The to_string() Functions

In the last few lessons, you've looked at the new C++ 11 functions for converting `string` values to **numbers**, such as `stoi()`, `stod()` and so on. Along with these functions, C++ 11 also introduced a group of which go the other direction, converting **numbers to `string`**. Instead naming the functions `itos()` or `dtos()`, (similar to `stoi()`), these **overloaded functions** are **all** named `to_string()`.

Just like the string-to-number conversion functions, the `to_string()` functions do not appear in C++98. So, as you did with `stoi()` and friends, you must to write your our own, using the `ostringstream` class like this:

```
string to_string(int n)
{
    ostringstream out;
    out << n;
    return out.str();
}
```

You can easily **overload** your version of `to_string()` to work with other types.

```
string to_string(double n) { . . . }
string to_string(long n) { . . . }
string to_string(unsigned n) { . . . }
```

The body of each function will have **the identical code** to the `int` version. Wrap all of these up inside an `#if __cplusplus <= 199711L` **preprocessor directive**, and you should be able to use the `to_string()` function in both C++ 17 and in C++ 98.