Computational Design on Mobility (모빌리티 전산설계)

Prof. Ki-Yong Oh

Email: kiyongoh@hanyang.ac.kr Office: Industrial Center 309-1

Applied Dynamics and Intelligent Prognosis Laboratory

(https://adip.hanyang.ac.kr)

Contents





- Python
- Examples

1주차 복습



Ubuntu 기초 Command

- ✓ 파일 관련 command
 - ▶ wget: 현재 디렉토리에 있는 파일과 디렉토리 목록을 보여주는 명령어
 - > cd: 디렉토리를 변경하는 명령어로, 사용자가 작업하려는 디렉토리로 이동하는 명령어

```
$ cd ./ # 현재 경로로 이동
```

\$ cd ../ # 상위 경로로 이동

\$ cd ~/ # home으로 이동

\$ cd / # root으로 이동

- mkdir: 새로운 디렉토리를 생성하는 명령어
- rm: 파일 및 디렉토리를 삭제하는 명령어
- cp: 파일 및 디렉토리를 복사하는 명령어
- mv: 파일 및 디렉토리를 이동하거나 이름을 변경하는 명령어
- > find
 - : 파일 시스템에서 파일을 검색하는 데 사용
 - \$ find /path/to/search -name "filename"
 - \$ find | grep "hi"



1주차 복습



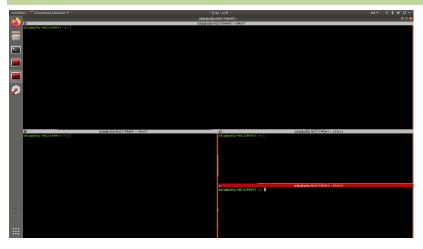
♦ Ubuntu 기본 설정

✓ 비밀번호 변경

\$ sudo passwd (\$ 계정=mobility)

- 변경 전: 12345678qwer!@#\$%^&QWER
- 변경 후: (스페이스바) ** passwd로 입력할 경우, 경고 알람 후 변경 X → sudo를 통해 권한 부여
- ✓ Terminator : 한 터미널에 다중창이 가능토록 하는 터미널

\$ sudo apt install terminator



▶ 단축키 설명

Ctrl + Shift + O	위 · 아래로 화면 나누기
Ctrl + Shift + E	좌 · 우로 화면 나누기
Ctrl + Shift + W	현재 화면 닫기
Alt + 방향키	터미널간 이동



1

1주차 복습



♦ Ubuntu 기본 설정

- \$ sudo apt update
- \$ sudo apt install language-pack-ko
- \$ sudo apt install language-pack-gnome-ko
- \$ sudo apt install ibus ibus-hangul
- ✓ IBus 설정
 - : 설치된 한글을 설정 및 변환키 변경 (Shift + Enter)

\$ ibus-setup



Python



What is C++ & Python?



: C 표준 언어를 기반으로 하는 프로그래밍 언어로 객체 지향 프로그래밍이라는 특징을 가지고 있다.



: Phython은 가독성과 높은 객체 지향 프로그래밍 언어이다.

C++ vs python

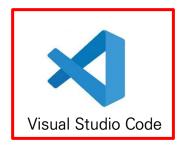
: C++은 정적으로 형식화 된 언어로 변수 선언, 변수의 데이터 형식 등이 컴파일 타임에 확인이 가능하여 오류 없이 유지 가능하다.

: 반면, Python은 정적으로 형식화 되어있지 않아 컴파일 타임에는 유형 검사가 수행되지 않아 코드의 오류 발생이 쉽다.

: 속도 측면에서 C++이 우세하지만, Python이 가독성이 높다

Integrated Development Environment (IDE)

:통합된 개발 환경으로, 개발자를 위한 종합 프로그램을 말합니다. 쉽게 말해 개발 도구 모음집







Simple Text





Python



♦ IDE 설치 및 환경 설정

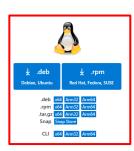
✓ Visual studio code (VS code) 설치

: https://code.visualstudio.com/download

Download Visual Studio Code

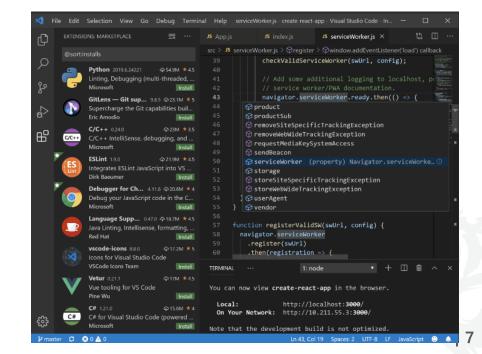
Free and built on open source. Integrated Git, debugging and extensions.















◆ IDE 설치 및 환경 설정

파일 생성 및 디렉토리 변경

\$ cd ~ # home으로 이동 \$ mkdir [폴더명] # 폴더 생성 \$ cd [폴더명] # 폴더 진입 \$ touch [파일명] # 파일 생성: 파일명은 C++으로 .cpp / Python은 .py 확장자로 파일 생성 \$ ls -al # 디렉토리 내 폴더, 파일 및 권한 확인

명령어	설명
ls	현재 디렉토리에 있는 파일 및 디렉토리의 이름을 출력한다.
ls -a	숨겨진 파일을 포함하여 출력한다.
ls -l	파일 및 디렉토리의 이름과 정보를 출력한다. (정보 : 소유자, 그룹, 수정일자, 파일크기)
ls -S	파일 및 디렉토리의 용량이 큰 순서대로 출력한다.
ls −h	파일 및 디렉토리의 용량을 단위(K,M,G)를 붙여 출력한다.
ls −R	포함된 하위 디렉토리의 내용까지 출력한다.
ls −u	파일 및 디렉토리의 접근시간을 출력한다.
ls −c	파일 및 디렉토리의 변경시간을 출력한다.
ls *문자	파일 및 디렉토리의 이름 뒷부분에 "문자" 포함되어 있는 것을 출력한다.
ls 문자*	파일 및 디렉토리의 이름 앞부분에 "문자" 포함되어 있는 것을 출력한다.







♦ C++ 기초 문법

◆ 변수 및 자료형

구분	자료형	크기	범위
기본형	void	_	_
문자형	(signed) char	1 byte	−128 ~ 127
	unsigned char	1 byte	0 ~ 255
	wchar_t	2 byte	0 ~ 65,535
정수형	bool	1 byte	0 ~ 1
	(signed) short (int)	2 byte	-32,768 ~ 32,767
	unsigned short (int)	4 byte	0 ~ 65,535
	(signed) int	4 byte	-2,147,483,648 ~ 2,147,483,647
	unsigned int	4 byte	0 ~ 4,294,967,295
	(signed) long (int)	4 byte	-2,147,483,648 ~ 2,147,483,647
	unsigned long (int)	4 byte	0 ~ 4,294,967,295
실수형	float	4 byte	3.4E-38 ~ 3.4E+38 (7digits)
	(long) double	8 byte	1.79E-308 ~ 1.79E+308 (15digits)





♦ Python 기초 문법

◆ 함수 생성 및 실행

```
def [함수명] (arg1, arg2):
d = arg1 + arg2
return d
```

add.py

```
a = 1
b = 2
c = add(a,b)
print(c)
```

\$ python add.py

→ 3



def add(a,b):

c=a+b

return c



♦ Python 기초 문법

◆ 변수 및 자료형

자료형	용도	
function	함수	
NoneType	None	
bool	True / False	
Int	정수	
float	실수	
complex	복소수	
str	문자열	
tuple	(1, 2, 3…,)	
list	[1, 2, 3,,]	
dict	{"name": "Hong-gil Dong", "age":2024}	
set	{"a", "b", "c", "d"}	





♦ Python 기초 문법

print("Hello world")
print("Hello \n Python")
print("Year is {}".format(2024))

a="hello"
b="python"
c,d="I am", "Doctor Hong"
print(a, b, c,d, type(a))

✓ 문자열

wrt = "python"
print(wrt[0], wrt[-1])
print(wrt[::-1])
wrt[0]=P
print(wrt)

print("5 / %d = %f" %(3, (5/3))) print("my name is %s" %('hkd'))

%s	%d	%f
문자열	정수	실수

✓ list / tuple

scores = [30, 40, 50] scores.append(60) print(sum(scores) / len(sum))

scores = (30, 40, 50) print(scores[0:])





♦ Python 기초 문법

Dictionary

과일	가격	개수
딸기	500	5
바나나	400	6
우유	2,500	2

dir(dict)

객체의 내장함수 확인

```
dict= {
    "딸기": [500, 5],
    "바나나": [400,6],
    "우유": [2500, 2],
}
```

```
for a in [1, 2, 3, 4]:
print(a) # 출력 1, 2, 3, 4 순서대로
```

✓ 조건문

```
a = 1
if a == 1:
    print("a is 1")
else:
    print("a is not 1: {}".format(a))
```



Examples



🔷 C++ 및 Python 예제

- ✓ Example 1: 자기소개 함수 만들기
 - ▶ [문제]: 변수를 array로 설정하여 이름, 나이와 함께 자기소개를 진행하여보자
 - ➤ [조건]: Python으로 Terminal에 출력될 수 있도록 만들기
- ✓ Example 2: 계산기 만들기
 - ▶ [문제]: Python을 이용하여 계산기를 만들어보자− 입력 값 2개와 mode (add, sub, mul, div)를 터미널에서 입력으로 받아 구현
 - ▶ [조건]: Python의 객체지향 프로그래밍 사용



Thank you