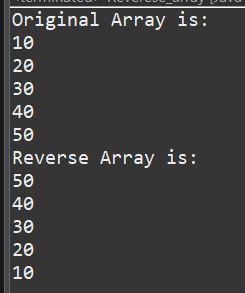
for (int i = 4; i>=0; i--) {

System.***out***.println(a[i] + " ");

}

}

}



**24)WAP to find the second largest element in an array.**

public class Second\_largest\_array\_find {

public static void main (String [] args) {

int [] arr = {45,22,89,75,12};

int largest = arr [0];

int second = arr [0];

for (int i=1; i<arr. length; i++) {

if(arr[i] > largest) {

largest = arr[i];

}

}

for (int i=0; i<arr. length; i++) {

if(arr[i] > second && arr[i] < largest) {

second = arr[i];

}

}

System.*out*.println("Second Largest Number is: "+ second);

}

}



**25) W.A.J.P. Create an abstract class 'Parent' with a method 'message'. It has two subclasses each having a method with the same name 'message' that prints "This is first subclass" and "This is second subclass" respectively. Call the methods 'message' by creating an object for each subclass.**

abstract class parent {

abstract void message ();

}

class sub1 extends parent {

void message () {

System.*out*.println("I am sub1 class");

}

}

class sub2 extends parent {

void message () {

System.*out*.println("I am sub2 class");

}

}

public class abstraction {

public static void main (String [] args) {

sub1 a = new sub1();

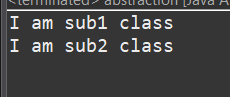
a. message ();

sub2 b = new sub2();

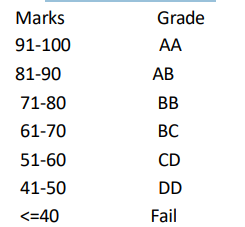
b. message ();

}

}



**26) W.A.J.P. which will ask the user to enter his/her marks (out of 100). Define a method that will display grades according to the marks entered as below:**

****

import java. util. Scanner;

public class grades {

public void disp (int marks) {

if (marks >= 91 && marks <= 100) {

System.*out*.println("Grade: AA");

}

else if (marks >= 81 && marks <= 90) {

System.*out*.println("Grade: AB");

}

else if (marks >= 71 && marks <= 80) {

System.*out*.println("Grade: BB");

}

else if (marks >= 61 && marks <= 70) {

System.*out*.println("Grade: BC");

}

else if (marks >= 51 && marks <= 60) {

System.*out*.println("Grade: CD");

}

else if (marks >= 41 && marks <= 50) {

System.*out*.println("Grade: DD");

}

else if (marks >= 0 && marks <= 40) {

System.*out*.println("Grade: Fail");

}

else {

System.*out*.println("Invalid Marks");

}

}

public static void main (String [] args) {

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

System.*out*.println("Enter your marks (out of 100): ");

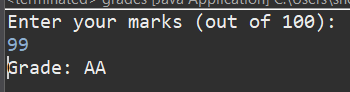
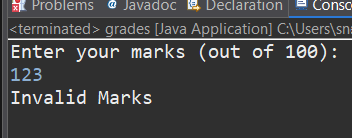
int marks = sc. nextInt ();

grades g = new grades ();

g.disp(marks);

}

}

**27)** **W.A.J.P. to create a custom exception if Customer withdraw amount which is greater than account balance then program will show custom exception otherwise amount will deduct from account balance. Account balance is:2000 Enter withdraw amount:2500 Sorry, insufficient balance, you need more 500 Rs. To perform this transaction.**

class MyException extends Exception {

MyException (String message) {

super(message);

}

public static void main (String [] args) {

int acBalance = 2000;

int withdrawAmount = 500;

try {

if (withdrawAmount > acBalance) {

int store = withdrawAmount - acBalance;

throw new MyException ("Sorry, insufficient balance you need more " + store + "Rs. to perform this transaction");

}

else {

acBalance -= withdrawAmount;

System.*out*.println("Withdraw successful. Remaining balance: " +acBalance);

}

} catch (MyException e) {

System.*out*.println(e. getMessage ());

}

}

}

