

Stream Manipulators and Formatting Solutions

Stream Manipulators

- Explain what is meant by a "stream manipulator"
 - A stream manipulator is pushed onto a stream to modify the stream's behaviour
 - Usually, this alters the formatting of output
 - `std::flush` and `std::endl` affect the timing of the stream's next buffer flush

Printing Booleans

- Write a simple program which prints out Booleans as "true" or "false"

setw() manipulator

- What does setw() do?
 - It puts extra spaces into an output field to make it up to the required width
- Describe how to use setw()
 - Push setw onto the output stream
 - Pass the required width of the following data output field as the argument to setw()
 - The next data item pushed onto the stream after the setw() call will be padded to the required width

Left Justification

- What is meant by "justification" in the context of text formatting?
 - It describes where extra spaces are added
 - Left-justification means the text is at the left of the output field, with the extra spaces added after the data
 - Vice versa for right-justification
- How can the justification be changed for a C++ stream?
 - The left manipulator makes the output left justified.
 - The right manipulator (which is the default) makes the output right justified

Left Manipulator

- Write a simple program which produces output formatted like this

```
Penguins 5  
Polar Bears 2
```

"Sticky" Manipulators

- What does it mean when a manipulator is "sticky"?
 - A sticky manipulator permanently changes the state of the stream
 - The stream will remain in this state until we push another manipulator which has the opposite effect
- How does this differ from a non-sticky manipulator?
 - A non-sticky manipulator only affects the next operation on the stream

Setfill Manipulator

- What effect does applying the setfill manipulator have on the output?
 - Instead of using a space character for padding, the argument of setfill will be used
- Write a simple program which produces output formatted like this

```
#####Penguins 5  
###Polar Bears 2
```