

Init Statement for if and switch

C++17 adds new features to 'if' and 'switch' statements. See more details in this lesson.

C++17 provides new versions of the `if` and `switch` statements:

```
if (init; condition)
```

And

```
switch (init; condition)
```

In the `init` section you can specify a new variable, similarly to the `init` section in `for` loop. Then check the variable in the `condition` section. The variable is visible only in `if/else` scope.

To achieve a similar result, before C++17 you had to write:

```
{  
    auto val = GetValue();  
    if (condition(val))  
        // on success  
    else  
        // on false...  
}
```



Please notice that `val` has a separate scope, without that it 'leaks' to enclosing scope.

Now, in C++17, you can write:

```
if (auto val = GetValue(); condition(val))  
    // on success  
else  
    // on false...
```



Notice that `val` is visible only inside the `if` and `else` statements, so it doesn't 'leak'.

leak.

condition might be any boolean condition.

Here is an example:

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

int main() {

    srand (time(NULL));

    if (auto val = (rand() % 100); val > 50)
        // on success
        cout << val << " is greater than 50";
    else
        // on false...
        cout << val << " is less than 50";
}
```



Why is this useful? We'll discuss and example in the next lesson.