# Chapter 1 - Basic Skills

## Fundamentals

## Printing

## Learning Objectives - Printing

- **Use the `print` command to write text to the screen**
- **Remove the newline character from the `print` statement**

# Printing to the Console

## Printing to the console

As you learn Python, you will write code in the text editor to the left. Use a `print` command to see the output of your program. Enter the code below (or copy/paste the code) and click the `TRY IT` button.

```
print("Hello world")
```

The reason you were able to see the words appear is because of the `print` command. Change your code to look like this and run it again.

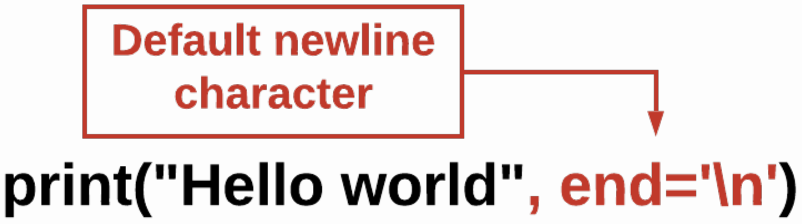
```
"Hello world"
```

# Printing

## Newline Character

The `print` command automatically adds a newline character each time you use it. This is the default behavior. The code below will not print the two words on the same line. Enter the code below (or copy/paste the code) and click the `TRY IT` button.

```
print("Hello")
print("world")
```



Newline Character

The text in red shows the newline character which is added even if you do not type it. (The newline character is what is inserted when you press “Enter” or “Return”).

## Removing the Newline Character

Add `, end=''` (two quotes with nothing between them) to your `print` command. This overrides the default newline character.

```
print("Hello ", end='')
print("world")
```

challenge

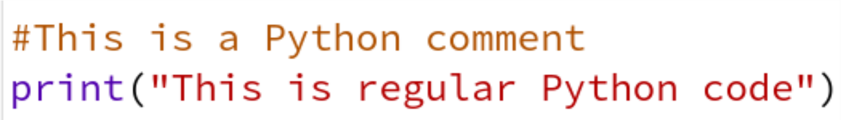
## What happens if you:

- Use double quotes instead of single quotes with `end=''`
- Use `end='t'` in the `print` command
- Use `end='!'` in the `print` command

# Comments

## Comments

You may have wondered why a couple of lines of code are a different color (in the below example, light brown, but it depends on the Theme you have picked):



```
#This is a Python comment
print("This is regular Python code")
```

.guides/img/comments

In Python, to write notes in code without effecting it’s function we can use `#` to make a **comment**.

Comments can also be used to help you fix your code. You can “comment out” lines of code that are not working or you suspect are causing problems.

challenge

### What happens if you:

- Change `print("Red")` to `prnt("Red")`?
- Comment out the line of code with the typo?

## Comment Blocks

To make a multi-line comment you can either combine the single line characters `#` or wrap the set of lines in triple quotes (`'''`).

```
'''
This is a multi-line comment
You can then easily end the comment with a triple quote (see below)
'''

#You can also do a multi-line comment
#like this!

print("Notice code that runs is not the same color as single-line comm

print("This feature is called syntax highlighting");
print("It is a common feature of IDEs");
```

The syntax highlighting is different for comments with `#`and comments with `'''`. That is because the triple quotation marks are also used for multi-line strings (see the Strings lesson). When a multi-line string is by itself (no `print` statement), then Python treats it as multiline comment.

### ▼ What is an IDE?

> An integrated development environment, or IDE, is a computer program that makes it easier to write other computer programs. They are used by computer programmers to edit source code, and can be easier to use than other text editors for new programmers. They can have compilers so programmers don't have to open other programs to compile the source code. They also often have syntax highlighting. … It also may have predictive coding that can finish lines with syntax such as brackets or semicolons and can suggest variables to be used. It also may have debuggers that can step through lines, take breaks and inspect variables.
>
> **Source: Simple Wikipedia**