Overview

This method of outputting strings will probably be the most common one you use. It is efficient, easy to maintain, and easy to make it look exactly the way you want. This method is called a `formatted string`.

Important Information

When it comes to formatted strings, the syntax is always the same. You put the character `f` outside of your string, and then everything else goes inside the string.

Handlebars

A common practice in multiple programming languages is using curly braces as “handlebars” to signify when a the value of a variable should be inserted to the string. In python formatted strings, `{` and `}` are used.

Strings Within Strings

Sometimes in a formatted string, you want to display quotation marks. both `'` and `"` can be used to begin a string, but the end of the string must use the same kind of quotation marks. That means if you want to display quotation marks, you just need your string to be enclosed in the other kind.

```python
my_string = "my 'string' inside quotes"
```

Escape Charcters

Occasionally you will also want a special character that is being interpreted as code to just be part of the string (like the quotation marks or handlebars for example). Escape characters like `\` can be used to ignore the special properties of these characters.

```python
my_string = "\"hello world!\""
print(my_string)
```

Set Up

Create a new file called `formattedstring.py`.

Copy

Run the following code and see what happens.

```python
fav_number = 99
name = "Mr. Forsyth"

print(f"Hello {name}, {fav_number} is a great favourite number!")

height_feet = input("How many feet tall are you?")
height_inch = input("And how many inches?")

print(f"You must be {height_feet} feet tall and {height_inch} inches.")
```

Change

Change the code so that the program asks for their birthday and displays that in the output at some point.

Challenge

Ask the user for their birth year, and then have the program output their possible age.