**Outline**

Access the Python Development environment and continue the tutorial to gain an additional exposure to the Python programming language. Begin to develop an familiarity with intermediate programming concepts.

**Objectives**

* Use correct terminology to describe programming concepts;
* Describe the types of data that computers can process and store (e.g., numbers, text);
* Explain the difference between constants and variables used in programming;
* Use variables, expressions, and assignment statements to store and manipulate numbers and text in a program

**Materials**

* Python3 Development Environment at: //repl.it/
* Python Tutorial at: <http://www.letslearnpython.com/learn/>

**Accessing the Tutorial**

Accessing the Tutorial

* Go to: <http://www.letslearnpython.com/learn/>
* Read up to “Lesson 12: Input”

**Level 1: Input & Output**

1. Read through “Lesson 12: Input – What Is Input?” and “Lesson 12: Input – Example” and “Lesson 12: Input – Shortcut”.
2. Type the following code into the white area of the IDE and run the program. Explain what you see in the black area of the IDE.

print("Type your name:")

name = input()

print("Hi", name, "how are you?")

It shows an error in the black area, as shown below:

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

Type your name:

Faheem

Traceback (most recent call last):

File "main.py", line 2, in <module>

name = input()

File "<string>", line 1, in <module>

NameError: name 'Faheem' is not defined

1. Create a short program that reads numerical input from the console and does the following:
   1. Uses the input() function to read a numerical value from the console.
   2. Calculates the square root of the number
   3. Prints the result to the console output
   4. Provides appropriate prompt and message strings to go with the input and output.
   5. Provide your complete program below.

**Level 2: Tic-Tac-Toe Game**

1. Write a Python program to play a game of Toc-Tac-Toe. (You may modify a program that you found on-line to meet the expectations of this module.)
   1. The program may be either player v. computer or player 1 v. player 2.
   2. The program does not need to determine a winner
   3. The program just needs to keep track of moves and spaces in the game board
2. Provide a complete listing of your program.
   1. Your listing **MUST** include line numbers .

1. pX = "X"

2. pO = "O"

3. pN = " "

4.

5. board = [pN,pN,pN,

6. pN,pN,pN,

7. pN,pN,pN]

8.

9. def printBoard() :

10. print()

11. print(board[0],"|",board[1],"|",board[2])

12. print("----------")

13. print(board[3],"|",board[4],"|",board[5])

14. print("----------")

15. print(board[6],"|",board[7],"|",board[8])

16. printBoard

17.

18.

19.

20. while True:

21. move = int(input("player X, Make a move:"))

22. board[move] = pX

23. printBoard

24.

25. move = int(input("player O, make a move:"))

26. board[move] = pO

27. printBoard()

28.

29. stop = input("Do you want to stop? Y/N")

30. if stop == "Y":

31. break

1. Explain how your program keeps track of the game board.   
   (Provide specific code references by line number.)
   1. What python types and data structures are used?

It is a boolean and list.

* 1. How are moves by player X and player O recorded?

By the players icon, on a certain place on the board.

* 1. How are free spaces recorded?

Every square has its own number and the square that still have numbers are free.

1. Explain how moves and commands are input from the console.  
   (Provide specific code references by line number.)
   1. How does the player tell the program about the move location (row, column)?

Every little box on the board has its own number starting from 0 to 8.

* 1. How does the program verify that the move location is valid?

It does not. I didn’t add that because you should know and also see where you’ve played.

* 1. How does the program verify that the space is free?

It would show on the board the no one has played in the place empty.

* 1. What does the program do if there is something wrong with the move?

For example, let's say you play over a spot that was already taken then the newer icon would overwrite the previous icon. I’ve never had something wrong with the moves. Unless you over write on a space that was already played on than the newer icon would overwrite.

1. Explain how the program keeps track of gameplay.  
   (Provide specific code references by line number.)
   1. How does the program switch between player X and player O moves?

You would have to type “X” if your that player or type “O” if your that player.

* 1. How does the program keep asking for moves?

player X, Make a move:

player O, make a move:

* 1. How does the program decide when to stop asking for moves?

It would ask you, and if one of the players win it would say: Do you want to stop? Y/N. If you win than you press “y” and then the game will stop.

**Level 3: Basic Enhancements**

1. Explain, in plain words, a strategy for determining if player “x” or player “O” has won the game after a move is made.

If you have a row with 3 “X” or “O”, then after every time you and your opponent it will ask you want to continue to play. Of you put “N”, the player with a row of “X” or “O” then that player wins.

1. Provide a function called “checkWinForX” that returns the Boolean value of “True” if player “x” won the game.
2. Modify your program to check and print a message, and stop the game of player “x” or player “O” wins the game.
3. Demonstrate your enhanced game to Mr. Nestor for credit for this level.

**Level 4: AI Enhancements**

1. Explain, in plain words, a strategy for suggesting the best move for player “x” or player “O” to make when it is their turn to move.

If you were to play, the person who plays first is going to be “X”, so my strategy for them is to always play in 3 corners then you’ll have one way to win if “O” blocks one way.

1. Create a function to implement your strategy for suggesting the best move.
2. Modify your program to print a suggested move when it is each player’s turn to move.
3. Demonstrate your AI enhanced game to Mr. Nestor for credit for this level.