Summary

Let's summarise the chapter.

WE'LL COVER THE FOLLOWING

- All Attributes available in C++17

All Attributes available in C++17

Attribute	Description
`[[noreturn]]`	a function does not return to the caller
`[[carries_dependency]]`	extra information about dependency chains
`[[deprecated]]`	an entity is deprecated
`[[deprecated("reason")]]`	provides additional message about the deprecation
`[[fallthrough]]`	indicates a intentional fall-through in a switch statement
`[[nodiscard]]`	a warning is generated if the return value is discarded
`[[maybe_unused]]`	an entity might not be used in the code

Each compiler vendor can specify their syntax for attributes and annotations. In Modern C++, the ISO Committee tries to extract common parts and standardise it as `[[attributes]]`.

There's also a relevant quote from Bjarne Stroustrup's C++11 FAQ about suggested use:

> There is a reasonable fear that attributes will be used to create language dialects. The recommendation is to use attributes to only control things that do not affect the meaning of a program but might help detect errors (e.g. `[[noreturn]]`) or help optimisers (e.g. `[[carries_dependency]]`).

Now that you've made it to the end of the course. It's time for a short quiz to evaluate your understanding of the concepts in the next section.