

Q&A

```
>>> 3 + 4 and 3 - 3
```

```
0
```

```
>>> 3 + (4 and 3) - 3
```

```
3
```

```
>>> (3 + 4) and (3 - 3)
```

```
0
```



Q: (4 and 3) is 3. Why?

A: Logical Expression의 Value는 마지막 참조값

(4 and 3)의 결과를 알려면 4를 보고 3을 봐야함 → 3

(0 and 3)의 결과를 알려면 0만 보면 끝 → 0

(True or 3)의 결과는 True를 보면 끝 → True

(False or 10)의 결과는 10까지 봐야 함 → 10

오늘의 강의 목표

- Console 개념에 대한 이해
- Console Input 방법 이해
 - Blocking Input과 Non-blocking Input의 구별
- Console Output 방법 이해

Console

Long time ago, there was a console.



http://en.wikipedia.org/wiki/Computer_terminal

No Graphics.
Only Characters.

Input



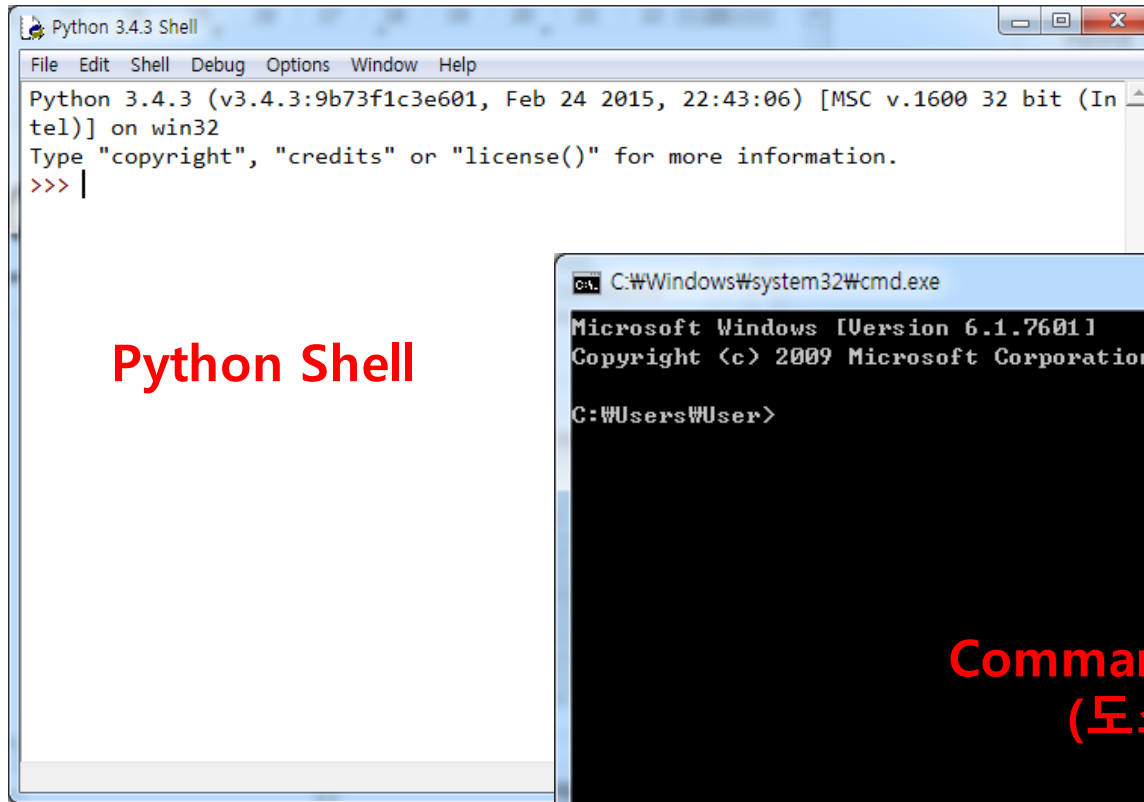
Output



**System
User**

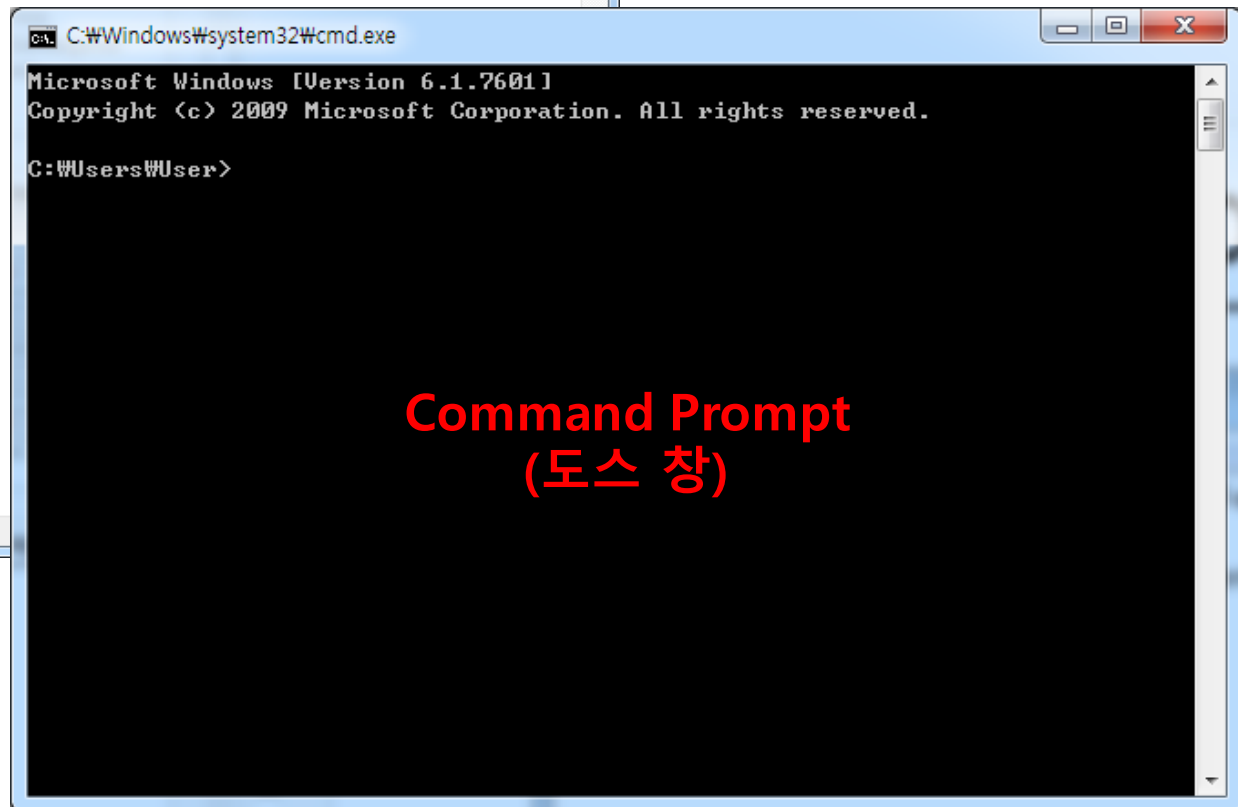
Console

Today's console

A screenshot of a Python 3.4.3 Shell window. The title bar reads "Python 3.4.3 Shell". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The text area shows the following: "Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.1600 32 bit (Intel)] on win32", "Type \"copyright\", \"credits\" or \"license()\" for more information.", and the prompt ">>> |".

```
Python 3.4.3 Shell
File Edit Shell Debug Options Window Help
Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Python Shell

A screenshot of a Windows Command Prompt window. The title bar reads "C:\Windows\system32\cmd.exe". The text area shows the following: "Microsoft Windows [Version 6.1.7601]", "Copyright (c) 2009 Microsoft Corporation. All rights reserved.", and the prompt "C:\Users\User>".

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\User>
```

Command Prompt
(도스 창)

Console Input

키보드 입력을 변수로 받기

방법 1) Python Shell을 이용

```
>>> name = '김국민'  
>>> print(name + ' 만세')  
김국민 만세
```

한계점: 독립 프로그램의 경우는?

방법 2) input()을 이용

```
name = input()  
print(name + ' 만세')
```

Python Script로 만들어서 실행 가능

test.py로 저장하여 F5
로 실행시켜 봅시다

input()

```
>>> name = input() # 키보드 입력을 변수에 입력
John
>>> print(name)
John
>>> type(name)
<class 'str'> # 생성된 변수는 str 타입
```

```
>>> age = input()
20
>>> print(age)
20
>>> type(age)
<class 'str'> # 숫자만 입력했지만 여전히 str 타입
```

input()은 키보드 입력을 str 타입으로 변환한다

input(*prompt*)

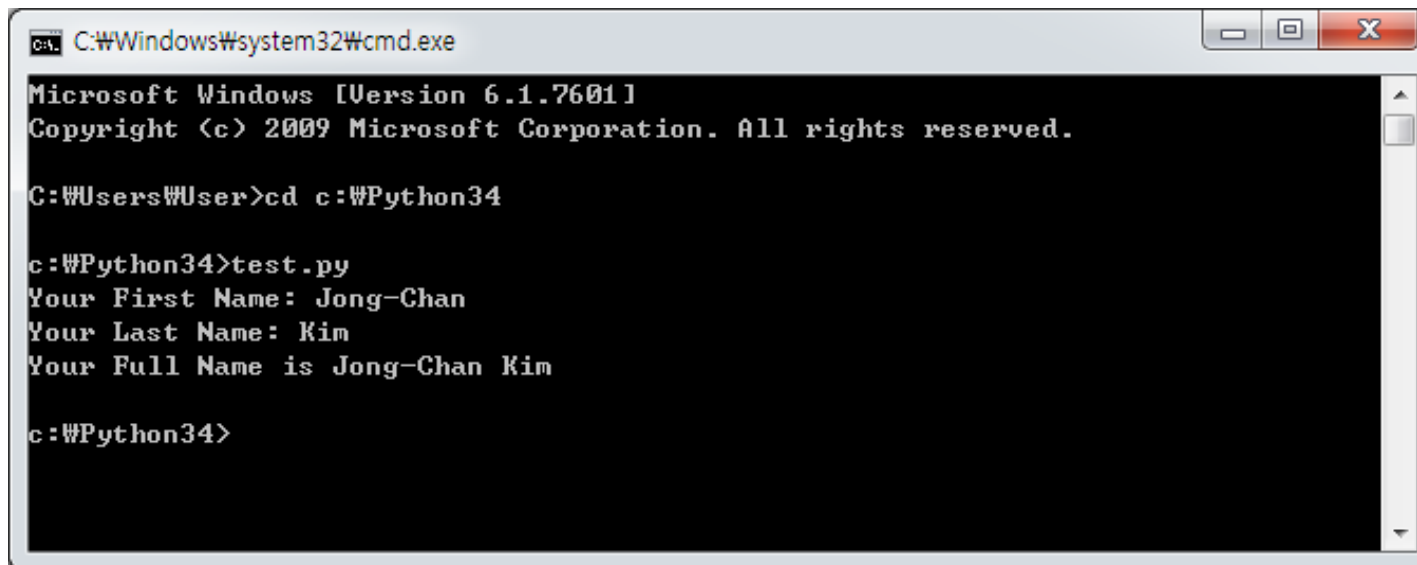
```
first_name = input("First name: ")  
last_name = input("Last name: ")  
print("Full name is " + first_name + " " + last_name)
```

Prompt String

test.py로 저장하여 F5
로 실행시켜 봅시다

도스 창에서 실행

cd 명령으로 py 파일이 있는 폴더로 이동 후 실행



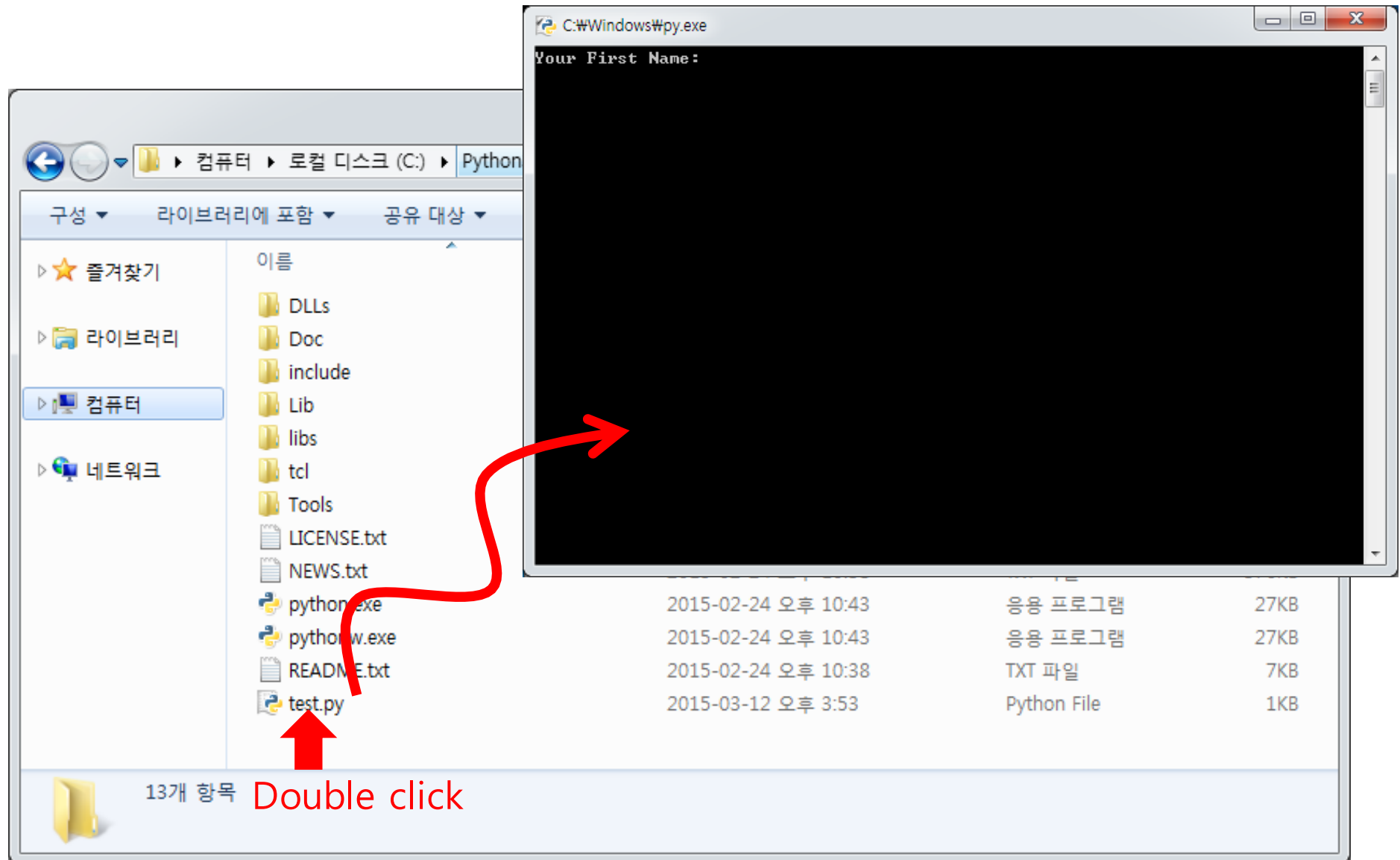
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\User>cd c:\Python34

c:\Python34>test.py
Your First Name: Jong-Chan
Your Last Name: Kim
Your Full Name is Jong-Chan Kim

c:\Python34>
```


윈도우에서 실행



Input Type Conversion

```
age_str = input("나이가? ")  
age_num = int(age_str)  
print("아직 " + str(age_num // 10 * 10) + "대시군요")
```

한줄로

```
age_num = int(input("나이가? "))  
print("아직 " + str(age_num // 10 * 10) + "대시군요")
```

Type Conversion Error 주의

나이가? 3학년5반이요

Traceback (most recent call last):

File "C:/Python34/test.py", line 1, in <module>

age_num = int(input("나이가? "))

ValueError: invalid literal for int() with base 10: '3학년5반이요'

Interactive Play with Turtle

```
import turtle

turtle.setup(500, 500, 0, 0)
t = turtle.Turtle()

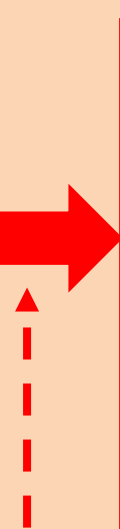
direction = input("Left(j) or Right(k): ")
if direction == "j":
    t.left(90)
if direction == "k":
    t.right(90)

distance = int(input("Distance: "))
t.forward(distance)
```

Continual Interaction

```
import turtle

turtle.setup(500, 500, 0, 0)
t = turtle.Turtle()
while True:
    direction = input("Left(j) or Right(k): ")
    if direction == "j":
        t.left(90)
    if direction == "k":
        t.right(90)
    distance = int(input("Distance: "))
    t.forward(distance)
```



Drag & Ctrl+]

주의: Python은 Indentation (들여쓰기) 자체도 문법임

Blocking Input

```
import turtle
```

```
turtle.setup(500, 500, 0, 0)
```

```
t = turtle.Turtle()
```

```
while True:
```

```
    direction = input("Left(j) or Right(k): ")
```

```
    if direction == "j":
```

```
        t.left(90)
```

```
    if direction == "k":
```

```
        t.right(90)
```

```
distance = int(input("Distance: "))
```

```
t.forward(distance)
```



Blocking

입력이 들어올 때까지 멈춤

Non-Blocking Input

```
import turtle
import msvcrt
import time

turtle.setup(500, 500, 0, 0)

t = turtle.Turtle()

while True:
    if msvcrt.kbhit():
        c = msvcrt.getch().decode(encoding = 'UTF-8')
        if c == 'j':
            t.left(10)
        if c == 'k':
            t.right(10)
        t.forward(1)
```

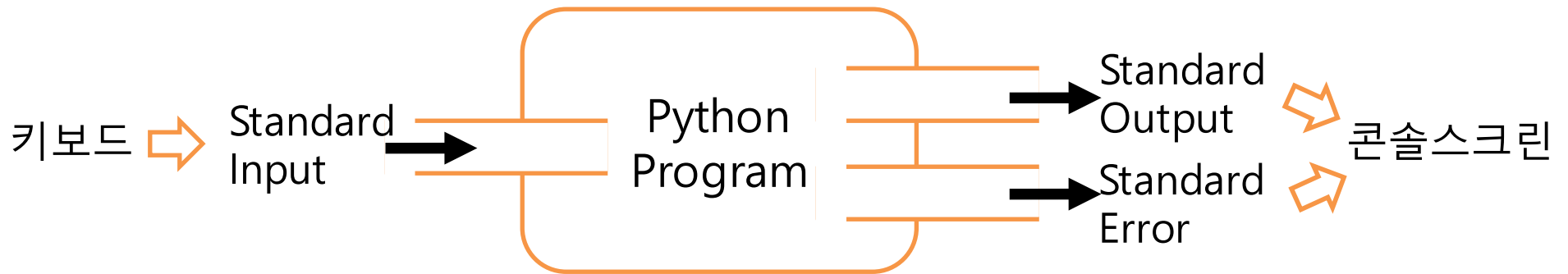
도스 창에서 실행시키고
윈도우 포커스를 도스
창에 맞춰야 함

Non-Blocking



Console Output

- 프로그램은 한 개의 Input과 두 개의 Output을 가짐
 - Standard input : 키보드 입력
 - Standard output : 정상 출력
 - Standard error : 에러 메시지 출력



Console Output

print() 함수

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



```
Hello  
World
```

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



```
Hello World
```

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



```
Hello ,World
```

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



```
Hello  
  
World
```


Escape Characters

아래와 같이 출력하려면?

```
>>>
```

```
My name is "John"
```

```
>>>
```

```
My name is 'John'
```

```
>>>
```

```
Light of the moon  
Moves west, flowers' shadows  
Creep eastward.
```

```
>>>
```

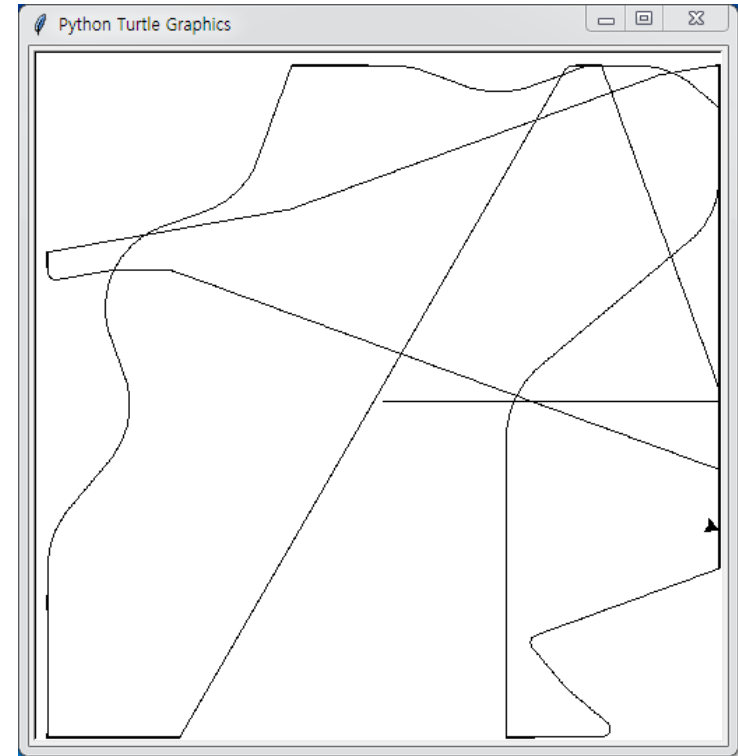
```
\n is a newline character
```

Escape Characters

Escape Character	Description
\'	단일 인용문자
\"	이중 인용문자
\\	역슬래시 문자
\a	Beep 소리 문자
\b	백스페이스
\f	From Feed (Printer-related)
\n	Line Feed (Newline)
\r	Carriage Return
\t	Tab (수평)
\v	Tab (수직) (Printer-related)
\0	Null character

Homework

첫 번째 목표
→ 거북이 탈출 방지



두 번째 목표
→ 거북이 속도 조절 (a: 가속, z: 감속)

Questions



메모

- 학생들 시행착오로 1시간으로 부족
- Input 까지 1시간 10분 걸림