# Article information:

Overview of modules in C++ | Microsoft Learn  
<https://learn.microsoft.com/en-us/cpp/cpp/modules-cpp?view=msvc-170>

# Article summary:

1. C++20 introduces modules, a modern solution that turns C++ libraries and programs into components.

2. Modules eliminate or reduce many of the problems associated with the use of header files and often reduce compilation times.

3. Visual Studio 2022 version 17.1 has full support for C++20 standard modules, and Visual Studio 2022 version 17.5 has full support for importing the Standard Library as modules.

# Article rating:

Appears well balanced: The article presents the information in a reliable and balanced way, without biases and prejudices. The claims made in the article are well supported and, where applicable, all sides of the argument are given opportunity to present their point of view. The article appears trustworthy and reliable.

# Article analysis:

The article is overall reliable and trustworthy, providing an overview of modules in C++ from Microsoft Learn. It provides clear information on how to enable modules in the Microsoft C++ compiler, how to consume the C++ Standard Library as modules (experimental), a basic example, module grammar, implementing modules, and more. The article also provides links to other related articles for further reading on topics such as comparing header units, modules, and precompiled headers; importing the C++ standard library using modules; and more.

The article does not appear to have any biases or one-sided reporting; it presents all relevant information in an unbiased manner without any promotional content or partiality towards any particular point of view. All possible risks are noted where applicable (e.g., when discussing experimental implementations). The article also does not appear to be missing any points of consideration or evidence for its claims made; all claims are supported by evidence provided in the text or linked resources. Furthermore, there are no unexplored counterarguments or missing sides presented equally; both sides are presented equally throughout the article with no favoritism towards either side.

In conclusion, this article is reliable and trustworthy overall with no apparent biases or one-sided reporting present in its content.

# Topics for further research:

* C++ Standard Library modules
* Comparing header units and modules
* Precompiled headers and modules
* Module grammar in C++
* Implementing modules in C++
* Importing C++ Standard Library using modules

# Report location:

<https://www.fullpicture.app/item/c92ad6c542957cbdd89f793b60be476f>