

Упаковка возвращаемого типа функции (не применимо к конструкторам/операторам):

```
|| template < ... >  
|| enable_if_t < Condition, Return_Type > f(args)
```

Добавление параметра функции с аргументом по умолчанию (Dummy необязательно):

```
|| template < ... >  
|| Return_Type f(args, enable_if_t < Condition, T > Dummy = v)
```

Добавление параметра шаблона с аргументом по умолчанию (по умолчанию тип `void`):

```
|| template < ..., typename Dummy = enable_if_t < Condition > >
```

Упаковка нешаблонного аргумента:

```
|| template < typename T >  
|| void f(T & t, enable_if_t < is_something < T > ::value, string & > s)
```

Выбор типа аргумента по умолчанию (добавление параметра функции с аргументом по умолчанию):

```
|| template < ... >  
|| Return_Type f(args, enable_if_t < Condition, void ** > Dummy = NULL)
```