Wrap Up

Let's summarise this chapter.

Here are the things to remember about std::string_view:

- It's a specialisation of std::basic_string_view<charType, traits<charType>> With charType equal to char.
- It's a non-owning view of a contiguous sequence of characters.
- It might not include null terminator at the end.
- It can be used to optimise code and limit the need for temporary copies of strings.
- It contains most of std::string operations that don't change the underlying characters.
- Its operations are also marked as constexpr.

But:

- Make sure the underlying sequence of characters is still present!
- While std::string_view looks like a constant reference to the string, the language doesn't extend the lifetime of returned temporary objects that are bound to std::string_view.
- Always remember to use stringView.size() when you build a string from string_view. The size() method properly marks the end of string_view.
- Be careful when you pass string_view into functions that accept null-terminated strings unless you're sure your string_view contains a null terminator.

Test your newly learnt knowledge with a quick quiz.