The Working Constructor

As you saw in the last lesson, working constructor is the short-hand description of a constructor that takes as many user-supplied arguments as possible. In the `Time` class the working constructor looks like this:

```
Time(int hours, int minutes);
```

In the `.cpp` file, you might have code that looks something like this, using the initializer list, also from the last lesson:

```
Time::Time(int hours, int minutes)
  : m_hours(hours), m_minutes(minutes)
{
    // validate the constructor arguments
    assert(hours >= 0 && hours < 24);
    assert(minutes >= 0 && minutes < 60);
}
```

It is important that your constructor intitializes your object so that it is **in a valid state**. In the example shown here, we've used `assert()` on the assumption that it is a programming error if an invalid `Time` is constructed. If, however, you were constructing `Time` objects using external data, it is possible you would want to `throw` and `catch` an exception instead.