

The Public Interface

The goal of information hiding is to make it possible for clients to **use** `Time` objects **without** ever directly accessing the data members themselves.

To do that, you create a collection of **functions**, which provides **a public interface** for your type. These functions are specially dedicated to an individual class, so they are called **member functions**.

Your interface should be as **small as possible**, yet comprehensive enough to meet all of your client's needs. What features might those be? (Let's consider **H01**).

- **Arithmetic**: calculate the **difference** and **sum** of two `Time` objects
- **Input and Output**: read into and write out a `Time` object

Here is your `Time` structure, including the above interface:

```
struct Time
{
    int hours;
    int minutes;

    Time sum(const Time& rhs);
    Time difference(const Time& rhs);

    std::istream& read(std::istream& in);
    std::ostream& write(std::ostream& out);
};
```



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.