**MODULE-6 (Core Java)**

**1.Take three numbers from the user and print the greatest number:**

import java.util.Scanner;

public class GreatestNumber {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter three numbers:");

int num1 = scanner.nextInt();

int num2 = scanner.nextInt();

int num3 = scanner.nextInt();

int greatest = num1;

if (num2 > greatest) {

greatest = num2;

}

if (num3 > greatest) {

greatest = num3;

}

System.out.println("The greatest number is: " + greatest);

}

}

**2. Display the first 10 natural numbers using a while loop:**

public class NaturalNumbers {

public static void main(String[] args) {

int i = 1;

while (i <= 10) {

System.out.println(i);

i++;

}

}

}

**3. Find factorial for a given number:**

import java.util.Scanner;

public class Factorial {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int num = scanner.nextInt();

int factorial = 1;

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

factorial \*= i;

}

System.out.println("The factorial of " + num + " is: " + factorial);

}

}

**4. Check if a given number is prime or not:**

import java.util.Scanner;

public class PrimeCheck {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int num = scanner.nextInt();

boolean isPrime = true;

for (int i = 2; i <= num / 2; i++) {

if (num % i == 0) {

isPrime = false;

break;

}

}

if (isPrime) {

System.out.println(num + " is a prime number.");

} else {

System.out.println(num + " is not a prime number.");

}

}

}

**5. Check if a given number is Armstrong or not:**

import java.util.Scanner;

public class ArmstrongCheck {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int num = scanner.nextInt();

int originalNum = num;

int result = 0;

while (originalNum != 0) {

int remainder = originalNum % 10;

result += Math.pow(remainder, 3);

originalNum /= 10;

}

if (result == num) {

System.out.println(num + " is an Armstrong number.");

} else {

System.out.println(num + " is not an Armstrong number.");

}

}

}

**6. Create Fibonacci Series:**

import java.util.Scanner;

public class FibonacciSeries {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the number of terms:");

int n = scanner.nextInt();

int a = 0, b = 1;

System.out.print("Fibonacci Series: " + a + " " + b);

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

int next = a + b;

System.out.print(" " + next);

a = b;

b = next;

}

}

}

**7. Print patterns:**

public class Patterns {

public static void main(String[] args) {

// Pattern 1

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

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

System.out.print(j);

}

System.out.println();

}

// Pattern 2

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

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

System.out.print(j);

}

System.out.println();

}

// Pattern 3

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

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

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

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

}

System.out.println();

}

// Pattern 5

int n = 3;

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

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

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

}

System.out.println();

}

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

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

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

}

System.out.println();

}

}

}

**8.Compute the sum of the first 100 prime numbers:**

public class SumOfPrimes {

public static void main(String[] args) {

int count = 0, num = 2, sum = 0;

while (count < 100) {

boolean isPrime = true;

for (int i = 2; i <= Math.sqrt(num); i++) {

if (num % i == 0) {

isPrime = false;

break;

}

}

if (isPrime) {

sum += num;

count++;

}

num++;

}

System.out.println("The sum of the first 100 prime numbers is: " + sum);

}

}

**9. Sum values of an array:**

public class SumArray {

public static void main(String[] args) {

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

int sum = 0;

for (int num : array) {

sum += num;

}

System.out.println("The sum of the array values is: " + sum);

}

}

**10. Calculate the average value of array elements:**

public class AverageArray {

public static void main(String[] args) {

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

int sum = 0;

for (int num : array) {

sum += num;

}

double average = (double) sum / array.length;

System.out.println("The average value of the array elements is: " + average);

}

}

**11. Find the index of an array element:**

import java.util.Scanner;

public class FindIndex {

public static void main(String[] args) {

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

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the element to find:");

int element = scanner.nextInt();

int index = -1;

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

if (array[i] == element) {

index = i;

break;

}

}

if (index != -1) {

System.out.println("Element found at index: " + index);

} else {

System.out.println("Element not found in the array.");

}

}

}

**12. Find the maximum and minimum value of an array:**

public class MaxMinArray {

public static void main(String[] args) {

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

int max = array[0];

int min = array[0];

for (int num : array) {

if (num > max) {

max = num;

}

if (num < min) {

min = num;

}

}

System.out.println("The maximum value in the array is: " + max);

System.out.println("The minimum value in the array is: " + min);

}

}

**13. Compare two strings:**

import java.util.Scanner;

public class CompareStrings {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the first string:");

String str1 = scanner.nextLine();

System.out.println("Enter the second string:");

String str2 = scanner.nextLine();

if (str1.equals(str2)) {

System.out.println("The strings are equal.");

} else {

System.out.println("The strings are not equal.");

}

}

}

**14. Concatenate a given string to the end of another string:**

import java.util.Scanner;

public class ConcatenateStrings {

public static void main(String[] args[\_{{{CITATION{{{\_1{](https://github.com/Madhukar97/BasicCorePrograms/tree/4fafe79bd9865d8ad6625f98ba4b42fdfdeb20e2/src%2Fmain%2Fjava%2Fcom%2Fbridgelabz%2FLargestOfThree.java)[\_{{{CITATION{{{\_2{](https://github.com/learnwithlakshman/b8-core-java/tree/c1148fadee229bdba9c2307c9fc719f7ca68a0bf/src%2Fcom%2Fcareerit%2Fcj%2Fday4%2FCheckPrimeOrNot.java)[\_{{{CITATION{{{\_3{](https://github.com/KirtiRanpise-QA/KirtiJavaPrograms/tree/648775e8f52dcaa8baec0a368a512e76906bc58f/JavaProgramingQuesWBL50%2Fsrc%2Fcom%2FLoops%2FNumberIsPrimeOrNot.java)[\_{{{CITATION{{{\_4{](https://github.com/tekZoli/Batch8/tree/786f74b8962f592e53a1bbce5776f1b157428d12/src%2FHomeWorkPractice%2FPrimeN.java)[\_{{{CITATION{{{\_5{](https://github.com/ImranNVohraTr/HitJAVA\_FullstackTraining/tree/687d48a55b3751516ee5950ff1579e0ec2e60407/CheckPrime.java)[\_{{{CITATION{{{\_6{](https://github.com/DilipAkbari/w8homework/tree/b80932f56530cee7706d15271042ae9cbc3af17c/homework29022020%2FQ10ArmStrongNumber.java)[\_{{{CITATION{{{\_7{](https://github.com/mounikajajala/company/tree/5a2aaccf793a02f7c54da3ef86fb83331f90b917/fundamentals%2FPattern2.java)[\_{{{CITATION{{{\_8{]([https://github.com/ThamizhPrabhakaran/Java-Programs/tree/3ed0ef3b63642ec087fda1c4419cee6e9b5ff132/Pattern.java)[\_{{{CITATION{{{\_9{](https://github.com/rakeshkamble826/Hefshinejavaprogram/tree/98db1a69965388720e0527f9fa2f49ce8af6590a/javabasics%2A%2FPattern4.java)[\_{{{CITATION{{{\_10{](https://github.com/Abotabraham/Java-basic-projects/tree/8d03b5857722c3515126df65d5fb571366535c92/Loops%2FDisplayPattern2.java)[\_{{{CITATION{{{\_11{](https://github.com/HemaReddy89/Java\_Programs/tree/c658efa1f7a4a0c214199bb0f2f906629c1c6b37/Java\_Programs%2Fsrc%2FDiamond.java)[\_{{{CITATION{{{\_12{](https://github.com/wkk0407/Java/tree/d4df20123af6314fe8a372bcbe30859a69b849ee/src%2FPrintStars2522.java)[\_{{{CITATION{{{\_13{](https://github.com/pankajtripathi/Interview-Prep/tree/deb19e28f15a78292b24305a29dc1a29486b3cfe/src%2Fstrings%2FOneAway.java)[\_{{{CITATION{{{\_14{](https://github.com/viniciusjavs/java-programming/tree/d35c37313d57e6f4410757e8aa6c5c505479c428/mooc-java-programming-i%2Fpart01-Part01\_34.Same%2Fsrc%2Fmain%2Fjava%2FSame.java](https://github.com/ThamizhPrabhakaran/Java-Programs/tree/3ed0ef3b63642ec087fda1c4419cee6e9b5ff132/Pattern.java)%5b_%7b%7b%7bCITATION%7b%7b%7b_9%7b%5d(https://github.com/rakeshkamble826/Hefshinejavaprogram/tree/98db1a69965388720e0527f9fa2f49ce8af6590a/javabasics%2A%2FPattern4.java)%5b_%7b%7b%7bCITATION%7b%7b%7b_10%7b%5d(https://github.com/Abotabraham/Java-basic-projects/tree/8d03b5857722c3515126df65d5fb571366535c92/Loops%2FDisplayPattern2.java)%5b_%7b%7b%7bCITATION%7b%7b%7b_11%7b%5d(https://github.com/HemaReddy89/Java_Programs/tree/c658efa1f7a4a0c214199bb0f2f906629c1c6b37/Java_Programs%2Fsrc%2FDiamond.java)%5b_%7b%7b%7bCITATION%7b%7b%7b_12%7b%5d(https://github.com/wkk0407/Java/tree/d4df20123af6314fe8a372bcbe30859a69b849ee/src%2FPrintStars2522.java)%5b_%7b%7b%7bCITATION%7b%7b%7b_13%7b%5d(https://github.com/pankajtripathi/Interview-Prep/tree/deb19e28f15a78292b24305a29dc1a29486b3cfe/src%2Fstrings%2FOneAway.java)%5b_%7b%7b%7bCITATION%7b%7b%7b_14%7b%5d(https://github.com/viniciusjavs/java-programming/tree/d35c37313d57e6f4410757e8aa6c5c505479c428/mooc-java-programming-i%2Fpart01-Part01_34.Same%2Fsrc%2Fmain%2Fjava%2FSame.java))

**15. Concatenate a given string to the end of another string:**

import java.util.Scanner;

public class ConcatenateStrings {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the first string:");

String str1 = scanner.nextLine();

System.out.println("Enter the second string:");

String str2 = scanner.nextLine();

String concatenatedString = str1 + str2;

System.out.println("The concatenated string is: " + concatenatedString);

}

}

**16.Demonstarte try-catch block:**

public class TryCatchDemo {

public static void main(String[] args) {

try {

int[] array = new int[5];

array[10] = 30; // This will throw ArrayIndexOutOfBoundsException

} catch (ArrayIndexOutOfBoundsException e) {

System.out.println("Array index is out of bounds!");

}

}

}

**17. Demonstrate multiple catch blocks:**

public class MultipleCatchDemo {

public static void main(String[] args) {

try {

int[] array = new int[5];

array[10] = 30; // This will throw ArrayIndexOutOfBoundsException

int result = 10 / 0; // This will throw ArithmeticException

} catch (ArrayIndexOutOfBoundsException e) {

System.out.println("Array index is out of bounds!");

} catch (ArithmeticException e) {

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

}

}

}

**18. Create one thread by implementing Runnable interface in a class:**

public class RunnableThread implements Runnable {

public void run() {

System.out.println("Thread is running...");

}

public static void main(String[] args) {

RunnableThread runnableThread = new RunnableThread();

Thread thread = new Thread(runnableThread);

thread.start();

}

}

**19. Create one thread by extending Thread class in another class:**

public class ThreadExample extends Thread {

public void run() {

System.out.println("Thread is running...");

}

public static void main(String[] args) {

ThreadExample threadExample = new ThreadExample();

threadExample.start();

}

}

**20. Iterate through all elements in an ArrayList:**

import java.util.ArrayList;

public class IterateArrayList {

public static void main(String[] args) {

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

list.add("Apple");

list.add("Banana");

list.add("Cherry");

for (String fruit : list) {

System.out.println(fruit);

}

}

}

**21. Update specific array element by given element:**

import java.util.ArrayList;

public class UpdateArrayList {

public static void main(String[] args) {

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

list.add("Apple");

list.add("Banana");

list.add("Cherry");

list.set(1, "Blueberry"); // Update element at index 1

System.out.println("Updated ArrayList: " + list);

}

}

**22. Remove the third element from an ArrayList:**

import java.util.ArrayList;

public class RemoveElement {

public static void main(String[] args) {

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

list.add("Apple");

list.add("Banana");

list.add("Cherry");

list.add("Date");

list.remove(2); // Remove element at index 2

System.out.println("ArrayList after removal: " + list);

}

}

**23. Copy one array into another:**

import java.util.Arrays;

public class CopyArray {

public static void main(String[] args) {

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

int[] copiedArray = Arrays.copyOf(originalArray, originalArray.length);

System.out.println("Original array: " + Arrays.toString(originalArray));

System.out.println("Copied array: " + Arrays.toString(copiedArray));

}

}

**24. Reverse an array of integer values:**

import java.util.Arrays;

public class ReverseArray {

public static void main(String[] args) {

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

int n = array.length;

for (int i = 0; i < n / 2; i++) {

int temp = array[i];

array[i] = array[n - i - 1];

array[n - i - 1] = temp;

}

System.out.println("Reversed array: " + Arrays.toString(array));

}

}

**25. Find the second largest element in an array:**

public class SecondLargest {

public static void main(String[] args) {

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

int firstLargest = Integer.MIN\_VALUE;

int secondLargest = Integer.MIN\_VALUE;

for (int num : array) {

if (num > firstLargest) {

secondLargest = firstLargest;

firstLargest = num;

} else if (num > secondLargest && num != firstLargest) {

secondLargest = num;

}

}

System.out.println("The second largest element is: " + secondLargest);

}

}

**26. 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 FirstSubclass extends Parent {

void message() {

System.out.println("This is first subclass");

}

}

class SecondSubclass extends Parent {

void message() {

System.out.println("This is second subclass");

}

}

public class AbstractClassDemo {

public static void main(String[] args) {

Parent obj1 = new FirstSubclass();

obj1.message();

Parent obj2 = new SecondSubclass();

obj2.message();

}

}

**27. Ask the user to enter his/her marks (out of 100) and display grades according to the marks entered:**

import java.util.Scanner;

public class GradeCalculator {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int marks = scanner.nextInt();

String grade;

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

grade = "AA";

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

grade = "AB";

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

grade = "BB";

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

grade = "BC";

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

grade = "CD";

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

grade = "DD";

} else {

grade = "Fail";

}

System.out.println("Your grade is: " + grade);

}

}

**28. Create a custom exception if a customer withdraws an amount greater than the account balance:**

class InsufficientBalanceException extends Exception {

public InsufficientBalanceException(String message) {

super(message);

}

}

public class BankAccount {

private int balance = 2000;

public void withdraw(int amount) throws InsufficientBalanceException {

if (amount > balance) {

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

} else {

balance -= amount;

System.out.println("Transaction successful! Remaining balance: " + balance);

}

}

public static void main(String[] args) {

BankAccount account = new BankAccount();

try {

account.withdraw(2500);

} catch (InsufficientBalanceException e) {

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

}

}

}