

# CS 1340:Fall 2020:Lecture 02

## Intro to Python for CS and Data Science

---

Mark Fontenot, PhD

Southern Methodist University

1

### Reminders

- Slack
- Zybooks (hopefully you did the assignment due before class today?)
- Anaconda?

2

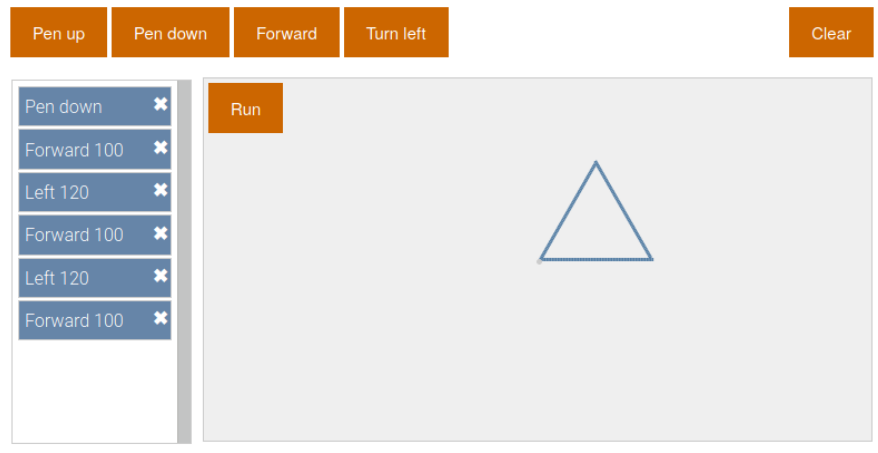
## Getting Started

---

All computer programs have ...

1. Input -
2. Processing -
3. Output -

## Drawing with the Turtle



1. What's the input?
2. What's the processing?
3. What's the output?

4

## Challenge on your own time ..

Can you draw 'SMU' (block letters of course) with the Turtle?

5

## Humans vs. Computers

- Humans don't understand long strings of 1's and 0's
- Computers don't understand

```
print('Hello 1340')
```

So what do we do???

6

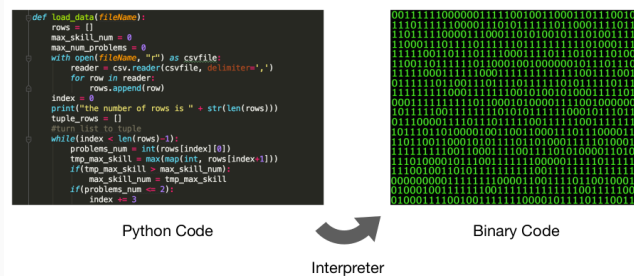
## General *Algo* for Programming

1. You write source code in a code editor or IDE (Integrated Development Environment)
2. Save it with a file extension of .py. Example: project01.py
3. Use the Python interpreter to execute the source code.
  - `python project01.py`
  - OR could be `python3 project01.py`
4. Back to Step 1 to add more code.

7

## Converting Python Code to Machine Code

- Python is an **interpreted** language
  - as your program is running, the Python Interpreter / Runtime is converting source code to machine code one line by line.
  - The alternative is a **compiled** language which converts all the source code to machine code before you run your program.



8

## Interactive Python

- `python somescript.py` - runs the code inside `somescript.py`
- `python` - Starts the **interactive interpreter**
  - each line of code is interpreted right after you type it.

A screenshot of a terminal window titled "python". The prompt is `>`. The user has entered `python`, which has started the Python 3.7.7 interpreter. The prompt is now `>>>`. The user has entered several lines of code: `print('Hello World!')`, `name = 'Mark'`, `print('Hello', name)`, `day = 27`, `print(day)`, `day = day + 1`, and `print(day)`. The output shows "Hello World!", "Hello Mark", and "27". The prompt is now `>>>` with a cursor.

9

## Elements of Python Code

- statement -
- variable -
- expression -
- assignment -

10

## Some Python Code

```
def advance_cars():
    """Calculate new positions of the cars"""
    global car1_speed, car1_location
    global car2_speed, car2_location
    car1_speed += car1_acceleration
    car1_speed = car1_top_speed if car1_speed > car1_top_speed else car1_speed
    car1_location += car1_speed

    car2_speed += car2_acceleration
    car2_speed = car2_top_speed if car2_speed > car2_top_speed else car2_speed
    car2_location += car2_speed
```

11

## Basic Output

```
print(...)
```

- Notice:
  - printed in mono-spaced font
  - ... means other stuff will be put there
  - () indicate a method or function call

```
print('Hello')
```

```
print('World')
```

```
print('Hello World')
```

'Hello', 'World' are called **string literals**.

12

## print(...)

```
name = 'Mark'
```

```
print(name)
```

- You can print **string literals** OR values contained in variables.
- What is the variable in this example?

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

```
print('World', end='')
```

- What will this print?

13

## Can you do it?

*Use print statements to draw a diamond shape.*

*Use print statements to draw a heart shape.*