1. Шаблоны автоматизируют создание функций и классов с разными типами
2. template <параметры>

заголовок функции

тело функции

1. template <параметры>

calss имя класса

тело класса

1. Параметры – это то, что мы передаем в шаблоне функции, чтобы использовать это в обращениях к функции
2. Имена параметров шаблона должны быть уникальными во всем определении шаблона.

Список параметров шаблона не может быть пустым.

В списке параметров шаблона может быть несколько параметров, и каждому из них должно предшествовать ключевое слово class.

1. template<class type1, class type2>
2. Да, шаблон функции может быть перегружен. Например, можно определить другие шаблоны, имеющие то же самое имя функции, но различные наборы параметров
3. Компонентные функции параметризованного класса автоматически являются параметризованными. Их не обязательно объявлять как параметризованные с помощью template.

Дружественные функции, которые описываются в параметризованном классе, не являются автоматически параметризованными функциями, т.е. по умолчанию такие функции являются дружественными для всех классов, которые организуются по данному шаблону.

Если friend-функция содержит в своем описании параметр типа параметризованного класса, то для каждого созданного по данному шаблону класса имеется собственная friend-функция.

В рамках параметризованного класса нельзя определить friend-шаблоны (дружественные параметризованные классы).

С одной стороны, шаблоны могут быть производными (наследоваться) как от шаблонов, так и от обычных классов, с другой стороны, они могут использоваться в качестве базовых для других шаблонов или классов.

Шаблоны функций, которые являются членами классов, нельзя описывать как virtual.

Локальные классы не могут содержать шаблоны в качестве своих элементов.

Статические переменные шаблонов классов необходимо инициализировать для каждого используемого типа данных

1. В