

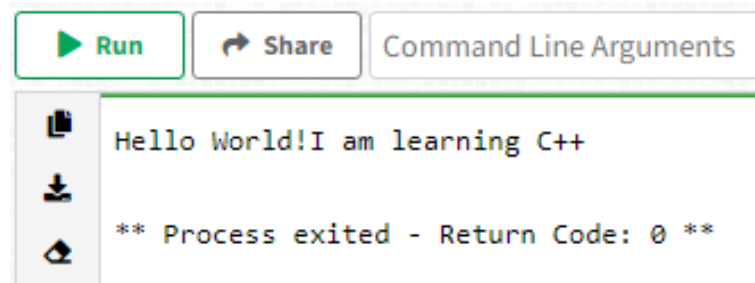
Limbajul C++, continuare

Compiler:

- https://www.onlinegdb.com/online_c++_compiler;
- <https://www.online-cpp.com/>;
- <https://onecompiler.com/cpp/42f4v8k4s>;

```
#include <iostream>
using namespace std;
```

```
int main() {
    cout << "Hello World!";
    cout << "I am learning C++";
    return 0;
}
```



```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello, World! Trecere pe următoarea linie\n";
    cout << "I am learning C++";
    return 0;
}
```

```
Hello World! Trecere pe următoarea linie
I am learning C++
```

```
** Process exited - Return Code: 0 **
```

```
#include <iostream>
using namespace std;
```

```
int main() {
    cout << "Hello World!" << endl;
    cout << "I am learning C++";
    return 0;
}
```

```
Hello World!Folosesc endl să trec pe linia urmatoare
I am learning C++
```

```
** Process exited - Return Code: 0 **
```

//același rezultat dar cu **endl**;

```
using namespace std;
```

```
int main()
{
    cout << "Size of char : " << sizeof(char) << endl;
    cout << "Size of int : " << sizeof(int) << endl;

    cout << "Size of long : " << sizeof(long) << endl;
    cout << "Size of float : " << sizeof(float) << endl;

    cout << "Size of double : " << sizeof(double) << endl;

    return 0;
}
```

```
Size of char : 1
Size of int : 4
Size of long : 8
Size of float : 4
Size of double : 8
```

<https://www.geeksforgeeks.org/cpp-data-types/>

```
#include <iostream>
#include <iostream>
#include <limits.h>
using namespace std;
```

```
int main()
{
    cout << "Size of char : " << sizeof(char) << " byte"<< endl;
    cout << "char minimum value: " << CHAR_MIN << endl;
    cout << "char maximum value: " << CHAR_MAX << endl;
    cout << "Size of int : " << sizeof(int) << " bytes"<< endl;
    cout << "Size of short int : " << sizeof(short int) << " bytes" << endl;
    cout << "Size of long int : " << sizeof(long int)<< " bytes" << endl;
    cout << "Size of signed long int : "<< sizeof(signed long int) << " bytes" << endl;
    cout << "Size of unsigned long int : " << sizeof(unsigned long int) << " bytes" << endl;
    cout << "Size of float : " << sizeof(float) << " bytes"<< endl;
    cout << "Size of double : " << sizeof(double) << " bytes" << endl;
    cout << "Size of wchar_t : " << sizeof(wchar_t) << " bytes" << endl;
    return 0;
}
```

```
Size of char : 1 byte
char minimum value: -128
char maximum value: 127
Size of int : 4 bytes
Size of short int : 2 bytes
Size of long int : 8 bytes
Size of signed long int : 8 bytes
Size of unsigned long int : 8 bytes
Size of float : 4 bytes
Size of double : 8 bytes
Size of wchar_t : 4 bytes
```

```
#include <iostream>
#include <string>
using namespace std;
```

```
int main() {
// Integer data types
int a = 10;
short b = 20;
long c = 30;
long long d = 40;
cout << "Integer data types: " << endl;
cout << "int: " << a << endl;
cout << "short: " << b << endl;
cout << "long: " << c << endl;
cout << "long long: " << d << endl;

// Floating-point data types
float e = 3.14f;
double f = 3.141592;
long double g = 3.14159265358979L;
cout << "Floating-point data types: " << endl;
cout << "float: " << e << endl;
```

Integer data types:

int: 10

short: 20

long: 30

long long: 40

Floating-point data types:

float: 3.14

double: 3.14159

long double: 3.14159

Character data types:

char: a

wchar_t: b

char16_t: 99

char32_t: 100

Boolean data type:

true: 1

false: 0

String data type:

Hello, world!

```

cout << "double: " << f << endl;
cout << "long double: " << g << endl;

// Character data types
char h = 'a';
wchar_t i = L'b';
char16_t j = u'c';
char32_t k = U'd';
cout << "Character data types: " << endl;
cout << "char: " << h << endl;
wcout << "wchar_t: " << i << endl;
cout << "char16_t: " << j << endl;
cout << "char32_t: " << k << endl;

// Boolean data type
bool l = true;
bool m = false;
cout << "Boolean data type: " << endl;
cout << "true: " << l << endl;
cout << "false: " << m << endl;

// String data type
string n = "Hello, world!";
cout << "String data type: " << endl;
cout << n << endl;

return 0;
}

```