MILDRED KIRERU – 22/02798

1. Explain the differences between primitive and reference data types.

Primitive data types are immutable and stored directly in memory. They include number, string, boolean, null, undefined, and symbol. Primitive data types are passed by value.

Reference data types are mutable and stored in memory as a reference or pointer to the location in memory where the value is stored. They include object, array, and function. Reference data types are passed by reference

2. Define the scope of a variable (hint: local and global variable)

Local variables are those that are initialized within a function and are unique to that function. It cannot be accessed outside of the function. Let’s look at how to make a local variable

3. Why is initialization of variables required.

This is to avoid unnecessary memory usage and enhance code readability because of the reduced scope.

4. Differentiate between static, instance and local variables.

Local Variable-Defined within a method or a code block

Instance variable-Defined outside a method at the class level

Static variable-Defined outside a method at the class level

5. Differentiate between widening and narrowing casting in java.

Widening Casting (automatically) - converting a smaller type to a larger type size. byte -> short -> char -> int -> long -> float -> double. Narrowing Casting (manually) - converting a larger type to a smaller size type. double -> float -> long -> int -> char -> short -> byte 1.

6.

|  |  |  |  |
| --- | --- | --- | --- |
| **TYPE** | **SIZE (IN BYTES)** | **DEFAULT** | **RANGE** |
| boolean | 1 bit | False | true, false |
| Char | 2 | '0’ | ‘\0000’ to ‘\ffff’ |
| Byte | 1 | 0 | -128 to + 127 |
| Short | 2 | 0 | -215 to +215-1 |
| Int | 4 | 0 | -2147,483,648 to +2147483647 |
| Long |  | 0L | - 9223375036854775808 to +9223372036854775807 |
| Float | 4 | 00.0f | +34 +38 to+34 +38 |
| Double | 8 | 0.0d | -1.8E+308 to +1.8E+308 |

7. Define package as used in java programming

a package is a group of similar types of classes, interfaces and sub-packages. It is used to group related classes together

8. Explain the importance of using Java packageser.

Help developers organize and manage their code.

Provide a way to group related classes and interfaces together.

Make it easier to find and use code.

Help prevent naming conflicts between classes.

Provide access protection

**SECTION 2**

1. **Write a Java program that asks the user to enter their sur name and current age then print the number of characters of their sir name and even or odd depending on their age number.**

**Example of Expected result:**

**If sir name is Saruni and age is 29, output will be;**

**then the number of characters is 6.**

**Your current age is an odd number**

**import java.util.Scanner;**

**public class SurnameAndAge {**

**public static void main(String[] args) {**

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

**System.out.print("Enter your surname: ");**

**String surname = input.nextLine();**

**System.out.print("Enter your current age: ");**

**int age = input.nextInt();**

**int surnameLength = surname.length();**

**String ageType = (age % 2 == 0) ? "even" : "odd";**

**System.out.println("The number of characters in your surname is: " + surnameLength);**

**System.out.println("Your current age is an " + ageType + " number");**

**}**

**}**

1. **Write Java program to ask student to enter the marks of the five units they did last semester, compute the average and display it on the screen. (Average should be given in two decimal places).**

import java.util.Scanner;

public class AverageMarks {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter the marks for the five units:");

int totalMarks = 0;

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

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

int marks = input.nextInt();

totalMarks += marks;

}

double average = (double) totalMarks / 5;

System.out.printf("The average marks is: %.2f%n", average);

}

}

1. Write a program that will help kids learn divisibly test of numbers of integers. The program should check whether the given integer is divisible by integers in the range of 0-9. For example, if a number (955) is divisible by five, the program should print, the number is divisible by 5 because it ends with a 5, and 900 is divisible by 5 because it ends with a 0(zero).

import java.util.Scanner;

public class AverageMarks {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter the marks for the five units:");

int totalMarks = 0;

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

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

int marks = input.nextInt();

totalMarks += marks;

}

double average = (double) totalMarks / 5;

System.out.printf("The average marks is: %.2f%n", average);

}

}

1. **Write a Java program to display all the multiples of 2, 3 and 7 within the range 71 to 150**.

public class MultiplesInRange {

public static void main(String[] args) {

int start = 71;

int end = 150;

System.out.println("Multiples of 2, 3, and 7 within the range " + start + " to " + end + ":");

for (int i = start; i <= end; i++) {

if (i % 2 == 0 || i % 3 == 0 || i % 7 == 0) {

System.out.println(i);

}

}

}

}

1. **Create a calculator using java to help user perform the basic operations (+, -, \* and /).**
   1. **User should be asked to enter a number, then an operation, the program computes the operation and display the output to the computer screen.**

import java.util.Scanner;

public class Calculator {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter the first number: ");

double num1 = input.nextDouble();

System.out.print("Enter the operation (+, -, \*, /): ");

char operator = input.next().charAt(0);

System.out.print("Enter the second number: ");

double num2 = input.nextDouble();

double result = 0;

switch (operator) {

case '+':

result = num1 + num2;

break;

case '-':

result = num1 - num2;

break;

case '\*':

result = num1 \* num2;

break;

case '/':

result = num1 / num2;

break;

default:

System.out.println("Invalid operator!");

return; // Terminate the program

}

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

}

}