

C++ allows for us to use the power of templates in building our own classes.



Templated Functions

A template variable is defined by declaring it before the beginning of a class or function:

```
template <typename T>
int max(T a, T b) {
  if (a > b) { return a; }
  return b;
}
```



Compile-Time Binding

Templated variables are checked at compile time, which allows for errors to be caught before running the program:



cpp-templates/main.cpp

```
15 template <typename T>
16 T max(T a, T b) {
17 |
    if (a > b) { return a; }
18 |
    return b;
19 }
20
21
   int main() {
22
     cout << "max(3, 5): " << max(3, 5) << endl;</pre>
23 |
     cout << "max('a', 'd'): " << max('a', 'd') << endl;</pre>
     cout << "max(\"Hello\", \"World\"): " << max("Hello", "World")</pre>
24
                                                                        << endl:
25
     cout << "max( Cube(3), Cube(6) )" << max( Cube(3), Cube(6) ) << endl;</pre>
26
27
     return 0;
28
```