

# Compile time vs Runtime

Compile-time and Runtime are the two programming terms used in the software development. Compile-time is the time at which the source code is converted into an executable code while the run time is the time at which the executable code is started running. Both the compile-time and runtime refer to different types of error.

## Compile-time errors

Compile-time errors are the errors that occurred when we write the wrong syntax. If we write the wrong syntax or semantics of any programming language, then the compile-time errors will be thrown by the compiler. The compiler will not allow to run the program until all the errors are removed from the program. When all the errors are removed from the program, then the compiler will generate the executable file.

The compile-time errors can be:

- Syntax errors
- Semantic errors

## Syntax errors

When the programmer does not follow the syntax of any programming language, then the compiler will throw the syntax error.

For example,

```
int a, b:
```

The above declaration generates the compile-time error as in C, every statement ends with the semicolon, but we put a colon (:) at the end of the statement.

## Semantic errors

The semantic errors exist when the statements are not meaningful to the compiler.

For example,

```
a+b=c;
```

The above statement throws a compile-time errors. In the above statement, we are assigning the value of 'c' to the summation of 'a' and 'b' which is not possible in C programming language as it can contain only one variable on the left of the assignment operator while right of the assignment operator can contain more than one variable.

The above statement can be re-written as:

```
c=a+b;
```

## Runtime errors

The runtime errors are the errors that occur during the execution and after compilation. The examples of runtime errors are division by zero, etc. These errors are not easy to detect as the compiler does not point to these errors.

### Let's look at the differences between compile-time and runtime:

| Compile-time  | Runtime  |
|---|--|
| The compile-time errors are the errors which are produced at the compile-time, and they are detected by the compiler. | The runtime errors are the errors which are not compiler and produce an unpredictable result time. |
| In this case, the compiler prevents the code from execution if it detects an error in the program.                    | In this case, the compiler does not detect the error and prevent the code from the execution.      |
| It contains the syntax and semantic errors such as missing semicolon at the end of the statement.                     | It contains the errors such as division by zero, square root of a negative number.                 |

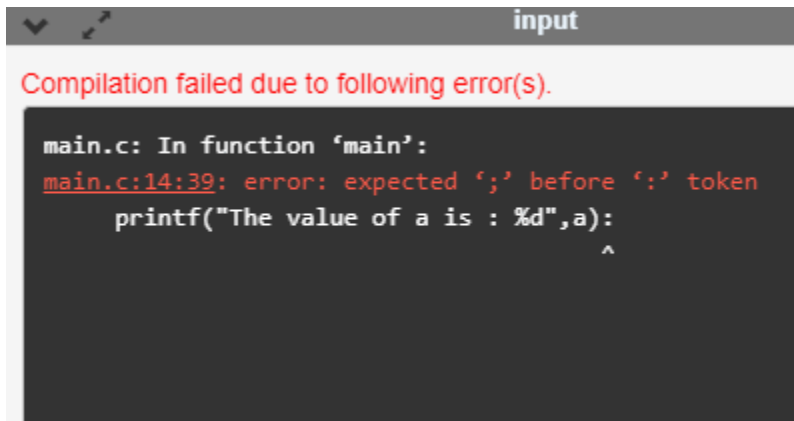
### Example of Compile-time error

1. `#include <stdio.h>`
2. `int main()`
3. `{`
4. `int a=20;`
5. `printf("The value of a is : %d",a);`

6. `return 0;`
7. `}`

In the above code, we have tried to print the value of 'a', but it throws an error. We put the colon at the end of the statement instead of a semicolon, so this code generates a compile-time error.

### Output



```
input
Compilation failed due to following error(s).
main.c: In function 'main':
main.c:14:39: error: expected ';' before ':' token
printf("The value of a is : %d",a):
                                ^
```

### Example of runtime error

1. `#include <stdio.h>`
2. `int main()`
3. `{`
4. `int a=20;`
5. `int b=a/0; // division by zero`
6. `printf("The value of b is : %d",b);`
7. `return 0;`
8. `}`

In the above code, we try to divide the value of 'b' by zero, and this throws a runtime error.