

# Setters

---

In Java, many classes have member functions that start with `set`. These are called **mutators**, since they **change the state** of the object. Mutators should **validate data** written to the object to **enforce the class invariants**.

With properly written mutators, the errors described in earlier lessons **cannot occur**. Consider your `Time` class. If you were to add `setHours()` and `setMinutes()` members to the class, you would have to enforce these restrictions:

- `m_hours` must be between `0` and `23` inclusive.
- `m_minutes` must be between `0` and `59` inclusive.

Unlike the `read()` member function, where you could put the stream into a failed state, if these conditions were not met, in a mutator you need to **throw** an exception like this:

```
void Time::setHours(int h)
{
    if (h < 0 || h > 23) throw out_of_range("...");
    m_hours = h;
}
```



This course content is offered under a CC Attribution Non-Commercial license. Content in this course can be considered under this license unless otherwise noted.