

Шаблонный класс.

Шаблонный позволяет определить универсальные типы, которые не зависят от определенного типа.

Шаблонный класс позволяет задать тип объектов, используемых в классе.

Шаблонные классы позволяют обобщенную логику для различных типов. Когда мы не знаем какого типа объект будет использоваться в классе, можно задать шаблонный класс, который в последствии можно конкретизировать определенным типом.

Обобщенное описание — шаблон класса.

Шаблонный класс — класс, порожденный из шаблона класса, после передачи типа в качестве шаблонного аргумента.

При использовании имени `Stack` внутри определения шаблона класса `Stack` список пар-ов $\langle T \rangle$ можно опустить. При использовании `Stack` как-то объекта-члена другого класса необходимо указать `Stack < T >`.

При определении методов шаблона класса вне, нужно указать перед его именем конструкцию `Stack < T >`,

а перед самим определением `template < class T >`. Также для выходного значения.

Функция-член шаблона класса сама является шаблоном. Стандарт C++ требует, чтобы она конкретизировалась только при вызове, либо при взятии ее адреса. Некоторые старые компиляторы конкретизируют такие функции одновременно с конкретизацией самого шаблона класса. При конкретизации функции-члена

используется тип того объекта, для которого ф-ция вызвана. Например

```
Stack <int> stack;
```

ф-ция конкретизируется типом `int`.

При определении шаблона класса нужно руководствоваться правилом: он должен предоставлять только необходимый функционал. Некоторые ф-ции-члены могут исп-ть для нек-а типов, но тем не менее данный класс все равно можно будет использовать. Невозможность исп-ия всех операций шаблоном класса.