- **Elementos y su Uso en Programación + + C**

- **;**:
  - Uso: Finaliza una declaración en C + +. Es obligatorio al final de la mayoría de las declaraciones.
  - Ejemplo:
    cpp
    int a = 10;

- **std**:
  - Uso: Es el espacio de nombres estándar en C + +. Contiene todas las funciones y clases estándar como cout, cin, string, etc.
  - Ejemplo:
    cpp
    std::string texto = "Hola, mundo!";

- **endl**:
  - Uso: Se usa para insertar un carácter de nueva línea y vaciar el buffer del flujo de salida.
  - Ejemplo:
    cpp
    std::cout << "Hola, mundo!" << std::endl;

- **String**:
    - Uso: Se utiliza para almacenar y manipular cadenas de caracteres (texto).
    - Ejemplo:
      cpp
      std::string nombre = "Juan";

- **Cin**:
    - Uso: Se usa para leer entrada desde el teclado.
    - Ejemplo:
      cpp
      int edad;
      std::cin >> edad;

- **Cout**:
    - Uso: Se usa para imprimir salida en la consola.
    - Ejemplo:
      cpp
      std::cout << "Hola, mundo!" << std::endl;

- **Else**:
  - Uso: Se usa junto con if para ejecutar un bloque de código alternativo si la condición del if es falsa.
  - Ejemplo:
    cpp
    if (edad > 18) {
        std::cout << "Eres adulto." << std::endl;
    } else {
        std::cout << "Eres menor de edad." << std::endl;
    }

- **" "**:
  - Uso: Se usan para delimitar cadenas de caracteres literales.
  - Ejemplo:
    cpp
    std::string saludo = "Hola, mundo!";

- **( )**:
  - Uso: Se usan para agrupar operaciones y parámetros en funciones.
  - Ejemplo:
    cpp
    int suma(int a, int b) {
        return a + b;
    }

- **{ }**:
  - Uso: Se usan para definir bloques de código.
  - Ejemplo:
    cpp
    if (condicional) {
        // Bloque de código
    }

- **[ ]**:
  - Uso: Se usan para acceder a elementos de arreglos y vectores.
  - Ejemplo:
    cpp
    int numeros[5] = {1, 2, 3, 4, 5};
    std::cout << numeros[0] << std::endl; // Imprime 1

- **==**:
  - Uso: Se usa para comparar dos valores para la igualdad.
  - Ejemplo:
    cpp
    if (a == b) {
        std::cout << "a y b son iguales." << std::endl;

    }

- **==**:
  - Uso: Se usa para comparar dos valores y determinar si son iguales.
  - Ejemplo:
    cpp
    if (a == b) {
        // Código a ejecutar si a es igual a b
    }

- **//**:
  - Uso: Indica el inicio de un comentario en una sola línea. El compilador ignora todo lo que viene después de // en esa línea.
  - Ejemplo:
    cpp
    // Este es un comentario de una línea

- **::**:
  - Uso: Es el operador de resolución de ámbito. Se usa para acceder a miembros de una clase, estructura, o espacio de nombres.
  - Ejemplo:
    cpp
    std::cout << "Hola, mundo!";

- **if**:
  - Uso: Se usa para ejecutar un bloque de código sólo si una condición específica es verdadera.
  - Ejemplo:
    cpp
    if (a > b) {
        // Código a ejecutar si a es mayor que b
    }

- **While** (Mientras)

  - Uso: Se utiliza para crear un bucle que repite un bloque de código mientras una condición especificada es verdadera.
  - Ejemplo:

```
cpp
int i = 0;
while (i < 5) {
    std::cout << "i es: " << i << std::endl;
    i++;
}
```

- **For**:

  - Uso: Se utiliza para crear un bucle que repite un bloque de código un número específico de veces. Es útil cuando sabes cuántas veces quieres repetir el bloque de código.
  - Ejemplo:

```
cpp
for (int i = 0; i < 5; i++) {
    std::cout << "i es: " << i << std::endl;
}
```

- **Do-While**:

  - Uso: Similar a while, pero ejecuta el bloque de código al menos una vez antes de evaluar la condición.
  - Ejemplo:

```
cpp
int i = 0;
do {
    std::cout << "i es: " << i << std::endl;
    i++;
} while (i < 5);
```

- **Switch**:
  - Uso: Se utiliza para seleccionar entre múltiples bloques de código según el valor de una expresión.
  - Ejemplo:
    cpp
    int opcion = 2;
    switch (opcion) {
        case 1:
            std::cout << "Opción 1 seleccionada" << std::endl;
            break;
        case 2:
            std::cout << "Opción 2 seleccionada" << std::endl;
            break;
        default:
            std::cout << "Opción no válida" << std::endl;
    }

- **Break**:
  -Uso: Se usa para salir inmediatamente de un bucle o de una estructura switch.
  - Ejemplo:
    cpp
    for (int i = 0; i < 10; i++) {
        if (i == 5) {
            break; // Sale del bucle cuando i es 5
        }
        std::cout << i << std::endl;
    }

- **Continue**:
  -Uso: Se usa para saltar la iteración actual de un bucle y pasar a la siguiente iteración.
  - Ejemplo:
    cpp
    for (int i = 0; i < 10; i++) {
        if (i == 5) {
            continue; // Salta la iteración cuando i es 5
        }
        std::cout << i << std::endl;
    }

- **Goto**:
  - Uso: Proporciona una forma de saltar a otra parte del código. No se recomienda generalmente debido a que puede hacer el código difícil de seguir y mantener.
  - Ejemplo:
    cpp
    int i = 0;
    punto_inicial:
    std::cout << "i es: " << i << std::endl;
    i++;
    if (i < 5) {
        goto punto_inicial; // Vuelve a la etiqueta punto_inicial
    }

- **Try-Catch**:
  - Uso: Se usa para manejar excepciones, lo que permite controlar errores en tiempo de ejecución de manera más elegante.
  - Ejemplo:
    cpp
    try {
        // Código que puede lanzar una excepción
        throw std::runtime_error("Ocurrió un error");
    } catch (const std::exception& e) {
        std::cout << "Excepción capturada: " << e.what() << std::endl;
    }

% es igual que usar /: su significado es representar la división de un número entre otro