Include guard

В языке C++ недопустимо повторное объявление одного и того же идентификатора. Однако, оно часто возникает в модульном программировании, к примеру, при подключении заголовочных файлов.

В языках программирования Си и C++ **#includeguards** (**защита подключения**), иногда также называемая **macroguard** (**макрозащита**) — это особая конструкция, применяемая для избежания проблем с «двойным подключением» при использовании директивы компилятора #include. Добавление #includeguards в заголовочный файл является одним из способов сделать этот файл идемпотентным, то есть таким, что многократные его подключения эквивалентны однократному и не приводят к ошибкам.

Решение простое — использовать header guards (защиту подключения в языке C++). **Header guards** — это директивы [**условной компиляции**](https://ravesli.com/urok-22-preprotsessor-direktivy/#toc-4), которые состоят из следующего:

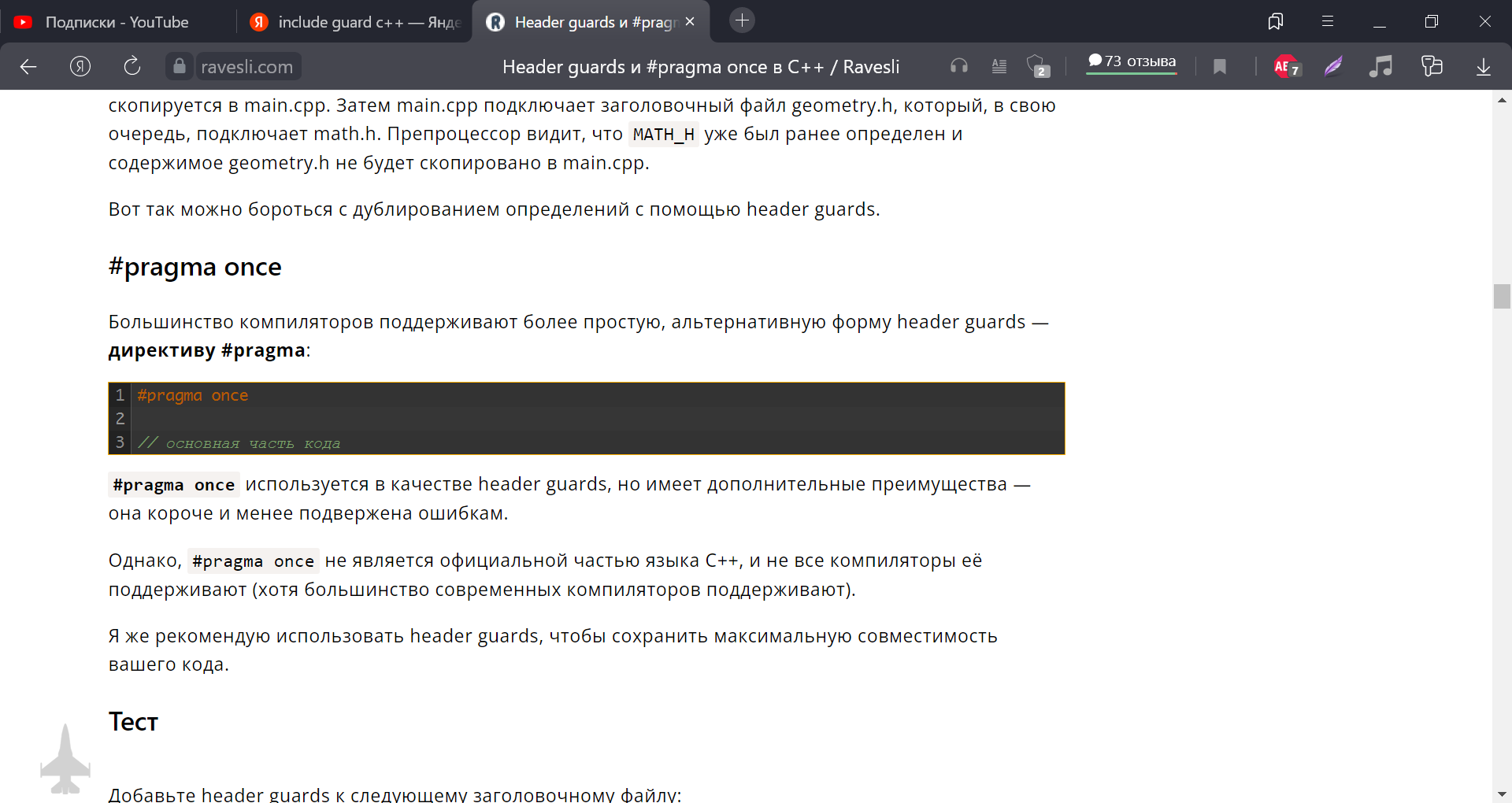
|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #ifndef SOME\_UNIQUE\_NAME\_HERE  #define SOME\_UNIQUE\_NAME\_HERE    // Основная часть кода    #endif |

Если подключить этот заголовочный файл, то первое, что он сделает — это проверит, был ли ранее определен идентификатор SOME\_UNIQUE\_NAME\_HERE. Если мы впервые подключаем этот заголовок, то SOME\_UNIQUE\_NAME\_HERE еще не был определен. Следовательно, мы определяем SOME\_UNIQUE\_NAME\_HERE (с помощью директивы #define) и выполняется основная часть заголовочного файла. Если же мы раньше подключали этот заголовочный файл, то SOME\_UNIQUE\_NAME\_HERE уже был определен. В таком случае, при подключении этого заголовочного файла во второй раз, его содержимое будет проигнорировано.

Все ваши заголовочные файлы должны иметь header guards. SOME\_UNIQUE\_NAME\_HERE может быть любым идентификатором, но, как правило, в качестве идентификатора используется имя заголовочного файла с окончанием \_H.

## #pragma once

Большинство компиляторов поддерживают более простую, альтернативную форму header guards — **директиву #pragma**:



**#pragma once** используется в качестве header guards, но имеет дополнительные преимущества — она короче и менее подвержена ошибкам. Однако, #pragma once не является официальной частью языка C++, и не все компиляторы её поддерживают.

pragma once – для краткости и простоты, header guards – для надёжности и совместимости.

Источники:

<https://ravesli.com/urok-23-header-guards/>

Конспект Александра Номоконова