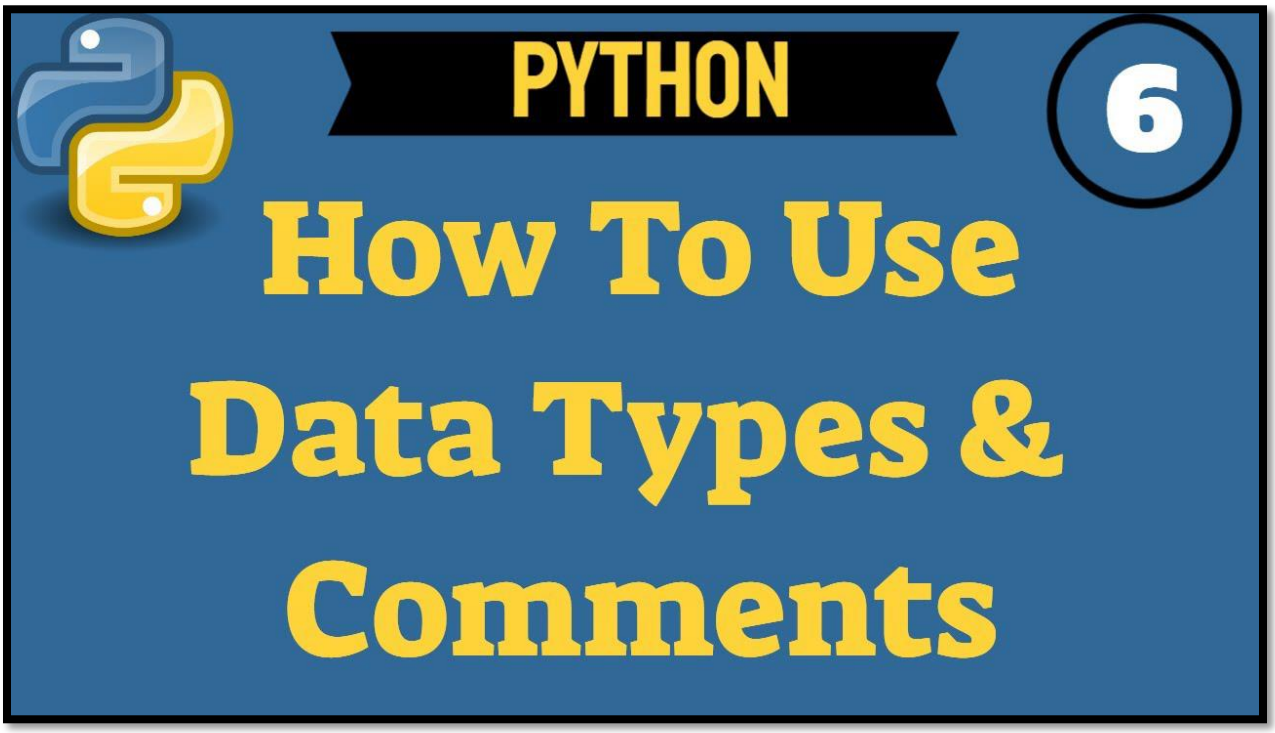
Rex Jones II

Python Comments & Data Types



Video https://www.youtube.com/watch?v=PZ6gvza8kmo&list=PLfp-cJ6BH8u8iMtCoea_mWkGesJGQG-vZ&index=6

In this session, we are going to discuss Comments and Data Types. A comment allows us to write notes in our program. A Data Type has a category for every value. For Python, we do not have to declare the Data Type for a variable.

Comments

Let's start with comments. So far, in our program we have written Python code. We use comments to explain what our code should do and how we make our code work. A hash tag (#) is an indicator that our program has a comment. There are 3 Ways To Write A Comment (Single-Line, Multi-Line, & Docstring).

```
# 3 Ways To Write A Comment (Single-Line, Multi-Line, & Docstring)
```

The interpreter will ignore everything following the hash tag. This is a single-line comment. We use single-line comments to explain a code statement. A multi-line comment has multiple # # hash tags. Most of the times, we use multi-line comments to help our fellow programmers read the code and explain our code in more detail. The docstrings comment is different. We write the double quotes 6 times and add a note after the 3rd double quote. A docstrings comment is used to describe a module,

function, or object. It's common practice to start a Python file with a few lines of comments to describe the purpose of the file.

Data Types

When it comes to Data Types, #Python Has 3 Common Data Types. #(integer, float, string).

```
# Python Has 3 Common Data Types
# integer, float, string
```

Integer is also known as int, float is also known as a Floating-point number, and string can be written as str. An integer is a value with a whole number. Floats are numbers with a decimal point. Strings are a series of characters surrounded by quotes. They are surrounded by quotes so Python understand where the string start and stop. """ Strings Are Written Inside Double Quotes Or Single Quotes """

```
""" Strings Are Written Inside Double Quotes Or Single Quotes """
```

Since Python is a dynamically type language, we can bypass writing the data type. However, if we want to be explicit, we can write the variable and data type. For example, the normal way of writing a variable is age = 34. That's the convention. This is an int data type. But we can add a colon : after age then write int. Recall int data types are whole numbers so it can be written as temp = -3. Negative values are also whole numbers. How about a floating-point number? price: float = 56.78. The conventional syntax is number = -12.34. A string is written the same way but only use 3 letters full_name: str = "Rex Jones II". This is double quotes and single quotes are written like nick_name = 'James Jones'. It's true that a string is a series of characters. That means it can be any letters, numbers, and special characters. So we can also write a string like cost_of_product = "74.99". This is also valid. Also when it comes to a string, we can write blank_string which can be written as = "". These are the 3 common data types: integers, floats, and strings. However, we have more data types like Boolean Types, Sequence Types, Mapping Type, Set Types, and Binary Types. Boolean is known as bool and only contains True and False values. isMale: bool = True. The True and False values must begin with a capital letter. Let me do one more like isDisplayed = False.

Rex Jones II

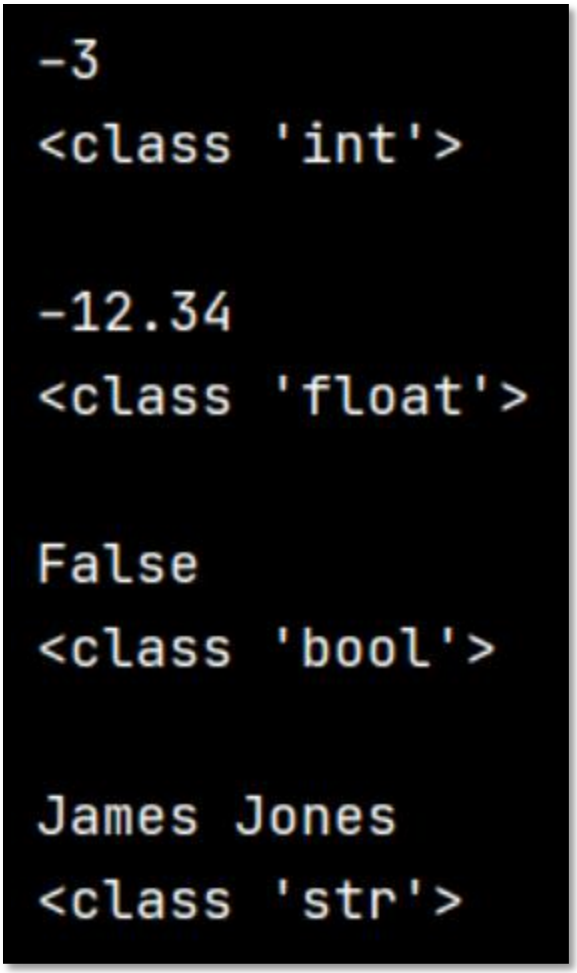
```
age: int = 34
temp = -3
price: float = 56.78
number = -12.34
full_name: str = "Rex Jones II"
nick_name = 'James Jones'
cost_of_product = "74.99"
blank_string = ""
isMale: bool = True
isDisplayed = False
```

Print some of these values.

print(temp) / print(type(temp)) / print()
print(number)/ print(type(number)) / print()
print(isDisplayed) / print(type(isDisplayed)) /print()
print(nick_name) / print(type(nick_name))

Let's run. We see the console shows each value with their data type. -3 is an int, 12.34 is a float, False is a boolean, and James Jones is a string. That's it for Comments and Data Types.

Rex
Jones II



Contact Info

- ✔ Email Rex.Jones@Test4Success.org
- ✔ YouTube https://www.youtube.com/c/RexJonesII/videos
- ✔ Facebook https://facebook.com/JonesRexII
- ✔ Twitter https://twitter.com/RexJonesII
- ✔ GitHub https://github.com/RexJonesII/Free-Videos
- ✔ LinkedIn https://www.linkedin.com/in/rexjones34/