

1-Realizar un programa que imprima por pantalla su **nombre** y **correo electrónico**.

Sugerencia: Utiliza `System.out.println()` para mostrar los datos en consola.

The screenshot shows a Java development environment. In the code editor, there is a single class named `HolaM` with a `main` method that prints the name and email address. In the output window, the program is run, and the output shows the printed values.

```
public class HolaM {
    public static void main(String[] args) {
        System.out.println("Nombre: Alber Rodriguez");
        System.out.println("Correo: AlberRodriguez@gmail.com");
    }
}
```

Output - JavaHolaM (run) ×

```
run:
Nombre: Alber Rodriguez
Correo: AlberRodriguez@gmail.com
BUILD SUCCESSFUL (total time: 0 seconds)
```

2-Realizar un programa que incluya las **4 operaciones matemáticas básicas** (Suma, Resta, Multiplicación y División).

Sugerencia: Declara dos variables tipo `int` o `double` y realiza las operaciones con `+`, `-`, `*`, `/`

The screenshot shows a Java development environment. In the code editor, there is a class named `HolaM` with a `main` method that performs arithmetic operations on two variables `a` and `b`. The results are printed to the console. In the output window, the program is run, and the output shows the calculated values.

```
public class HolaM {
    public static void main(String[] args) {
        int a = 10;
        int b = 5;

        System.out.println("Suma: " + (a + b));
        System.out.println("Resta: " + (a - b));
        System.out.println("Multiplicación: " + (a * b));
        System.out.println("División: " + (a / b));
    }
}
```

Output - JavaHolaM (run) ×

```
run:
Suma: 15
Resta: 5
Multiplicación: 50
División: 2
BUILD SUCCESSFUL (total time: 0 seconds)
```

3-Realizar un programa que imprima los **números impares del 1 al 100** utilizando ciclos `for`.

Sugerencia: Usa `for(int i = 1; i <= 100; i += 2)` o verifica con `if(i % 2 != 0)`

```
public class HolaM {
    public static void main(String[] args) {
        for (int i = 1; i <= 100; i += 2) {
            System.out.println(i);
        }
    }
}
```

Output - JavaHolaM (run) ×

```
run:
1
3
5
7
9
11
13
15
```

4-Realizar un programa que imprima los **números pares del 2 al 100** utilizando ciclos `while`.

Sugerencia: Inicializa `int num = 2` y ve incrementándola de 2 en 2 dentro del `while`.

```
public class HolaM {
    public static void main(String[] args) {
        int num = 2;

        while (num <= 100) {
            System.out.println(num);
            num += 2;
        }
    }
}
```

Output - JavaHolaM (run) ×

```
run:
2
4
6
8
10
12
14
16
```

5-Realizar un programa que imprima la **sumatoria de los números del 1 al 50** utilizando ciclos **do while**.

💡 **Nota:** El ciclo **do-while** ejecuta el código al menos una vez antes de verificar la condición.

💡 **Sugerencia:** Usa una variable acumuladora `int suma = 0` y ve sumando cada número.

The screenshot shows a Java code editor with the following code:

```
public class HolaM {
    public static void main(String[] args) {
        int suma = 0;
        int i = 1;

        do {
            suma += i;
            i++;
        } while (i <= 50);

        System.out.println("La sumatoria es: " + suma);
    }
}
```

Below the code editor is an output window titled "Output - JavaHolaM (run)" showing the result of the program execution:

```
run:
La sumatoria es: 1275
BUILD SUCCESSFUL (total time: 0 seconds)
```

6-Realizar un programa que mediante la utilización de bucles, debe permitir imprimir **cualquier tabla de multiplicar**.

💡 **Sugerencia:** Solicita al usuario un número usando `Scanner` y luego usa un ciclo `for` del 1 al 10 para multiplicar.

The screenshot shows a Java code editor with the following code:

```
import java.util.Scanner;

public class HolaM {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Digite un número: ");
        int num = sc.nextInt();

        for (int i = 1; i <= 10; i++) {
            System.out.println(num + " x " + i + " = " + (num * i));
        }
    }
}
```

Below the code editor is an output window titled "Output - JavaHolaM (run)" showing the result of the program execution for the number 24:

```
run:
Digite un número: 24
24 x 1 = 24
24 x 2 = 48
24 x 3 = 72
24 x 4 = 96
24 x 5 = 120
24 x 6 = 144
24 x 7 = 168
```

7-Realizar un programa que dado dos números, me indique **cual es el mayor y cual es el menor** de ambos.

Sugerencia: Usa estructuras condicionales **if-else** para comparar los dos números.

```

1 import java.util.Scanner;
2
3 public class HolaM {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Digite el primer número: ");
8         int a = sc.nextInt();
9
10        System.out.print("Digite el segundo número: ");
11        int b = sc.nextInt();
12
13        if (a > b) {
14            System.out.println("El mayor es: " + a);
15            System.out.println("El menor es: " + b);
16        } else if (b > a) {
17            System.out.println("El mayor es: " + b);
18            System.out.println("El menor es: " + a);
19        } else {
20            System.out.println("Ambos números son iguales");
21        }
22    }
23 }

Output - JavaHolaM (run) ×
▶ run:
Digitel primer nñmero: 28
Digitel segundo nñmero: 20
El mayor es: 28
El menor es: 20
BUILD SUCCESSFUL (total time: 14 seconds)

```

Realizar un programa que ingresado un numero, me indique **si es par o impar**.

Sugerencia: Usa el operador módulo **%** para verificar si el residuo de dividir entre 2 es 0 (par) o no (impar).

```

1 import java.util.Scanner;
2
3 public class HolaM {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Digite un número: ");
8         int num = sc.nextInt();
9
10        if (num % 2 == 0) {
11            System.out.println("El número es PAR");
12        } else {
13            System.out.println("El número es IMPAR");
14        }
15    }
16 }

Output - JavaHolaM (run) ×
▶ run:
Digitel número: 26
El nñmero es PAR
BUILD SUCCESSFUL (total time: 12 seconds)

```