Primitive or Fundamental Data Types

In this lesson, we will discuss the fundamental data types.

We'll cover the following

Integer data type

The integer data type comprises of all positive and negative whole numbers. We use the `int` keyword to define the integer data type. A variable of `int` type is allocated **4 bytes** of memory. It can store any value from -2^{31} to $2^{31}-1$.

```
int integer_number = 100;
```

> **Do you know?** If you store 100.5 in a variable of integer type, it would be truncated to 100.

Floating-point data type

The floating-point data type contains a number with a fractional part. We use the `float` keyword to define the floating-point data type. A variable of a `float` type is allocated **4 bytes** of memory. It can store any value from -2^{31} to $2^{31}-1$.

```
float float_number = 10.7;
```

Double data type

The double data type contains the number with the fractional part. We use the `double` keyword to define the double data type. A variable of `double` type is allocated **8 bytes** of memory. It can store any value from -2^{63} to 2^{63}-1.

```
double double_number = 10.65417;
```

Difference between float and double

The precision of a floating-point number is the number of digits that can be stored after a decimal point. A `float` can store seven digits after a decimal point precisely. Whereas, `double` can store 15 digits after a decimal point precisely. It is recommended to use `double` for floating-point values.

> 📝 **Note:** We can store scientific notation numbers in a float or double data type.

Example program

Run the code below and see the output!

```
#include <iostream>

using namespace std;

int main() {
  // Create variable of different types
  int integer_number = 10;
  float float_number = 10.5;
  /* Stores scientific value. The number after "e"
  represents the power of 10*/
  float float_scientific = 97e4;
  double double_number = 10.5;


  // Prints value of variables
  cout << "int = " << integer_number << endl;
  cout << "float = " << float_number << endl;
  cout << "float_scientific = " << float_scientific << endl;
  cout << "double = " << double_number << endl;

}
```

Numeric data types

In the code given above, we declare and initialize the variables of `int`, `float`, and `double` types. Then, we print their values on the console.

Character data type

The character data type contains a single character from the ASCII set. We use the `char` keyword to define the character data type. A variable of `char` type is allocated **1 byte** of memory. It can store any Unicode value from -2^7 to 2^7-1.

```
char character = 'b';
```

> 📝**Note:** A `char` value is always written in single quotation marks.

Boolean data type

The boolean data type stores a logical value. It can store `true` and `false`. We can also use `1` to represent `true` and `0` to represent `false`. We use the `bool` keyword to define the boolean data type. A variable of `bool` type is allocated **1 byte** of memory.

```
bool boolean = false;
```

Example program

Try running the code below!

```
#include <iostream>

using namespace std;

int main() {
  char character = 'A';
  bool boolean = true;

  // Prints value of variables
  cout << "char = " << character << endl;
  cout << "bool = " << boolean << endl;

}
```

Non-numeric data types

In the code given above, we declare and initialize the variables of `char` and `bool` types. Then, we print their values on the console.

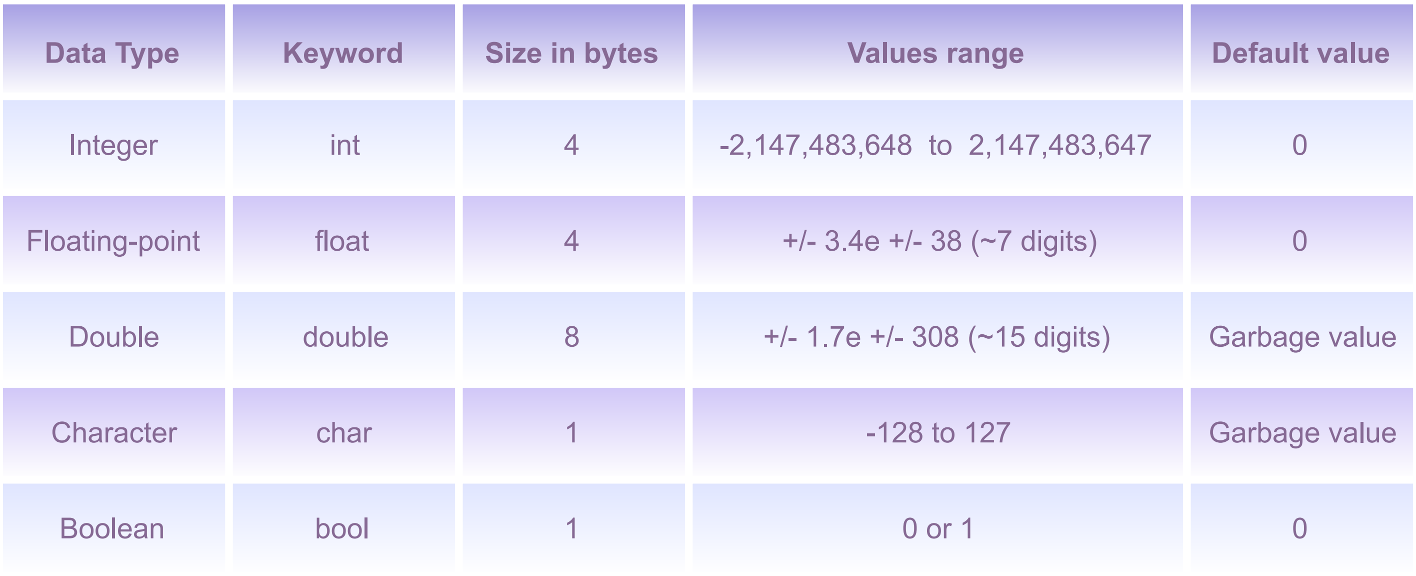
> **Do you know?** If you don’t initialize a variable of `int`, `float`, and `bool` data type, it will be automatically initialized to zero.

Void data type

The void data type represents an entity without a value. When the data type is `void`, no memory is allocated.

> **Note:** We will see the use of `void` data type in functions.

The table given below summarizes the primitive data types.



Data Type	Keyword	Size in bytes	Values range	Default value
Integer	int	4	-2,147,483,648 to 2,147,483,647	0
Floating-point	float	4	+/- 3.4e +/- 38 (~7 digits)	0
Double	double	8	+/- 1.7e +/- 308 (~15 digits)	Garbage value
Character	char	1	-128 to 127	Garbage value
Boolean	bool	1	0 or 1	0

Primitive data types

Quiz

Q

```
int main() {
  int number = 18.9;

  cout << "Number = " << number << endl;
}
```

What is the output of the code given above?

Retake Quiz

Let’s discuss the data type modifiers in the next lesson.

See you there!