Removing Deprecated Exception Specifications

This lesson shows how exception handling has evolved in C++17.

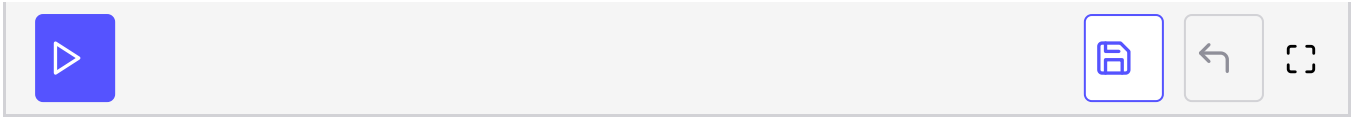
In C++17, exception specification will be part of the type system (as discussed in the next chapter about Language Clarification).

However, the standard contains an old and deprecated exception specification that appeared to be impractical and unused.

For example:

```
#include <iostream>

void fooThrowsInt(int a) throw(int) {
  printf("can throw ints\n");
  if (a == 0)
    throw 1;
}
```



Pay special attention to that `throw(int)` part.

The above code has been deprecated since C++11. The only practical exception declaration is `throw()` which means - this code won't throw anything. Since C++11 it's been advised to use `noexcept`.

For example in clang 4.0 you'll get the following error:

```
error: ISO C++1z does not allow dynamic exception specifications
[-Wdynamic-exception-spec] note: use 'noexcept(false)' instead
```

> *Extra Info:* The change was proposed in: P0003R5.

Let’s move further and look at more removed features.