

# Stream Member Functions and Formatting

## Formatting Output

```
cout << "Penguins " << 5 << "\n" << "Polar Bears " << 2 << "\n";
```

// The output looks like this:

Penguins 5

Polar Bears 2

- Sometimes we want to present our results in the form of a table. To make the output line up, we need to make all the output fields the same width

Penguins      5

Polar Bears   2

- The setw() manipulator will pad the output field to make it the width of its argument. It differs from other manipulators we have seen so far:
  - It is defined in <iomanip>, not <iostream>
  - It takes an argument
  - It does not "stick" - it only affects the next data item pushed on the stream

## setw() example

```
cout << setw(15) << "Penguins" << 5 << "\n" << setw(15) << "Polar Bears" << 2 << "\n";
```

// The output looks like this:

```
Penguins5
```

```
Polar Bears2
```

- By default, `setw()` will add blanks at the left of the output field, making the output right-justified

- We can use the `left` manipulator to make the output left-justified, by adding padding blanks at the right:

```
cout << left << setw(15) << "Penguins" << 5 << "\n" << setw(15) << "Polar Bears" << 2 << "\n";
```

// The output looks like this:

```
Penguins      5
```

```
Polar Bears   2
```

- This manipulator is "sticky"
- To restore the default setting, apply the `right` manipulator

- The `setfill()` manipulator from `<iomanip>` sets the padding character to its argument

```
cout << setfill('#');           // Set the padding character to #  
cout << left << setw(15) << "Penguins" << 5 << "\n" << setw(15) << "Polar Bears" << 2 << "\n";  
cout << setfill(' ');          // Set the padding back to the default space character
```

// The output looks like this:

```
Penguins#####5
```

```
Polar Bears####2
```

- Unlike `setw()`, this is "sticky" and should be reset after use

## Summary

- C++ manipulators can be used to change the properties of output fields. This is useful for displaying results in tables
- `setw()` sets the width of a single output field by adding padding characters
- `left` and `right` make the output left and right justified (right is the default)
- `setfill()` changes the padding character used by `setw()`