Template Instantiation

In this lesson, we'll learn about template instantiation.

WE'LL COVER THE FOLLOWING

Template Instantiation

Templates can be implicitly and explicitly instantiated. Implicit instantiation means automatically and explicit means manually.

Implicit

```
std::vector<int> vec{};
bool isSmaller<double>(fir, sec);
bool isSmaller(fir, sec);
```

Explicit

```
template class std::vector<int>;
template bool std::vector<double>::empty() const;
template bool isSmaller<double>(double, double);
template bool isSmaller(double, double);
```

Lazy Evaluation

When a class is instantiated, only the method declarations are available.

> The definition of a method is only instantiated when it is used.

It is not necessary that all methods of class templates are valid for the template arguments. You can only use the methods, which are valid for a given instantiation.

In the next lesson, we'll look at an example of template instantiation.