Texas Global Introduction to Python

Aashish Gottipati

Lecture 2: Variables and Expressions

Spring 2025

Don’t Stress, Enjoy your Time at UT!

Exercise: HelloWorld.py

Python Reference Sheets

- https://www.pythoncheatsheet.org/cheatsheet/basics
- https://quickref.me/python.html
- Feel free to look things up as needed

Agenda

- Recap
- Variables
- Types
- Arithmetic operators

Recap

- Computers have hardware and memory for calculation
- Only do what we tell specify. Input / output machines
- Programming language helps us instruct computer
  - Syntax – Rules that define what's valid
  - Primitives – Basic building blocks for operations
  - Algorithm – Instructions for computer

Python as a calculator

- Let us calculate the distance between Edinburgh and London in km

```
403 * 1.60934
```

```
648.56402
```

Variables

- Great calculator but how can we make it store values?
- Do this by defining variables
- Can later be called by the variable name
- Variable names are case sensitive and unique

```
distanceToLondonMiles = 403
mileToKm = 1.60934
distanceToLondonKm = distanceToLondonMiles * mileToKm
distanceToLondonKm
```

```
648.56402
```

We can now reuse the variable mileToKm in the next block without having to define it again!

```
marathonDistanceMiles = 26.219
marathonDistanceKm = marathonDistanceMiles * mileToKm
print(marathonDistanceKm)
```

```
42.19528546
```

Types

Variables actually have a type, which defines the way it is stored.

The basic types are:

Type	Declaration	Example	Usage
Integer	int	`x = 124`	Numbers without decimal point
Float	float	`x = 124.56`	Numbers with decimcal point
String	str	`x = "Hello world"`	Used for text
Boolean	bool	`x = True` or `x = False`	Used for conditional statements
NoneType	None	`x = None`	Whenever you want an empty variable

Why should we care?

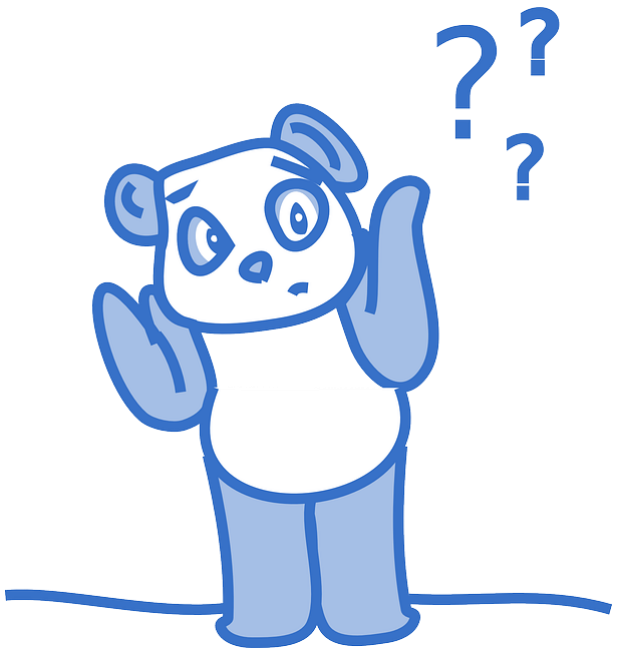

Image by Clker-Free-Vector-Images on Pixabay

```
In [4]: x = 10    # This is an integer
        y = "20"  # This is a string
        x + y
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-4-f1463b8b4c2e> in <module>()
      1 x = 10    # This is an integer
      2 y = "20"  # This is a string
----> 3 x + y

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Important lesson to remember!

We can't do arithmetic operations on variables of different types. Therefore make sure that you are always aware of your variables types!

You can find the type of a variable using **type()**. For example type **type(x)**.

Casting types

Luckily Python offers us a way of converting variables to different types!

Casting – the operation of converting a variable to a different type

```
x = 10    # This is an integer
y = "20"  # This is a string
x + int(y)
```

```
30
```

Similar methods exist for other data types: **int()**, **float()**, **str()**

Quick quiz

```
x = "10"
y = "20"
x + y
```

What will be the result?

```
'1020'
```

Arithmetic operations

Similar to actual Mathematics.

Order of precedence is the same as in Mathematics.

We can also use parenthesis ()

Symbol	Task Performed	Example	Result
+	Addition	4 + 3	7
-	Subtraction	4 - 3	1
/	Division	7 / 2	3.5
%	Mod	7 % 2	1
*	Multiplication	4 * 3	12
//	Floor division	7 // 2	3
**	Power of	7 ** 2	49

Order precedence example

```
16 ** 2 / 4
```

```
64.0
```

Quick quiz

```
4 + 3 ** 2
```

13

vs

```
(4 + 3) ** 2
```

49

Exercise: Echo

- Task: Create a program that accepts user input and echoes the result back to the user. You will need to create variable that stores user input and then print out the variable.

- Example:
- >>> Hello World!
- >>> Hello World!

- HINT: You can retrieve user input with the **input()** command.