Write a short Java method, inputAllBaseTypes, that inputs a different value of each base type from the standard input device and prints it back to the standard output device.

java

import java.util.Scanner;

public class Main {

public static void inputAllBaseTypes() {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter an integer: ");

int intValue = scanner.nextInt();

System.out.println("Integer value: " + intValue);

System.out.print("Enter a floating-point number: ");

float floatValue = scanner.nextFloat();

System.out.println("Float value: " + floatValue);

System.out.print("Enter a double-precision number: ");

double doubleValue = scanner.nextDouble();

System.out.println("Double value: " + doubleValue);

System.out.print("Enter a character: ");

char charValue = scanner.next().charAt(0);

System.out.println("Character value: " + charValue);

System.out.print("Enter a boolean (true/false): ");

boolean booleanValue = scanner.nextBoolean();

System.out.println("Boolean value: " + booleanValue);

scanner.close();

}

public static void main(String[] args) {

inputAllBaseTypes();

}

}

2- Write a short Java method, isMultiple, that takes two long values, n and m, and returns true if and only if n is a multiple of m, that is, n = mi for some integer i.

java

public class Main {

public static boolean isMultiple(long n, long m) {

if (m == 0) {

// Division by zero is undefined, so we consider it false

return false;

}

return n % m == 0;

}

public static void main(String[] args) {

long n = 15;

long m = 3;

System.out.println("Is " + n + " a multiple of " + m + "? " + isMultiple(n, m));

n = 17;

m = 5;

System.out.println("Is " + n + " a multiple of " + m + "? " + isMultiple(n, m));

}

}

3- Write a short Java method, isEven, that takes an int i and returns true if and only if i is even. Your method cannot use the multiplication, modulus, or division operators, however

java

public class Main {

public static void main(String[] args) {

int number = 10;

System.out.println(isEven(number)); // Output: true

}

public static boolean isEven(int i) {

if (i == 0) {

return true;

} else if (i == 1 || i == -1) {

return false;

}

return isEven(i - 2);

}

}

4-Write a short Java method that takes an integer n and returns the sum of all positive integers less than or equal to n.

java

public class Main {

public static void main(String[] args) {

int n = 5;

int sum = sumPositiveIntegers(n);

System.out.println("Sum: " + sum); // Output: Sum: 15

}

public static int sumPositiveIntegers(int n) {

int sum = 0;

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

sum += i;

}

return sum;

}

}

5-Write a short Java method that takes an integer n and returns the sum of all the odd positive integers less than or equal to n.

java

public class Main {

public static void main(String[] args) {

int n = 10;

int sum = sumOddPositiveIntegers(n);

System.out.println("Sum: " + sum); // Output: Sum: 25

}

public static int sumOddPositiveIntegers(int n) {

int sum = 0;

for (int i = 1; i <= n; i += 2) {

sum += i;

}

return sum;

}

}

6- Write a short Java method that takes an integer n and returns the sum of the squares of all positive integers less than or equal to n.

java

public class Main {

public static void main(String[] args) {

int n = 5;

int sum = sumOfSquares(n);

System.out.println("Sum of squares: " + sum); // Output: Sum of squares: 55

}

public static int sumOfSquares(int n) {

int sum = 0;

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

sum += i \* i;

}

return sum;

}

}

9:40 PM