First, write GUESS NUMBER and test names in PYTHON, and then compare the differences between the given C++ implementation and the PYTHON implementation.

The difference between python and C/C++ code is the main() function. C/C++ languages, being compiled in nature, mandatorily require a main() function in order to indicate to the compiler the starting point of the program. However, there is no need to declare main in Python code, unless you declare another main function in your code. Every c/c++ program has a header file and libraries, which the programmer includes at the beginning of the code in order to borrow some functionality from these libraries. In Python, these header files are replaced by modules.

In python, "import random" is logically equivalent to "#include<iostream>". Other intuitive differences are that PYTHON is a "dynamically typed language" and does not need to mention the "data type" of the variable being used. Once a value has been assigned, its type is interpreted by python's interpreter. When implementing C++, the type of a variable needs to be declared at initialization time and only the value of that data type is allowed for that variable, since C++ is a statically typed language.

Similarities

Their logic is the same, what differs is the syntax of your program , so main focus should be on learning the syntax that varies.

Differences

C++ is a compiled programming language, while Python is an interpreted programming language

Compiled languages have a separate compilation process that translates the program into machine language before the program is executed. When this program is executed later, there is no need to compile it and the executable file is allowed directly.

An interpreted language is a language that uses a specialized interpreter to interpret the source program line by line into platform-specific machine code and execute it immediately. An interpreted language does not usually do the whole compilation and linking process; an interpreted language is equivalent to mixing the compilation and interpretation processes of a compiled language and doing them together at the same time.