

2021년 1학기

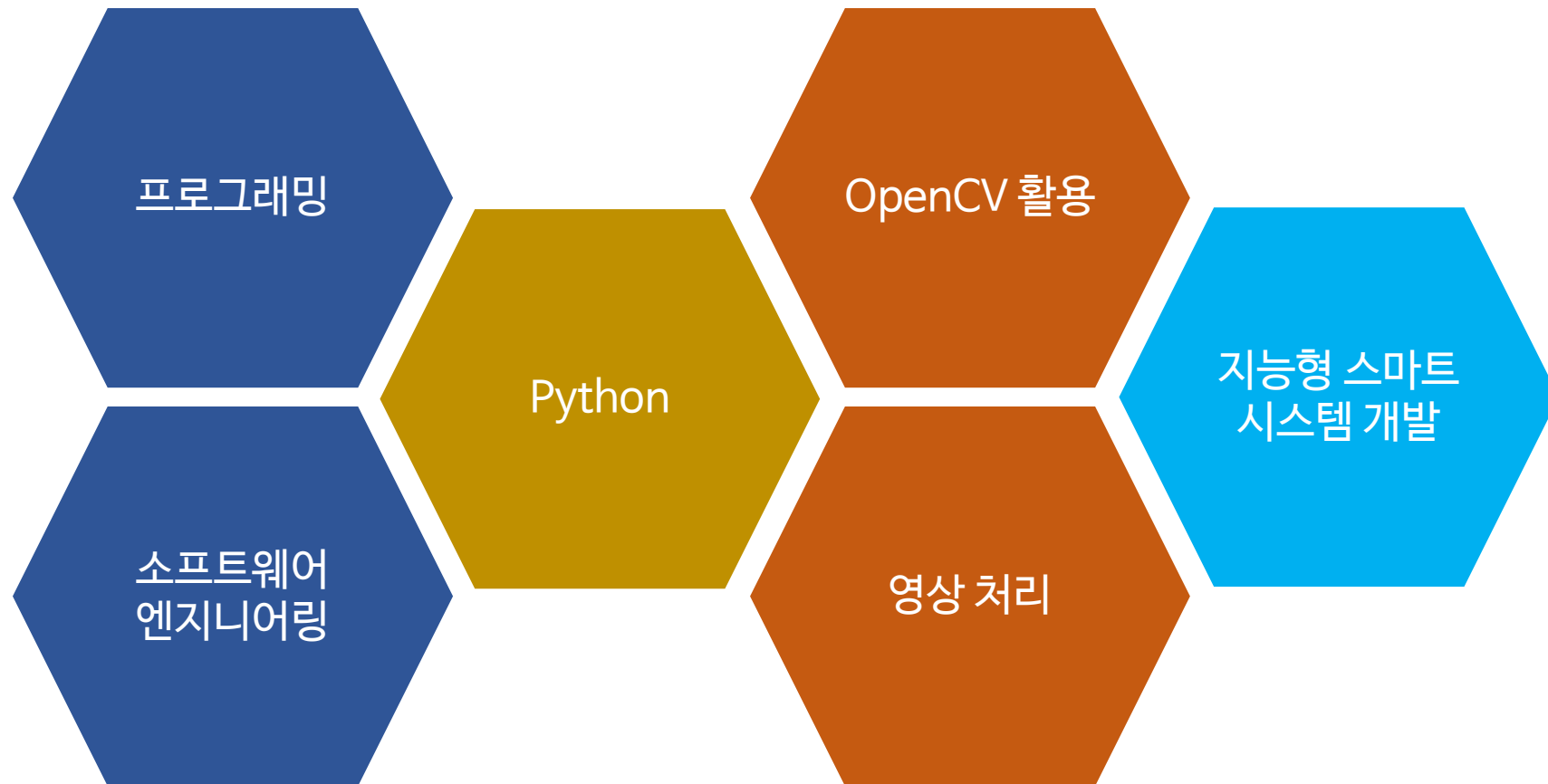
1

제어지능SW개발

openCV를 활용한 지능형시스템 개발

교수 도경민

컴퓨터전자공학과





- 담당 교수 :
 - 도경민 (kmdo@induk.ac.kr)
 - 연구실 제2공학관 519호
- 교 제
 - ppt 사용
- 참고 도서
 - python 관련 서적
 - openCV 관련 서적(python 기반)

1주	프로그래밍 기본 이해	9주	영상 디스플레이 및 색상 처리
2주	프로그래밍 기본 문법	10주	영상 변형 및 Threshold
3주	판단 및 반복 처리	11주	영상 필터링
4주	함수 및 예외 처리	12주	Edge detection
5주	영상 기본 이해 & Numpy	13주	Gradient 구현
6주	배열 및 thread 이해	14주	지능형 기능 구현
7주	다차원 배열 활용	15주	기말 시험
8주	중간시험		

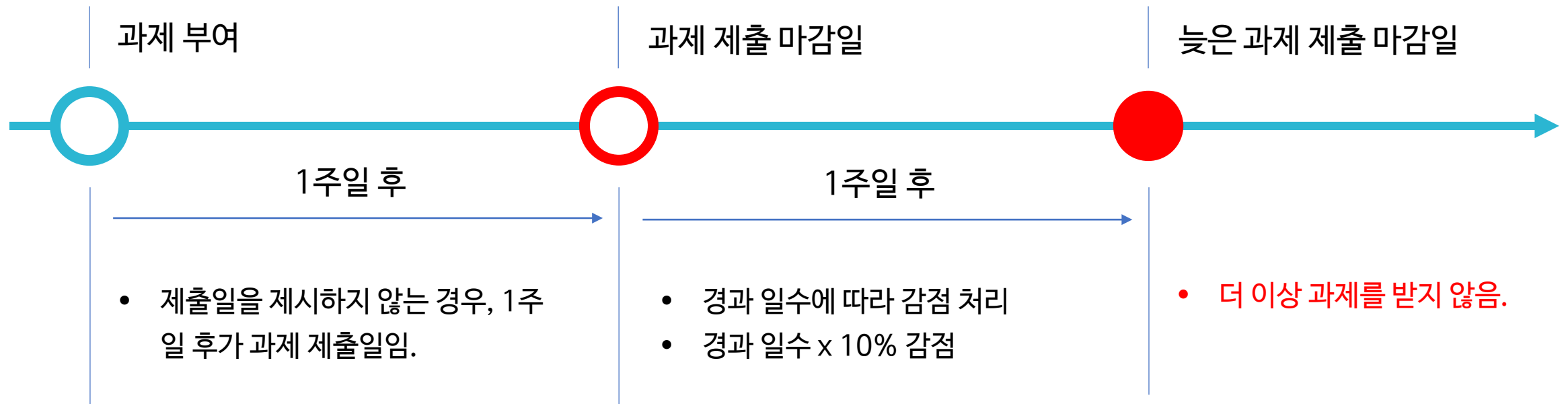
출석	Q&A	과제	중간시험	기말시험	합계
20%	10%	20%	25%	25%	100%

❖ Q&A 평가

- 매주 “자기평가”의 각 문항당 1점으로 계산
- 총 합계 80% 이상 획득시 만점 처리함.
- 80% 이하는 비율에 따라 평가함.

❖ 다음의 경우, 하나라도 해당하면 F 처리됨.

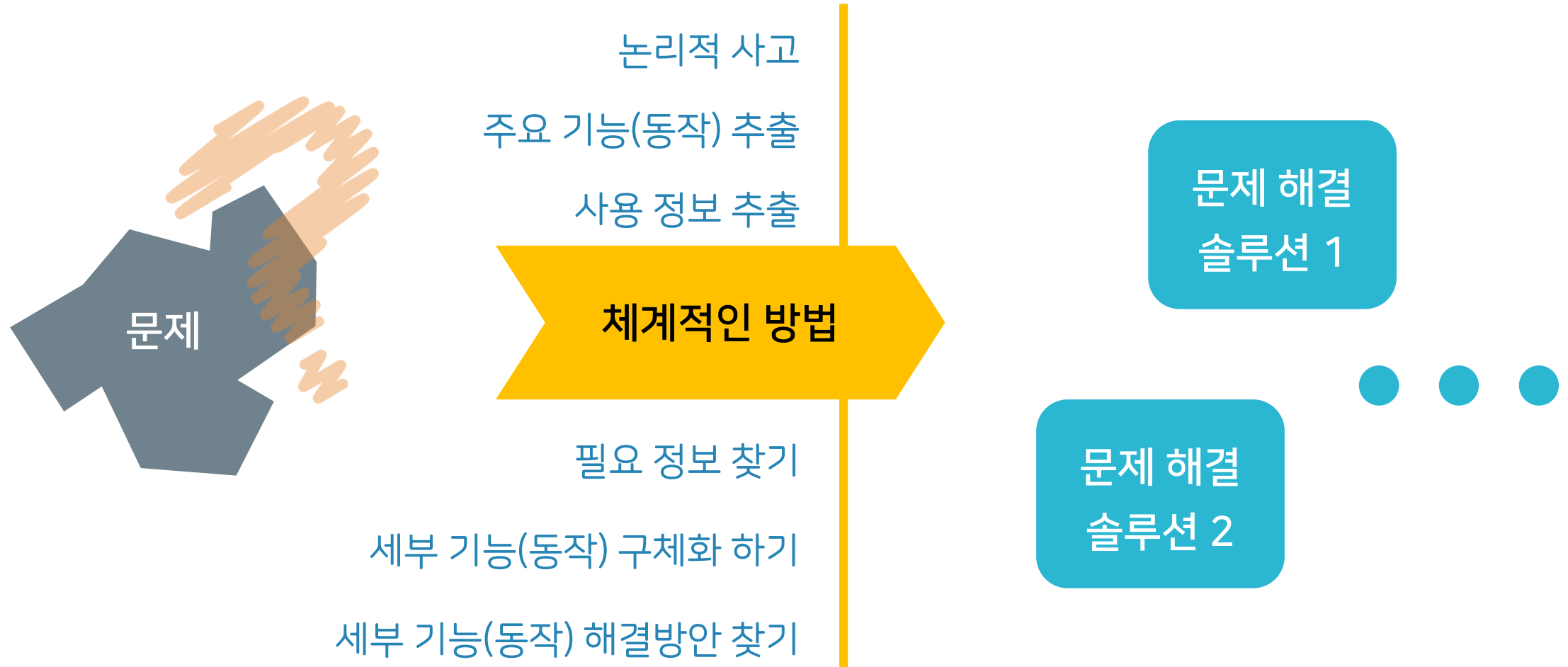
- 총수업일수 1/4 이상 결석(4일 결석, 3회 지각은 1회 결석으로 처리)
- 중간, 기말 시험 결시
- 보고서를 전혀 제출하지 않은 경우
- 보고서 전체 copy시





제어지능SW개발

Programming with Python



openCV를 활용한 지능형시스템 개발

Python & numpy

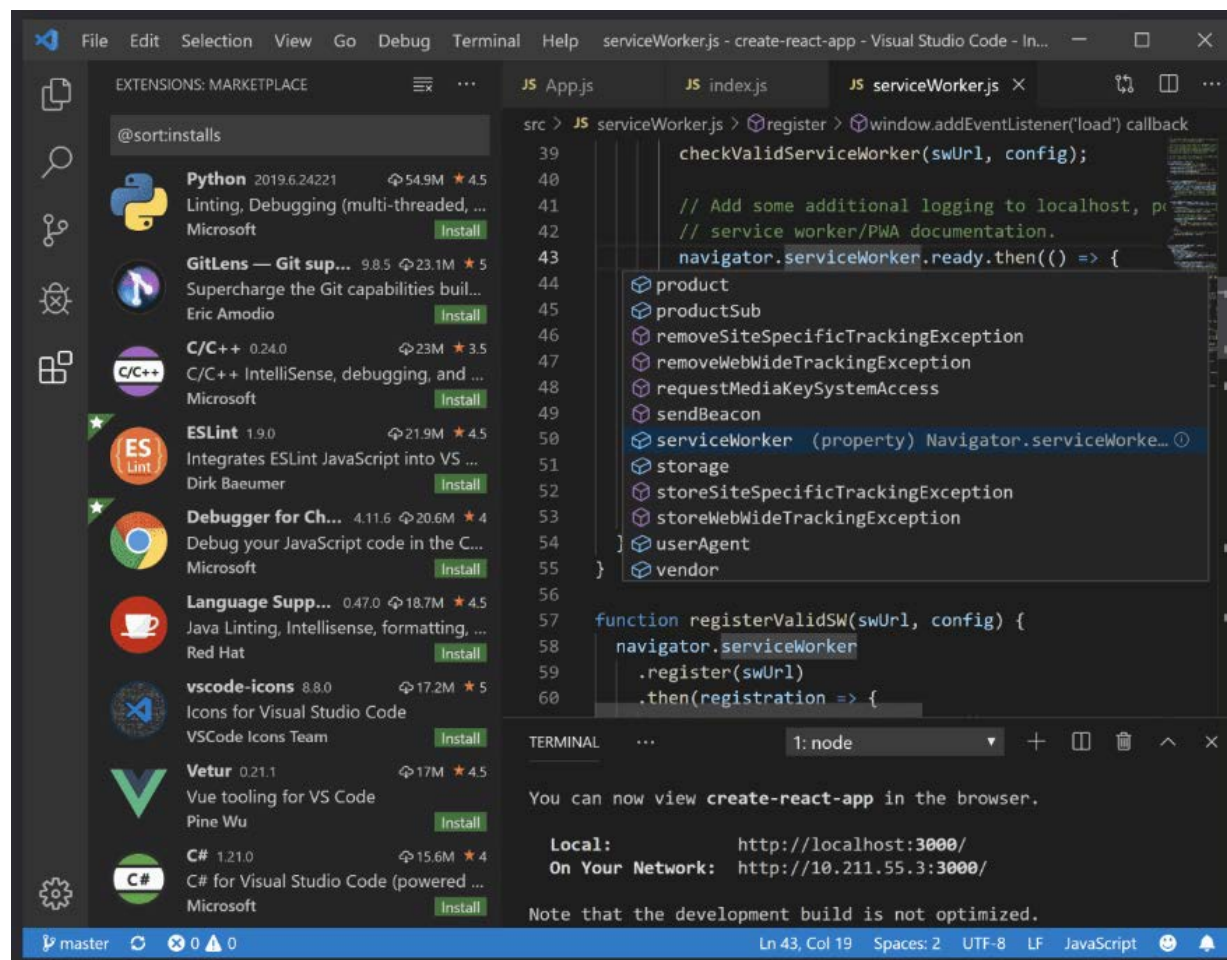
영상처리 및 openCV

openCV 응용



이미지 참조 <https://engineer.emilee.jp/blog/2239-20181030>

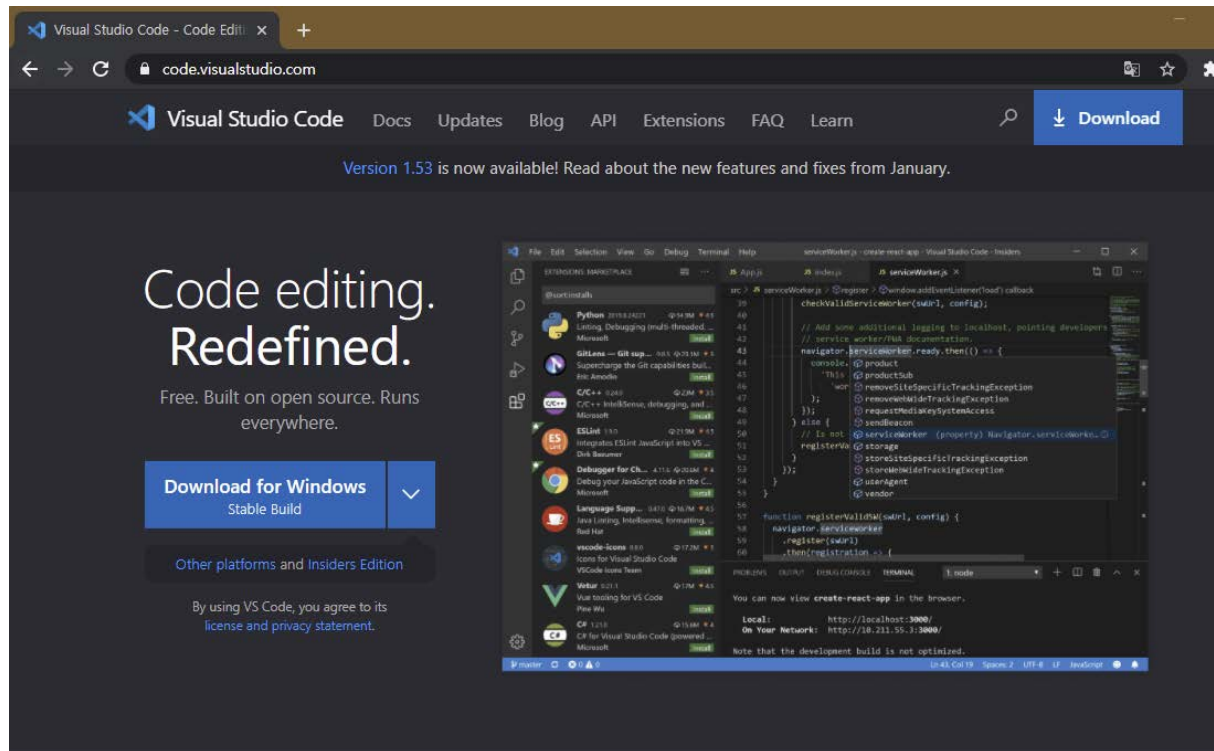
- Visual Studio Code 설치
- Python 설치
- numpy 설치



Visual Studio Code 설치

11

<https://code.visualstudio.com/>



IntelliSense



Run and Debug

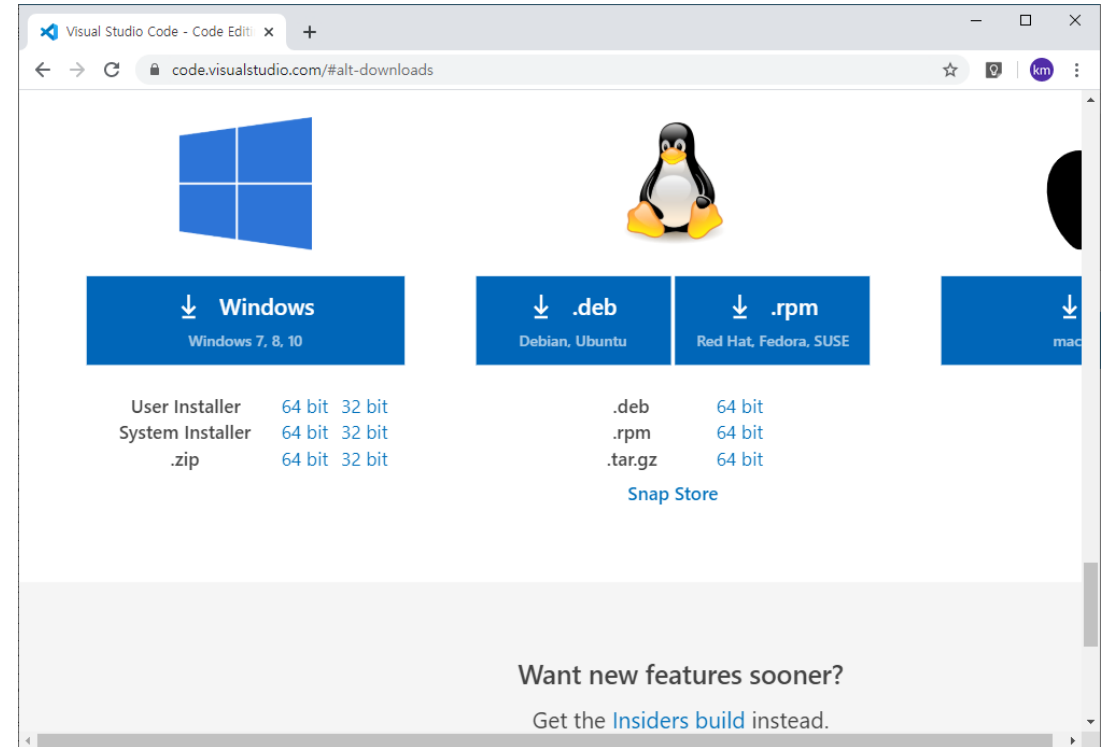


Built-in Git

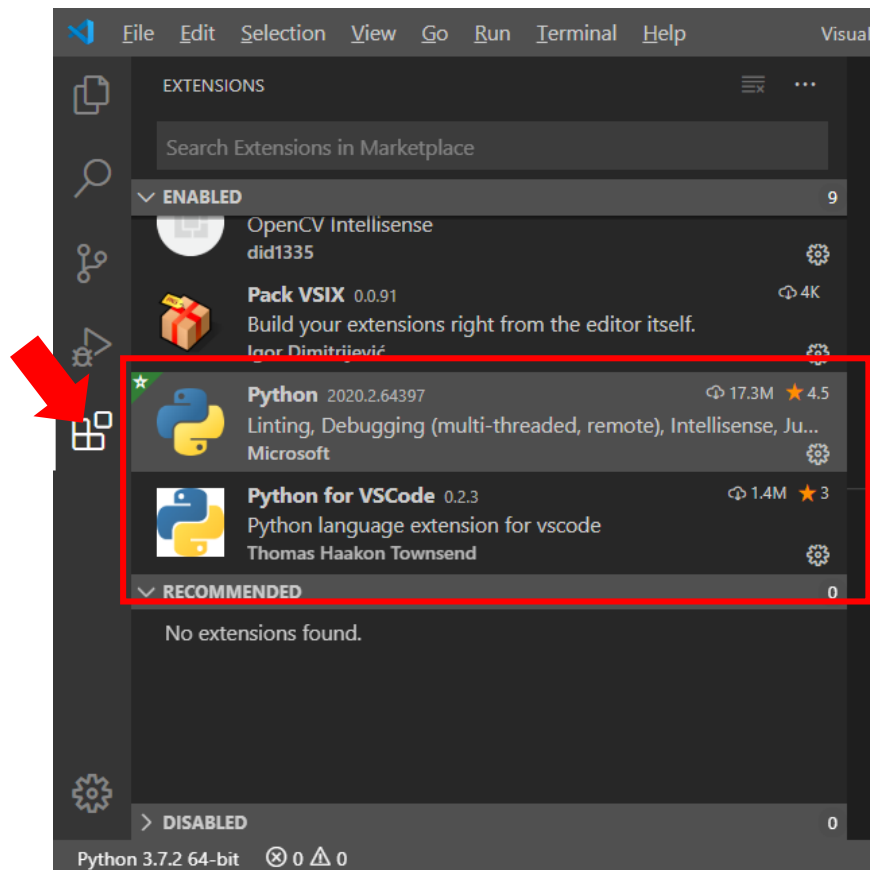


Extensions

운영체제에 맞는 설치 파일 선택



- extensions에서 python을 설치함



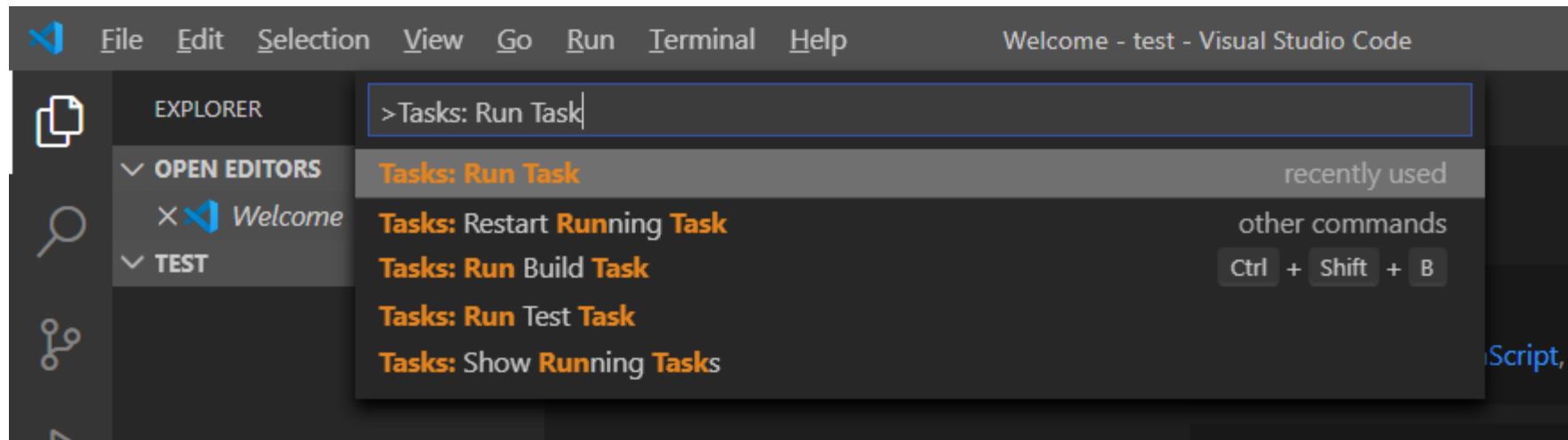
Python 3.x

버전	배포 날짜
3.7	2018-06-27
3.6	2016-12-23
3.5	2015-09-13
3.4	2014-03-17
3.3	2012-09-29
3.2	2011-02-20
3.1	2009-06-26
3.0	2008-12-03

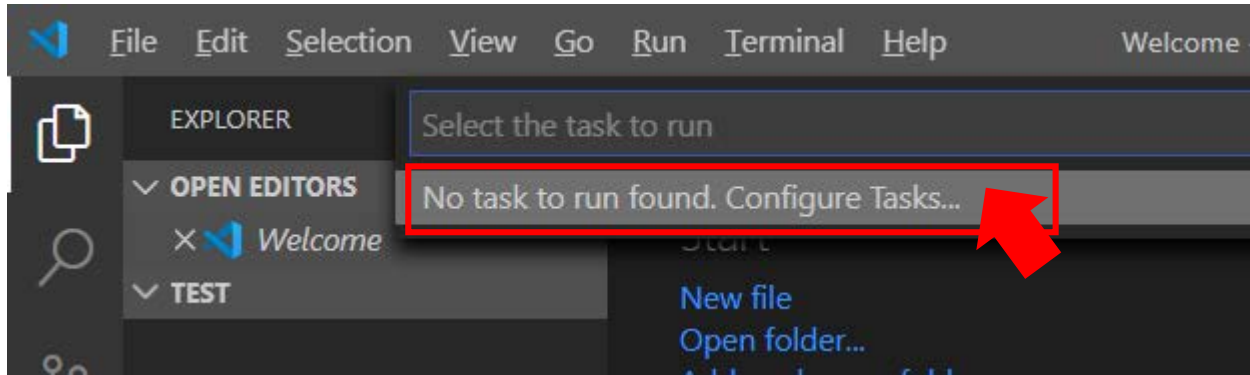
Python 2.x

버전	배포 날짜
2.7	2010-07-03
2.6	2008-10-02
2.5	2006-09-19
2.4	2004-11-30
2.3	2003-07-29
2.2	2001-12-21
2.1	2001-04-15
2.0	2000-10-16

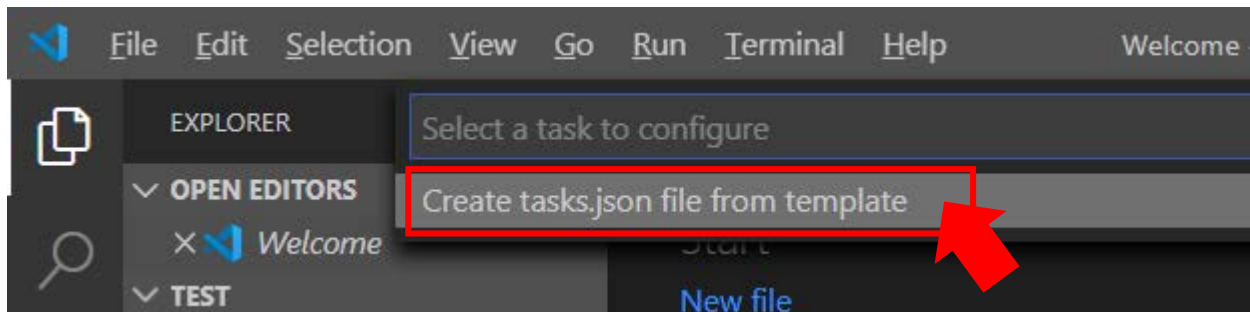
- 1) 폴더 생성 : File -> Open Folder -> 폴더 생성 후 선택
- 2) tasks.json 생성 후 수정
 - ① View -> Command Palette(Ctrl+Shift+P)
 - ② “Tasks: Run Task” 입력 후, enter



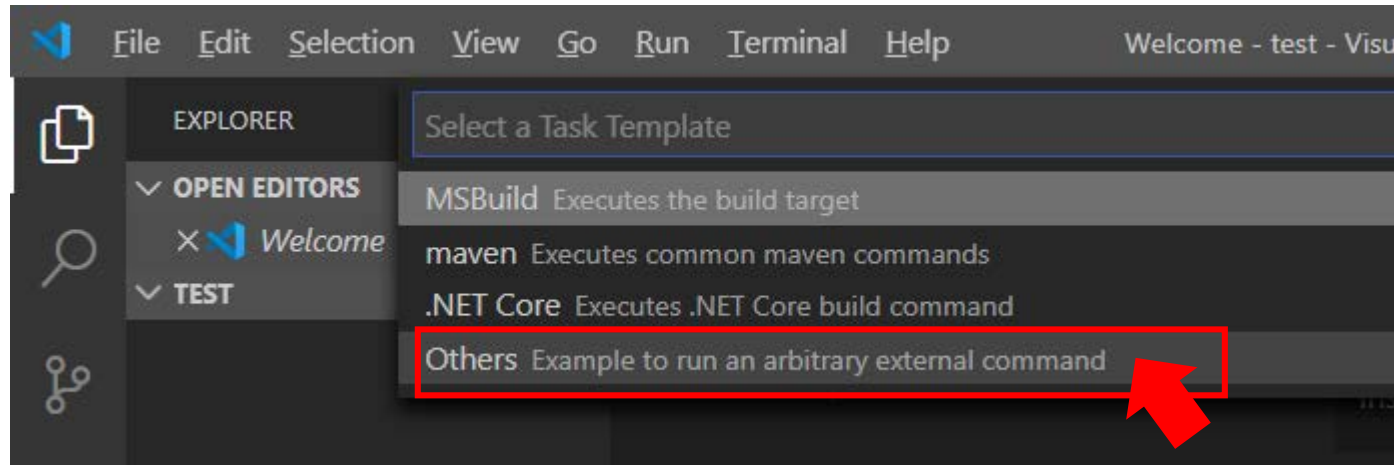
- ③ “실행할 작업이 없습니다. 작업 구성...”을 클릭함



- ④ “템플릿에서 tasks.json 파일 만들기” : Run Task” 입력

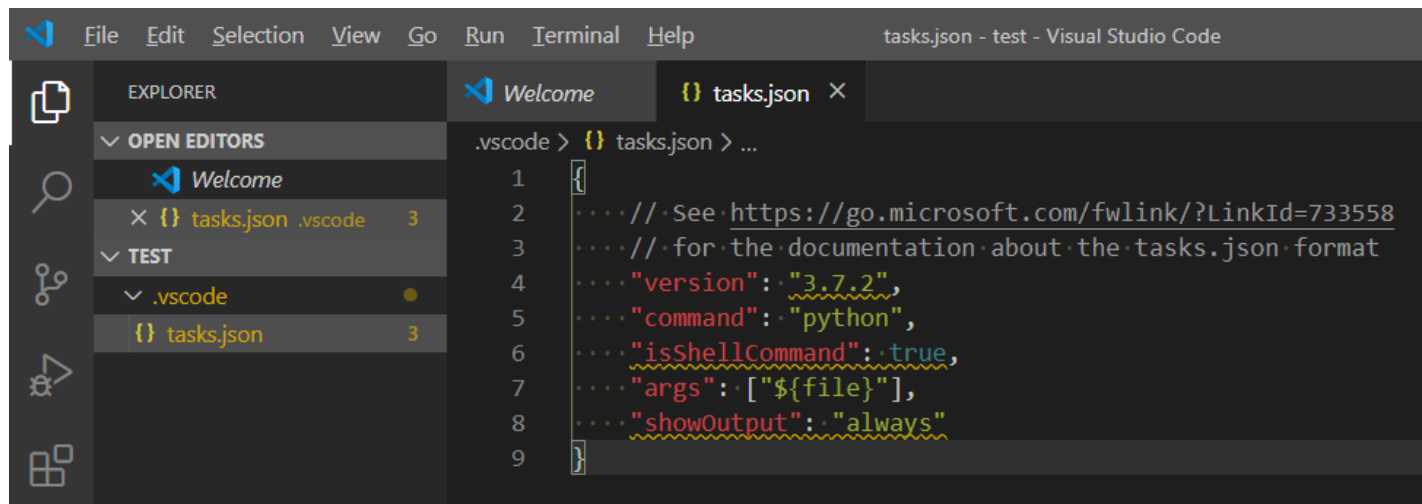


⑤ “Others …”를 클릭함

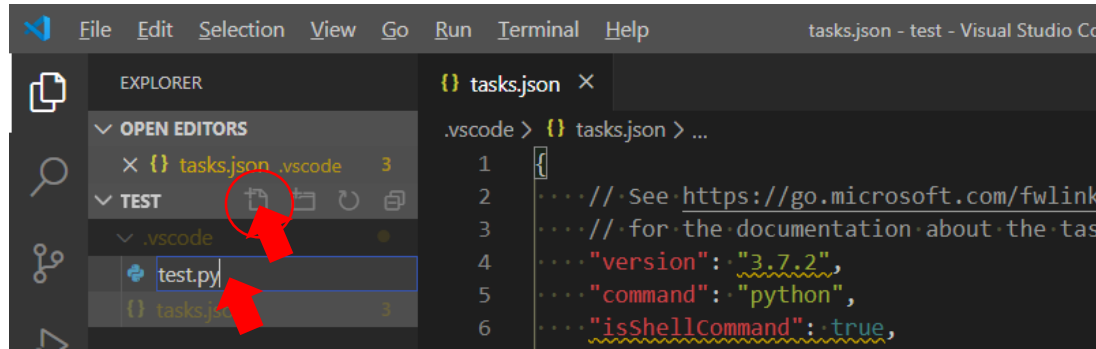


- ⑥ tasks.json 파일 내용을 아래 내용으로 변경함.

```
{  
  // See https://go.microsoft.com/fwlink/?LinkId=733558  
  // for the documentation about the tasks.json format  
  "version": "3.8.8",  
  "command": "python",  
  "isShellCommand": true,  
  "args": ["${file}"],  
  "showOutput": "always"  
}
```

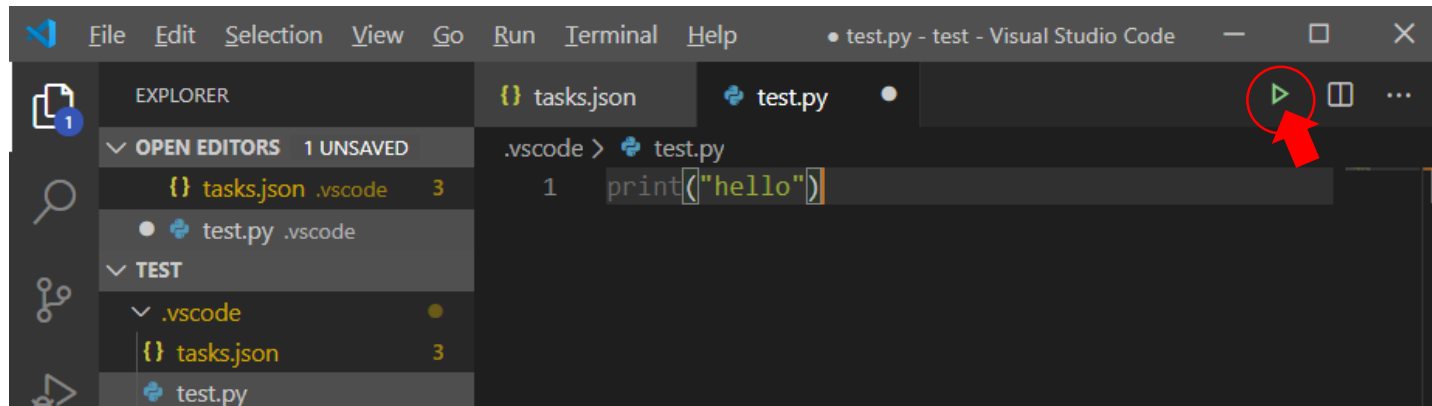
A screenshot of the Visual Studio Code editor interface. The title bar shows 'tasks.json - test - Visual Studio Code'. The Explorer sidebar on the left shows the 'tasks.json' file selected. The main editor area displays the content of 'tasks.json' with line numbers 1 through 9. The JSON content is: {, // See https://go.microsoft.com/fwlink/?LinkId=733558, // for the documentation about the tasks.json format, "version": "3.7.2", "command": "python", "isShellCommand": true, "args": ["\${file}"], "showOutput": "always", }.

⑦ 새 파일(.py)을 생성함



⑧ 프로그램을 작성하고 실행함.

Run->Start Debugging(F5) 또는 오른쪽 상단 초록삼각형 클릭



- 세미콜론(;)
 - 여러 명령문을 한 줄에 작성시
 - ex) `i=1; print(i); i=i+10`
- 주석
 - # : 오른쪽은 모두 주석으로 처리됨
 - ‘‘‘ 과 ‘’’ 사이는 모두 주석으로 처리됨, 여러 줄에 걸쳐서 사용 가능

- 들여쓰기
 - 공백 또는 탭
 - C언어의 { } 와 같은 의미로, 명령문이나 함수의 범위(block)를 지정하는데 사용함.
 - 탭, 공백을 섞어 쓰면 안됨.
 - 프로그램을 실행하였을 때 아래 오류가 나오면, 들여쓰기가 문제가 있는 것이므로 공백이나 탭을 확인함.

indentationError: expected an indented block

- 편집기에 탭이나 공백을 표시하여 주는 기능이 있으면, 매우 편리함.

- 모듈 : 변수, 함수, 클래스를 사용하여 여러 기능의 프로그램 파일(*.py 파일)
- 패키지 : 여러 모듈의 묶음
- 파이썬 표준 라이브러리 : 기본으로 설치되는 모듈, 패키지, 내장 함수를 파이썬 표준 라이브러리(Python Standard Library, PSL)

- 이미 만들어진 변수, 함수, 클래스 등을 사용할 수 있는 방법
- 기본 사용 방법

```
import 모듈명
```

- ex)

```
import keyword  
print(keyword.kwlist)
```

- 기본 출력 함수

```
print(변수명)
```

- ex)
 print("hello")
 print(i)

```
변수 = input(표시 문자열)
```

- 항상 string으로 입력됨
- 정수, 실수를 입력 받으려면 int(), float()을 이용하여 형 변환을 해야 함.
- ex)

```
i = input("값을 입력하세요... ")
```

```
j = input("값을 입력하세요... ")
```

```
k = int(j)
```

- 변수 선언 및 값 설정

변수명 = 값

- 유형을 선언할 필요 없음
- 지정된 값으로 유형이 결정됨
- 대소문자 구별함

Integer (ex: -10, 1)

Floating point (ex: 3.14)

String (ex: 'a', "hello")

Boolean (ex: True, False)

Null data type (ex: None)

- 1) 다음 각 변수를 출력하시오.
 - Integer (ex: -10, 1)
 - Floating point (ex: 3.14)
 - String (ex: 'a', "hello")
 - Boolean (ex: True, False)
 - Null data type (ex: None)
- 2) 두 수의 합을 출력하라.
- 3) 두 문자열을 하나의 문자열로 만들어 출력하라.

- 다음 프로그램에서 틀린 부분을 수정하시오.

```
str1 = 'python'  
print(str1)  
str2 = "p"  
print(str2)  
str3 = "python"  
print(str3)  
str4 = "python"  
print(str4)
```

if A :

①

④

⑤

if A :

①

else:

②

④

⑤

if A :

①

elif B :

②

else:

③

④

⑤



Boolean Operators

- 불리안 값은 참 또는 거짓 값을 가짐.
- 거짓 : False, None, 0, "", 0.0, 0L, 0j, (), [], {}
- 참 : 거짓을 제외한 모든 값



a = 0
b = 7.5
c = 'hi'

불리안 연산자	연산식	값
not	not c	False
or	a or b	True
and	c and b	True
==	a==0	True
!=	a!=b	True
<	a < b	True
>	a > b	False

- 다음 프로그램의 출력값을 구하시오.

```
a = 4
b = 7
c = -3
if c < a < b:
    print("flower")
else:
    print("bee")
```

```
a = 4
b = 7
c = -3
if c < a and a < b:
    print("flower")
else:
    print("bee")
```

```
a = 4; b = 7; c = -3; d = 8

if a != b < c >= d:
    print("flower")
else:
    print("bee")
```