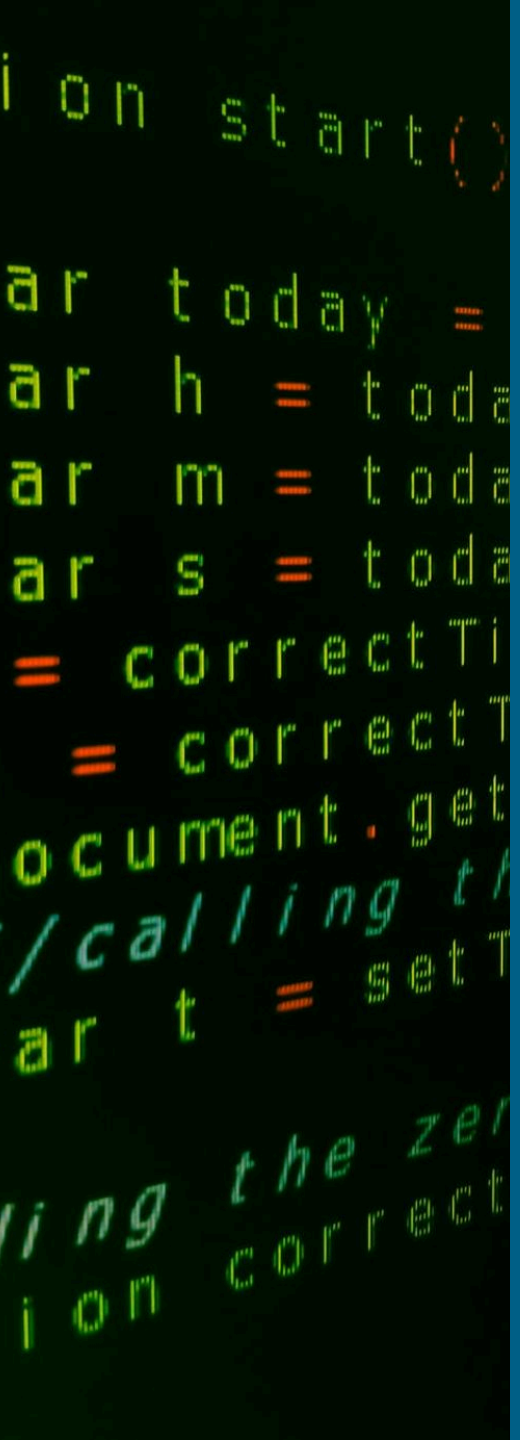
JANUARY 15TH, 2025

Basic Data Types

CE 311K - L02



Data Types

Data Types specify the kind of data stored in variables

Typing impacts how we work with variables when programming

Python uses **Dynamic Typing**, meaning we don't have to specify the data type

This can sometime lead to **TypeErrors**

What are some data types we might need?

Variables and Assignment

A **variable** is a named storage location in a program that holds a value used during the execution of the program

You assign values to variables using the **equal sign**

```
num = 42
word = "python"
```

Variables can be overwritten or **reassigned**

```
name = "Hagen Fritz"
print(name) # Hagen Fritz
name = "Fritz, Hagen" # overwrite name with a new value
print(name) # Fritz, Hagen
```

Integers

Integers are whole numbers: negative, zero, and positive

```
i = 0
y = -10
age = 31
```

What are some examples of variables that we would want to define as integers?

Floats

Floats are numbers with decimal points

```
pi = 3.14159
temperature = -7.3
```

A Python float type does not follow the typical definition

In programming, what is a **Double**?

Strings

Strings are sequence of characters: letters, numbers, and/or symbols

```
name = "Fritz, Hagen"
email = "HagenFritz@utexas.edu"
eid = "hef372"
```

You can define strings with single or double quotes

```
phrase = "It's up to you!"
```

More on Strings

Escape Characters are special combinations of characters that always start with a backslash

```
sentence = "He said,\t\"Whoa! Did you see that?\"\n"
```

You can define **Multiline Strings** by enclosing the text in three quotes

```
multiliner = """This is a
multiline string.
It spans multiple lines."""
```

Booleans

Booleans represent truth values i.e. yes or no

```
flag = True
active = False
```

Booleans are essential in controlling program flow

Affirmative: yes, true, 1

Opposing: no, false, 0

Checking a Variable's Type

Python has a **Built-In** function called *type()*

```
print(type(age))  # Should print <class 'int'>
print(type(pi))   # Should print <class 'float'>
print(type(name))  # Should print <class 'str'>
print(type(active))  # Should print <class 'bool'>
```

Use *type()* to verify a variable's type when debugging

Especially if you come across a **TypeError**

Casting

Casting is the term used to describe forcefully converting a variable to another type

You will use other built-in functions specific to the variable type you want to cast

```
age = 25 # age is in int: 25
age_str = str(age) # string representation of age:
"25"
age_float = float(age) # float representation of age:
25.0
```

Cautions About Casting

Truncation occurs when you cast a float as an integer

Truncation is **NOT** the same as rounding

```
height = 5.9167
height_int = int(height) # truncates to 5
```

A **ValueError** will occur if you cast incorrectly

```
num_str = "6.023x10^23"
num = float(num_str) # will raise a ValueError
```

Basic Variable Types

Four major, basic variable types

Integer (int)

Float (float)

String (str)

Boolean (bool)

Use casting to convert data types

int()

float()

str()

bool()

Check a variable's data type with built-in function *type()*