Maps in C++17 Solutions

Loops and Structured Bindings

- (Optional)
- Write a simple program which populates a std::map object
- Use a structured binding to print out its elements

Checking std::map insert() in C++17

- (Optional)
- Write a program which calls insert() to add an element to an std::map object
- Use a structured binding to check the return value from the call

insert_or_assign()

- Briefly describe the insert_or_assign() member function of std::map
  - This has the same functionality as operator []
  - If there is an element in the map with the same key, it will assign to its value member
  - If there is no element in the map with the same key, it will insert a new element
  - However, it has the same advantages as the insert() member function:
  - No requirement that the value's type has a default constructor
  - If an exception is thrown, the map is unmodified
  - The return value allows us to determine whether a new element was inserted

insert_or_assign()

- (Optional)
- Write a simple program which populates a std::map object
- Call the map's insert_or_assign() member function and use a structured binding to check its return value
- Modify your program to use an if statement with an initializer