Computer Science
Programming Fundamentals

Basic Output

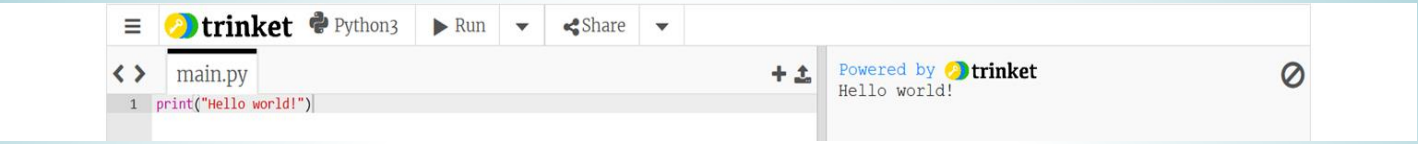
Basic Output Intro

For other people to know what we are thinking about, we have to say what's on our mind! Similarly, we can't read the computer's "mind." We have to tell it to communicate with us. When the computer gives us information, we call it **output**. Output can include words on the screen, pictures, and sound, among other things.

In this lesson, we'll learn how to tell Python to write words on the screen. Why do we need to know this? Because getting feedback from our programs lets us know if they're working correctly and lets the user know what's going on.

Basic Output pt. 1

- Run the code in the example.



Q: Based on the output we see on the right side, what must `print` be doing?

A: It takes what is between the parentheses and outputs it on the right side.

Q: Why didn't it output the quotes?

- Take away the quotes and run the code.

A: Python uses the quotes to understand what type of data it is dealing with. It needs them in this case.

Basic Output pt. 2

- Change the code so it outputs the following, running it for each item.
    - your name

```
print("Darth Vader")
```

    - That was interesting

```
print("That was interesting")
```

    - That was “interesting”

```
print("That was "interesting"")
```

Q: Why the error?

A: Some characters are special and need to be **escaped** to be printed exactly as they are. In this case, we need to use a backslash (****) to escape each quotation mark.

- Run:

```
print("That was \"interesting\"")
```

Basic Output pt. 3

Each `print` statement outputs on its own line.

- Run:

```
print("The sentence below is true.")
print("The sentence above is false.")
```

Basic Output pt. 4

Activity:

- Use your new Python print statement skills to output the good boy below.

```
   ^..^      /
   /_/\_____/
       /\   /\
      /  \ /  \
```

Hint: Remember that each print statement outputs onto a new line.

Hint: You may have to escape a few characters!

Ideal answer on the next page.

Basic Output pt. 5

Ideal answer:

```
print("^..^      /")
print("/_/\_____/")
print("   /\   /\\")
print("  /  \ /  \\")
```

🔎 Explanation:

The last two lines each contain an extra backslash (orange) to escape the backslash after it (green). Without the extra backslash, Python doesn't know where the print statement ends!