Learning Objectives: String Iteration

- **Define string iteration**
- **Identify two ways to iterate over a string**
- **Explain the inner workings of string iteration**

Iteration: For Loop

Iterating Over Strings

Iterating over a string allows you to deal with each character of a string individually without having to repeat certain commands. You start with the character at index `0` and move through the end of the string.

```
string my_string = "Hello world";

for (int i = 0; i < my_string.length(); i++) {
  cout << my_string.at(i);
}
```

→ Hello world

challenge

What happens if you:

- Change the value of `my_string` to `"\u25A3\u25A8\u25D3\u25CC\u25A2"`?
  - **Note:** Some Unicode characters are not compatible with `cout` and/or `endl;` commands.
- Change the value of `my_string` to `"10, 11, 12, 13, 14"`?
- Change the `cout` statement to `cout << my_string.at(i) << endl;`?
- Change the `cout` statement to `cout << my_string;`?

Note that you can also use a **range-based** or **enhanced** `for` loop to iterate over strings. Make sure to cast the iterating variable as `char`!

```
string my_string = "Hello world";

for (char c : my_string) {
  cout << c;
}
```

→ Hello world

Iteration: While Loop

While Loop

String iteration is most often done with a `for` loop. However, a `while` can be used as well.

```
string my_string = "Calvin and Hobbes";
int i = 0;

while (i < my_string.length()) {
  cout << my_string.at(i);
  i++;
}
```

→ Calvin and Hobbes

challenge

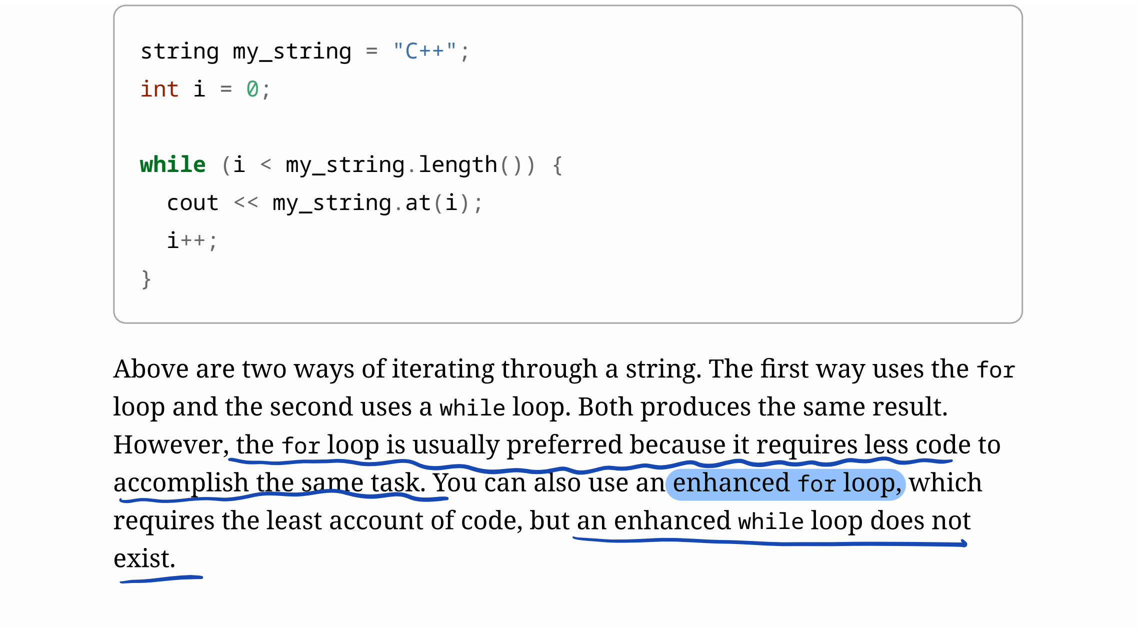
What happens if you:

- Change the loop to `while (i <= my_string.length())`? → error
- Copy the original code but change the `cout` statement to `cout << i;`? → 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
- Copy the original code but remove `i++;`? → endless loop

Comparing While & For Loops

```
string my_string = "C++";

for (int i = 0; i < my_string.length(); i++) {
  cout << my_string.at(i);
}
```



```
string my_string = "C++";
int i = 0;

while (i < my_string.length()) {
  cout << my_string.at(i);
  i++;
}
```

Above are two ways of iterating through a string. The first way uses the `for` loop and the second uses a `while` loop. Both produces the same result. However, the `for` loop is usually preferred because it requires less code to accomplish the same task. You can also use an enhanced `for` loop, which requires the least account of code, but an enhanced `while` loop does not exist.