**Question 1**

public class StudentMarks {

public static void main(String[] args) {

String name = "John Doe";

String[] subjects = {"Math", "English", "Science", "History", "Art"};

int[] marks = {85, 90, 78, 92, 88};

System.out.println("Student name: " + name);

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

System.out.println(subjects[i] + ": " + marks[i] + " (" + getGrade(marks[i]) + ")");

}

}

public static String getGrade(int mark) {

if (mark >= 90) {

return "A";

} else if (mark >= 80) {

return "B";

} else if (mark >= 70) {

return "C";

} else if (mark >= 60) {

return "D";

} else {

return "F";

}

}

}

**Question 2**

public class BreakStatement {

public static void main (String [] args) {

// Declare an array of numbers

Int [] numbers = {10, 20, 30, 40, 50};

// Unlabelled break statement

// Loop through the numbers and break when the number is 30

for (int num : numbers) {

System.out.println(num);

if (num == 30) {

break; // This will exit the for loop

}

}

// Labelled break statement

// Create a label called outer

outer:

// Loop through the numbers from 1 to 10

for (int i = 1; i <= 10; i++) {

// Loop through the numbers from 11 to 20

for (int j = 11; j <= 20; j++) {

System.out.println(i + " x " + j + " = " + (i \* j));

if (i == 5 && j == 15) {

break outer; // This will exit both loops and go to the label

}

}

}

System.out.println("End of program");

}

}

**Question 3**

Branching statements are used to alter the flow of execution in a program based on some conditions. There are three types of branching statements in Java:

* **break**: This statement is used to exit a loop or a switch case block. It can also be used with a label to exit a nested loop or block.
* **continue**: This statement is used to skip the current iteration of a loop and move to the next one. It can also be used with a label to skip the current iteration of a nested loop.
* **return**: This statement is used to return a value from a method and terminate the method execution. It can also be used without a value to end a void method.

**Question 4**

public class MaximumNumber {

public static void main(String[] args) {

int num1 = 10; // Replace 10 with the first number

int num2 = 20; // Replace 20 with the second number

int max;

if (num1 > num2) {

max = num1;

} else {

max = num2;

}

System.out.println("The maximum number is " + max);

}

}

**Question 5**

import java.util.Scanner;

public class StudentsGrade {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the full name of the student:");

String name = sc.nextLine();

System.out.println("Enter the score of the student:");

int score = sc.nextInt();

sc.close();

String grade;

if (score >= 70) {

grade = "A";

} else if (score >= 60) {

grade = "B";

} else if (score >= 50) {

grade = "C";

} else if (score >= 40) {

grade = "D";

} else {

grade = "F";

}

System.out.println("Student name: " + name);

System.out.println("Student grade: " + grade);

}

}

**Question 6**

public class LoopIteration {

public static void main (String [] args) {

// Use a for loop to iterate from 12 to 28 with a step of 2

for (int i = 12; i <= 28; i += 2) {

// Print the current value of i

System.out.println(i);

}

}

}

**Question 7**

public class SumNum {

public static void main (String [] args) {

// Declare a variable to store the sum

int sum = 0;

// Use a for loop to iterate from 20 to 25

for (int i = 20; i <= 25; i++) {

// Add the current value of i to the sum

sum += i;

}

// Print the sum

System.out.println("The sum is " + sum);

}

}

**Question 8**

public class DoWhile Iteration {

public static void main(String[] args) {

// Declare a variable to store the current number

int num = 12;

// Use a do-while loop to iterate until num is greater than 50

do {

// Print the current value of num

System.out.println(num);

// Increment num by 2 to get the next even number

num += 2;

} while (num <= 50);

}

}

**Question 9**

public class SumNum {

public static void main(String[] args) {

// Declare a variable to store the sum

int sum = 0;

// Declare a variable to store the current number

int num = 20;

// Use a do-while loop to iterate until num is greater than 25

do {

// Add the current value of num to the sum

sum += num;

// Increment num by 1 to get the next number

num++;

} while (num <= 25);

// Print the sum

System.out.println("The sum is " + sum);

}

}

**Question 10**

import java.util.Scanner;

public class SumNum {

public static void main(String[] args) {

// Create a scanner object to read user input

Scanner sc = new Scanner(System.in);

// Declare a variable to store the sum

int sum = 0;

// Declare a variable to store the current number

int num = 0;

// Use a do-while loop to iterate until the user enters 0

do {

// Ask the user to enter a number

System.out.println("Enter a number (0 to stop):");

num = sc.nextInt();

// Add the current value of num to the sum

sum += num;

} while (num != 0);

// Close the scanner

sc.close();

// Print the sum

System.out.println("The sum is " + sum);

}

}

**Question 11**

import java.util.Arrays;

public class PrimeNumbers {

public static void main(String[] args) {

int limit = 100;

boolean[] isPrime = new boolean[limit + 1];

Arrays.fill(isPrime, true);

// The Sieve of Eratosthenes algorithm

for (int p = 2; p \* p <= limit; p++) {

if (isPrime[p]) {

for (int i = p \* p; i <= limit; i += p) {

isPrime[i] = false;

}

}

}

// Print prime numbers from 2 to 100

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

if (isPrime[num]) {

System.out.println(num);

}

}

}

}

**Question 12**

public class SumOfOddNumbers {

public static void main(String[] args) {

// initialize the variables

int num = 1; // the number to be added

int sum = 0; // the sum of odd numbers

// loop from 1 to 50

while (num <= 50) {

// if the number is odd, add it to the sum

if (num % 2 != 0) {

sum += num;

}

// increment the number by 1

num++;

}

// print the sum

System.out.println("The sum of odd numbers from 1 to 50 is " + sum);

}

}