

# Programming Universals

Alark Joshi

# Program basics

- Input
- Processing
- Output
  - What do we know about output so far?

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

- Statements in a program execute in the order that they appear (top to bottom)

# Display output with `print`

- **Function:** piece of prewritten code that performs an operation
- **print** function: displays output on the screen
- **Argument:** data given to a function
- **Example:** data that is printed to screen  

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

# Why do we need input?

- What kind of input may we want from a user?
- What should we do with that input?
- Where should we store that input?
- How should we access that data?
  - A variable

# Input - Variables

- Variable: name that represents a value stored in the computer memory
  - Used to access and manipulate data stored in memory
  - A variable references the value it represents

# Input - Variables

```
name = input("Please enter your name:")
```

```
city = input("Favorite city:")
```

```
team = input("Favorite team:")
```

# Variables – Assignment statement

- Assignment statement: used to create a variable and make it reference data
- General format is variable = expression
  - Example: age = 29
- Assignment operator: the equal sign (=)

# Numerical input

- `temp = input("Current temperature:")`
- Mathematical operations
  - Addition, subtraction, multiplication, division
- `newTemp = temp * 2`
- `tip = total + total * 0.15`



# Reading numerical input

- `input()` function **always** returns a string
- Built-in functions convert between data types
  - `int(item)` converts item to an int
  - `float(item)` converts item to a float

```
temp = int(input("Current  
temperature:"))  
newTemp = temp * 2
```

# Python operator precedence

1. Operations enclosed in parentheses
  - Forces operations to be performed before others
2. Exponentiation (\*\*)
3. Multiplication (\*), division (/ and //), and remainder (%)
4. Addition (+) and subtraction (-)

# Precedence

- Higher precedence performed first
  - Same precedence operators execute from left to right

# Mixed-Type Expressions

- Data type resulting from math operation depends on data types of operands
- Two `int` values: result is an `int`
- Two `float` values: result is a `float`
- `int` and `float`: `int` temporarily converted to `float`, result of the operation is a `float`
- Type conversion of `float` to `int` **causes truncation** of fractional part

# Comments

- Notes of explanation within a program
- Ignored by Python interpreter
- Intended for a person reading the program / code

# Comments

- Begin with a # character
- End-line comment: appears at the end of a line of code
- Typically explains the purpose of that line

```
# Ask the user for their bank balance  
balance = int(input("Enter your bank  
balance:"))
```