

Printing Out Class Member Data

Solutions

- Add a print() member function to the class below which displays the value of the data members on the screen
- Create a program which calls the print() method

```
class refrigerator {  
    // Data members  
    double temperature;  
    bool door_open;  
    bool power_on;  
};
```

- Modify the `print()` member function so that it takes a `std::ostream` as its argument and sends the data there
- Create a program which implements the `print()` method and calls it, passing `cout` as the output stream argument
- Does it matter whether the output stream is passed by reference or by value?
 - Streams cannot be passed by value as they are not copyable
- Modify your program so that it opens a file. Call the `print()` member function to save the data to the file

- Explain why the code below does not work
 - Stream operators << are only provided for built-in and library types
 - There is no operator << that can take fridge as its argument
- What changes would you need to make for the code to work?
 - We need to provide a suitable operator <<

```
cout << "Current state of fridge: " << fridge << endl;
```

- The code below is converted into function calls. What would the code look like after being converted?

```
int i{1}, j{2};  
cout << i << j;  
operator <<(operator <<(cout, j), i);
```

- Explain why the overloaded << operator returns a reference to the output stream
 - The output stream is not copyable, so it can only be returned by reference
 - Returning the stream means that operator << calls can be chained

```
operator <<(operator <<(cout, j), i);
```
 - The next call in the chain needs to modify the stream by pushing data on it, so it must be a modifiable reference

- Modify your program so that the code below now compiles and runs correctly

```
cout << "Refrigerator status: " << fridge << endl;
```