

OpenGL 소개와 구현환경 구축

동명대학교 게임공학과







강영민

OpenGL

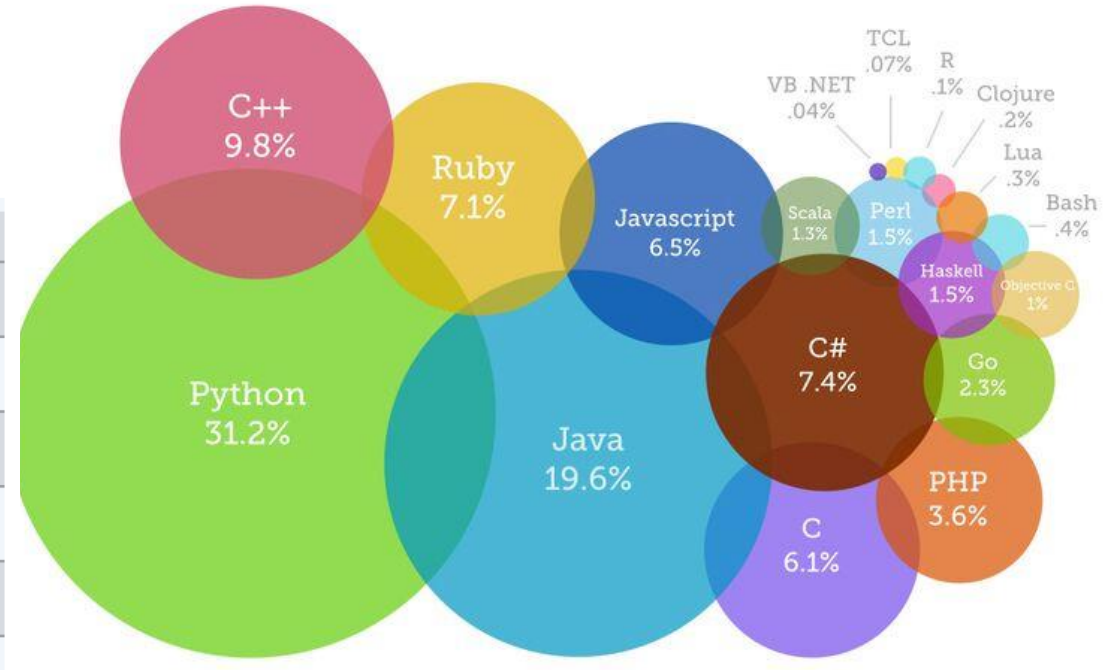
- 플랫폼 독립적인 그래픽스 라이브러리
 - 하드웨어나 운영체제 독립
 - GPU 활용 작업 지원
- 실시간 그래픽스 환경에서의 콘텐츠 생성이 주요 목적
 - 실사 수준 고품질 렌더링을 목표로 하지 않음
 - 최근의 하드웨어 발전에 따라 상당히 높은 수준의 렌더링 가능

프로그래밍 환경 - 언어

- 이 수업에서 사용할 언어
 - 파이썬(Python)

Jun 2023	Jun 2022	Change	Programming Language	Ratings	Change
1	1		 Python	12.46%	+0.26%
2	2		 C	12.37%	+0.46%
3	4	▲	 C++	11.36%	+1.73%
4	3	▼	 Java	11.28%	+0.81%
5	5		 C#	6.71%	+0.59%
6	6		 Visual Basic	3.34%	-2.08%

TIOBE 지수의 2023년 6월 순위표

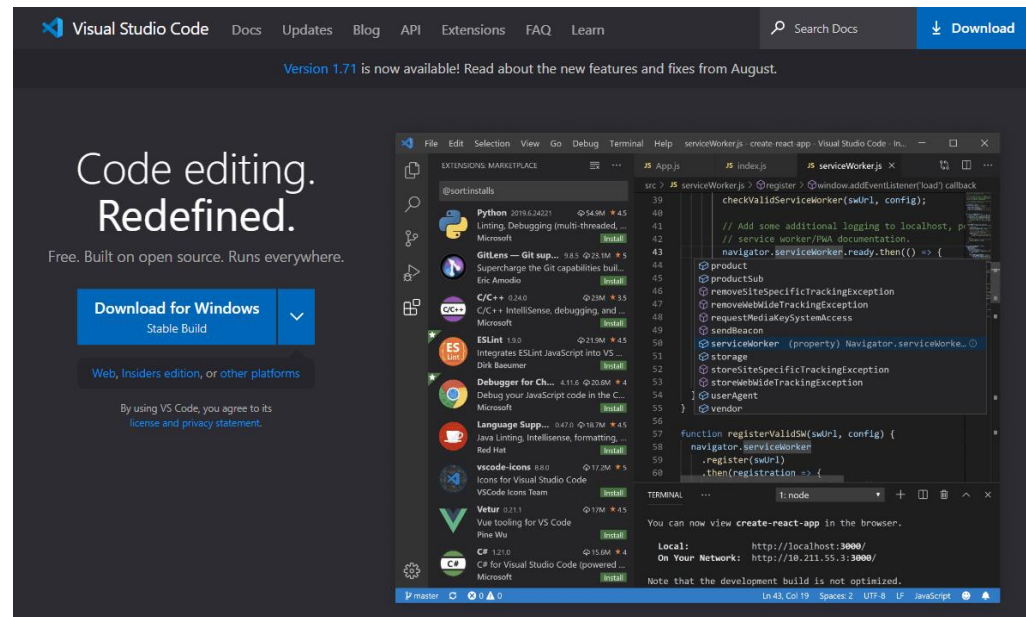


2019년 가장 트렌디한 프로그래밍 언어, 자료출처: <https://learnworthy.net/>

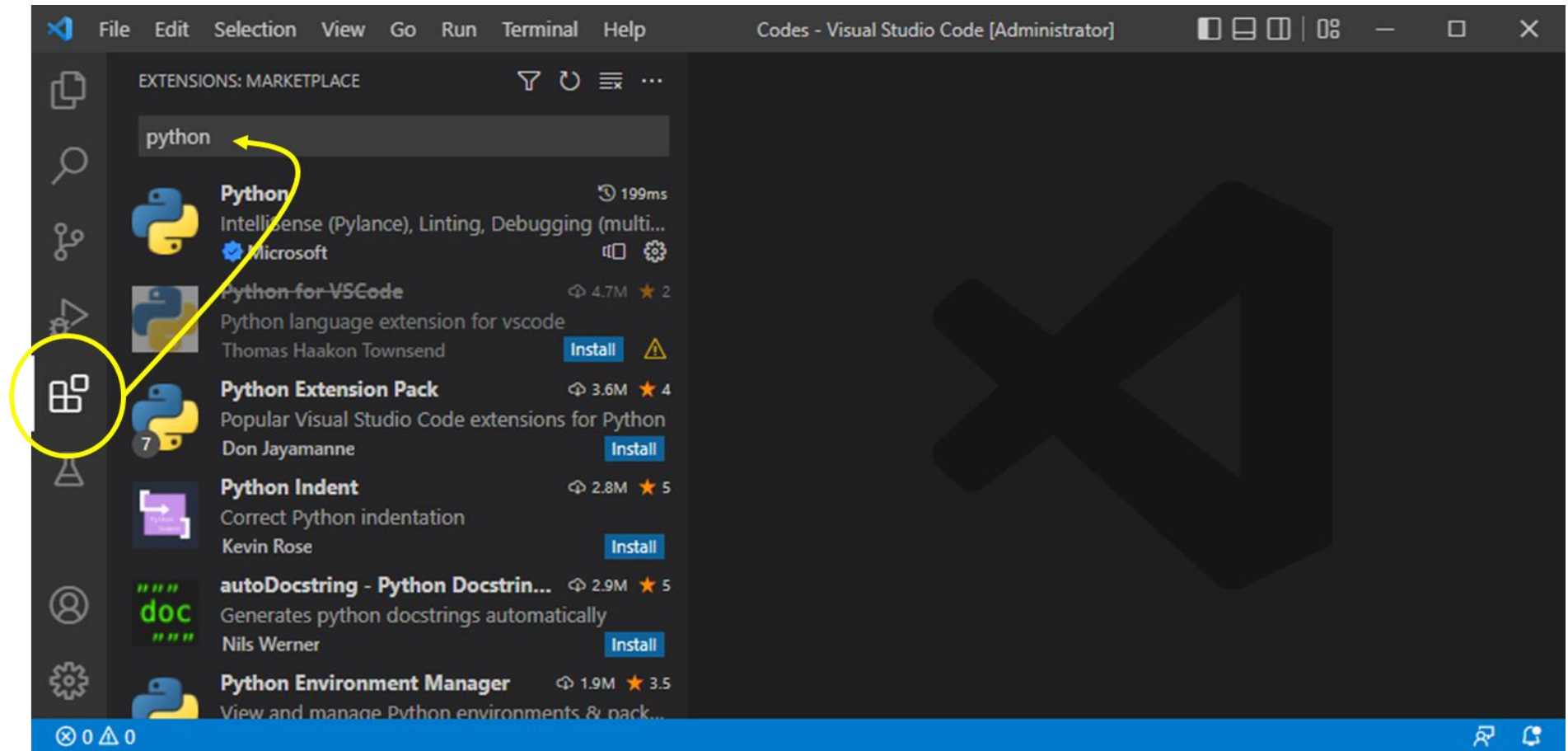
TIOBE is an acronym that stands for "The Importance Of Being Earnest." However, in the context of software development and programming, TIOBE typically refers to the TIOBE Index, which is a well-known index that ranks the popularity of programming languages based on their usage and availability.

Python 환경 구축

- 수업에서 사용할 환경
 - Python 재단의 인터프리터 설치
 - <https://www.python.org>
 - 코드 편집
 - Visual Studio Code
 - <https://code.visualstudio.com>
 - Microsoft의 무료 통합 개발 환경(IDE)
 - 다양한 언어 지원 – 강력한 코드 편집 – 디버깅
 - 확장 가능성
 - 버전 관리 지원
 - 테마 및 사용자 정의
 - 무료 및 오픈 소스

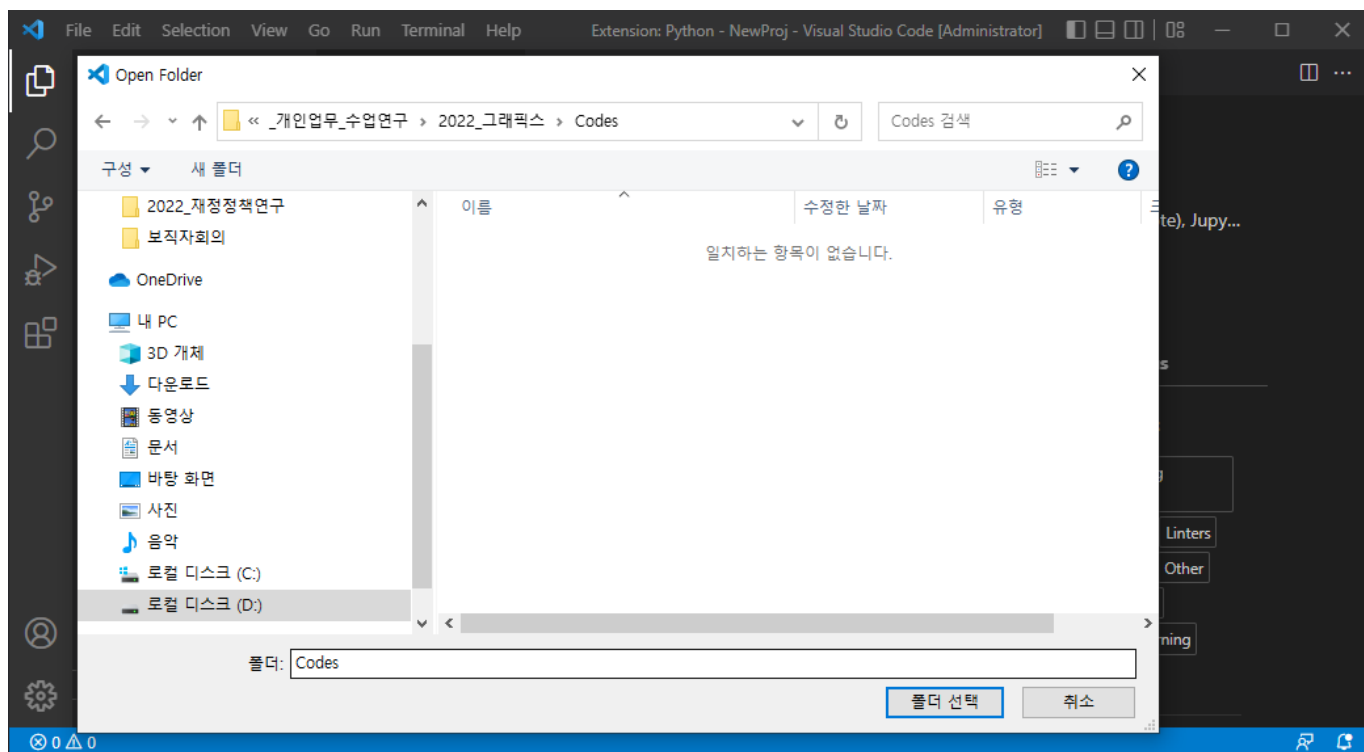


Python 확장 설치



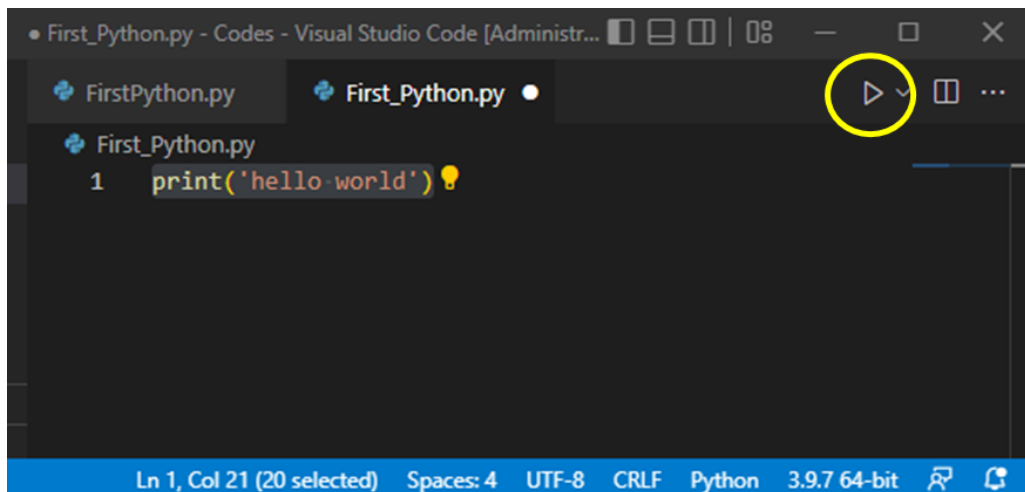
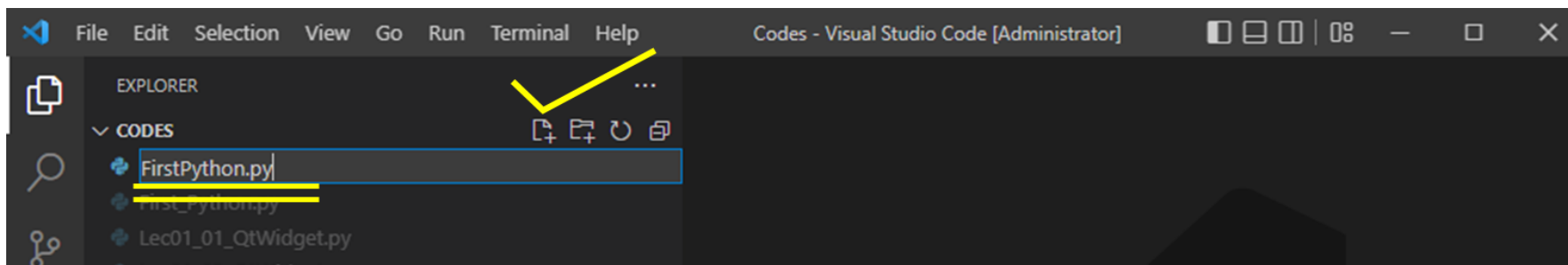
Python 연습

- 폴더를 만들어 프로그램 코드를 만들어 보자
 - File 메뉴의 Open Folder로 자신의 작업공간 지정



Python 연습

- 폴더 내에 파일을 만들어 코딩 연습



Python 연습

- 변수와 연산, 출력

```
# 파이썬 연습 - 이건 코멘트
```

```
# 변수를 사용하자
```

```
a = 3 # 처음으로 변수에 값이 저장될 때 선언이 알아서 이루어짐
```

```
b = 2 # 마찬가지로
```

```
c = a + b**3 ; # 세미콜론은 있든지 없든지...
```

```
print(c) # print는 내장함수
```


Python 연습

- 변수와 연산, 출력

```
a = '나는 정말 이상한 변수' # 변수에 다른 자료형이 들어가면 새롭게 선언됨
b = '나도 나도'

c = a + b
print(c)

a = 123434323
b = 234233423
c = a * b ** 100
print(c)
```

Python 연습

- 흐름제어

```
##### 프로그램의 흐름 제어 #####

age = int (input('나이를 입력하세요: '))
print(age , '세')

# 만약 18세 미만이면 출입금지를 출력하고
# 18세 이상이면 어떤 음료를 마실 것인지 묻는다

if age < 18 :
    print('출입 금지입니다')
elif age < 21:
    print('신분증을 보여주세요')
else :
    beverage = input('어떤 음료를 주문하시겠습니까? ')
    print('네, 알겠습니다.', beverage, '를 준비하겠습니다.')

### 반복 - for / while

for i in range(1, 10, 2) :
    print(i)

for i in [3, 4, 8, 9, 'hello'] :
    print(i)
```

Python 연습

- 리스트 데이터

```
presidents = ['이승만', '윤보선', '박정희', '최규하', '전두환', '노태우', '김영삼', '김대중', '노무현', '이명박', '박근혜', '문재인', '윤석열']
```

```
print(presidents)
```

```
for president in presidents:  
    print(president)
```

```
i = 0  
while i < 5 :  
    print(presidents[i])  
    i += 1
```

```
for i in range(5):  
    print(presidents[i])
```

Python 연습

- 함수

```
import math

def myFunction() :
    print('hahaha')

for i in range(10):
    myFunction()

def 빗변계산(밑변, 높이) :
    # 빗변을 계산하여 반환하자
    빗변 = math.sqrt(밑변**2 + 높이**2)
    return 빗변

result = 빗변계산(3, 4)
print(result)

while True: # 무한 루프
    밑변 = float (input('밑변은 얼마입니까?: '))
    높이 = float (input('높이는 얼마입니까?: '))

    if 밑변 < 0 or 높이 < 0 :
        print('종료합니다.')
        break

    빗변 = 빗변계산(밑변, 높이)
    print('이 삼각형의 빗변은 ', 빗변, '입니다.')
```

Python 학습

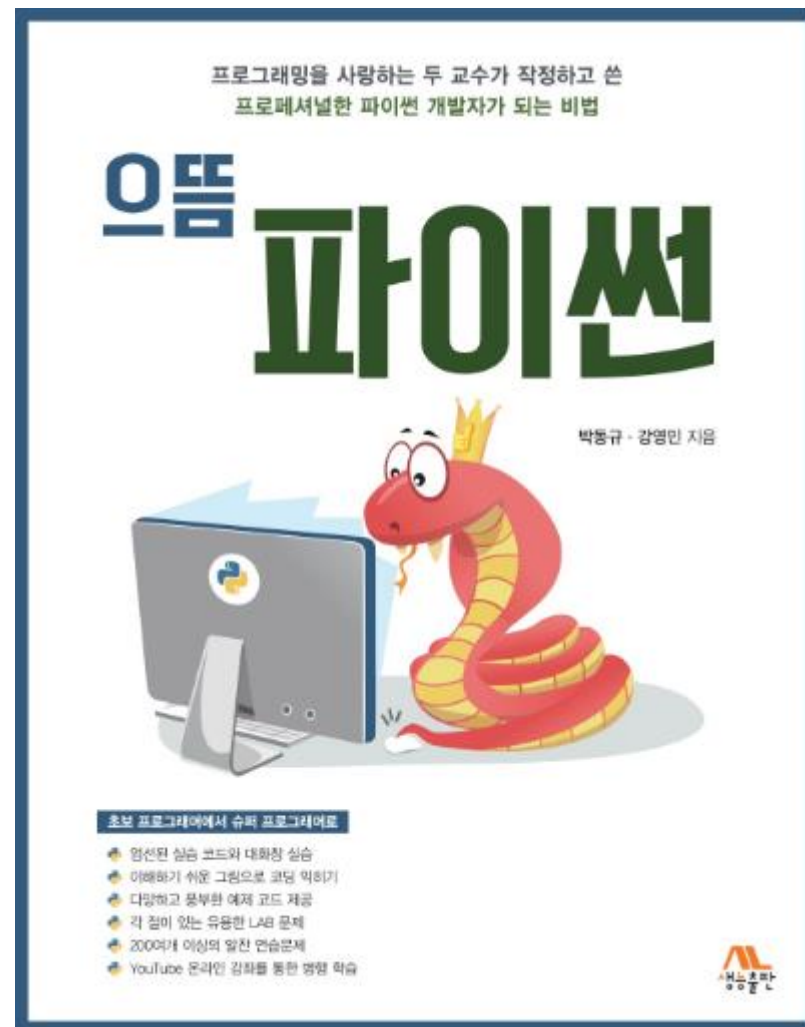
• 파이썬 자율학습

<https://github.com/dongupak/Prime-Python>



도움이 될만한 동영상 강좌

- 널널한 교수의 으뜸 파이썬 Youtube 강좌
 - <https://www.youtube.com/playlist?list=PL2P1Vm9k53HOflCAdyYaOx7hfBHydWNcL>
- 널널한 교수의 기초 파이썬 Youtube 강좌
 - <https://www.youtube.com/watch?v=TK6RLGaM3Ao&list=PL2P1Vm9k53HOEn-QYXAfbIQWkR0OW9mKX>
- 널널한 교수의 고급 파이썬 Youtube 강좌
 - https://www.youtube.com/watch?v=HYg_Zp1186M&list=PL2P1Vm9k53HMUI3i_UKRfWc3ePzAVerhc
- "따라하며 배우는 파이썬과 데이터 과학(2020 출판)" Youtube 강좌
 - https://www.youtube.com/playlist?list=PL2P1Vm9k53HOD_lesbnzAS51t71QeOPms



윈도우 관리

- PyQt 사용

- Qt – 크로스플랫폼 그래픽 사용자 인터페이스 생성 소프트웨어
 - 다양한 운영체제에서 동작하는 윈도우 기반 응용 프로그램 제작 지원
 - 1995년 최초 공개
 - 현재 PyQt6

- PyQt 설치

- Pip 이용
 - Python이 설치되어 있다면, pip 도구를 이용하여 설치 가능
 - `$ pip install PyQt6`

PyQt 설치

```
PS D:\myFolder> pip install PyQt6
Collecting PyQt6
  Downloading PyQt6-6.3.1-cp37-abi3-win_amd64.whl (6.3 MB)
    _____ 6.3/6.3 MB 5.9 MB/s eta 0:00:00
Collecting PyQt6-Qt6>=6.3.0
  Downloading PyQt6_Qt6-6.3.1-py3-none-win_amd64.whl (46.2 MB)
    _____ 46.2/46.2 MB 8.5 MB/s eta 0:00:00
Collecting PyQt6-sip<14,>=13.4
  Downloading PyQt6_sip-13.4.0-cp39-cp39-win_amd64.whl (72 kB)
    _____ 72.9/72.9 kB 4.2 MB/s eta 0:00:00
Installing collected packages: PyQt6-Qt6, PyQt6-sip, PyQt6
Successfully installed PyQt6-Qt6-6.3.1 PyQt6-sip-13.4.0 PyQt6-6.3.1
PS D:\myFolder>
```

윈도우 생성 프로그램

Qt를 이용한 윈도우 생성 프로그램

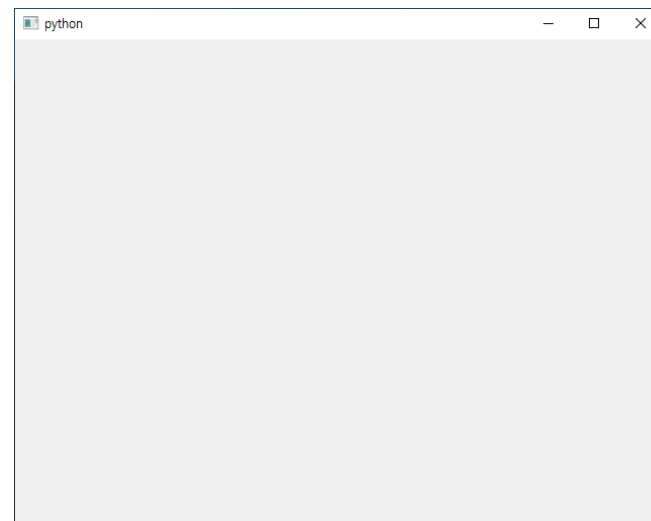
```
from PyQt6.QtWidgets import QApplication, QWidget

# 콘솔 명령으로 인자를 넘겨줄 때 필요 - 당분간 우리는 사용하지 않음
import sys

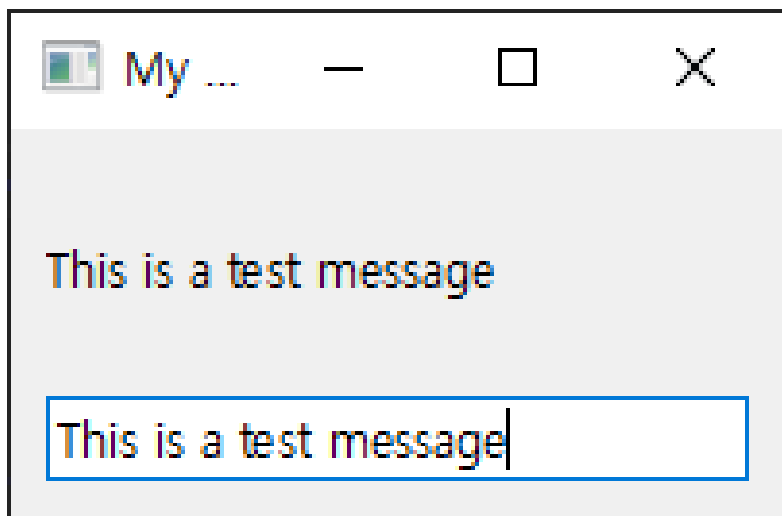
# 애플리케이션은 하나의 QApplication 인스턴스가 필요
# sys.argv에는 명령행에서 입력된 인자를 담고 있음 이를 QApplication 인자에 넘김
# 인자를 사용하지 않을 경우 sys.argv 대신 빈 리스트인 []를 넘길 수도 있음
# app = QApplication([])
app = QApplication(sys.argv)

# Qt 위젯을 생성 - 우리가 윈도우로 사용할 첫 위젯
window = QWidget()
window.show() # 기본적으로 화면에 나타나지 않게 설정되어 있어 보이게 변경

# 애플리케이션이 이벤트 루프에 들어가게 만든다.
app.exec()
```



윈도우 내에 위젯 배치



```
from PyQt6.QtWidgets import QApplication, QMainWindow
from PyQt6.QtWidgets import QLabel, QLineEdit, QVBoxLayout, QWidget

import sys

class MainWindow(QMainWindow):
    def __init__(self):
        # 슈퍼클래스의 초기화를 실행
        super().__init__()

        self.setWindowTitle("My App")

        # QLabel과 QLineEdit 클래스의 객체를 각각 생성한다
        self.label = QLabel()
        self.input = QLineEdit()

        # QLineEdit 객체의 값이 변경되면 QLabel로 전달된다.
        self.input.textChanged.connect(self.label.setText)

        # 생성해 놓은 두 위젯 객체를 수직으로 배치할 레이아웃 객체 생성
        layout = QVBoxLayout()

        # 레이아웃 객체에 두 위젯 객체를 추가하여 담기게 한다.
        layout.addWidget(self.label)
        layout.addWidget(self.input)

        # 윈도우에 표시될 위젯을 담는 컨테이너 객체 생성
        container = QWidget()

        # 컨테이너에 레이아웃 객체를 지정한다. 그 안에 담긴 위젯도 함께
        container.setLayout(layout)

        # 윈도우의 중심 위젯을 우리가 만든 것들이 담긴 컨테이너로 지정
        self.setCentralWidget(container)

app = QApplication(sys.argv)

window = MainWindow()
window.show()

app.exec()
```

OpenGL 설치

```
$ pip install pyOpenGL
```



OpenGL을 사용할 수 있는 시스템 환경 구축이 완료됨

OpenGL 윈도우 생성

```
from OpenGL.GL import *
from OpenGL.GLU import *

from PyQt6.QtWidgets import QApplication, QMainWindow, QWidget
from PyQt6.QtOpenGLWidgets import QOpenGLWidget

import sys

class MyGLWindow(QOpenGLWidget):
    def __init__(self, parent=None):
        super(MyGLWindow, self).__init__(parent)

def main(argv = []):
    app = QApplication(argv)
    window = MyGLWindow()
    window.setWindowTitle('Example1')
    window.setFixedSize(600, 600)
    window.show()
    sys.exit(app.exec())

if __name__ == '__main__':
    main(sys.argv)
```



main 함수를 진입점을 사용하는 형태로 코딩

OpenGL 위젯이 동작하도록 하는 기본 콜백

- 초기화^{initialize} – OpenGL 콘텐츠의 초기화 콜백
- 크기 변경^{resize} – 윈도우가 생성되거나 크기가 변경될 때 호출되는 콜백
- 디스플레이^{display} – 화면에 그림을 그리는 일이 필요할 때 호출되는 콜백

```
class MyGLWindow(QOpenGLWidget) : # QOpenGLWidget 상속

    def __init__(self):
        super().__init__() # 슈퍼클래스 QMainWindow 생성자 실행

        self.setWindowTitle('나의 첫 OpenGL 앱')

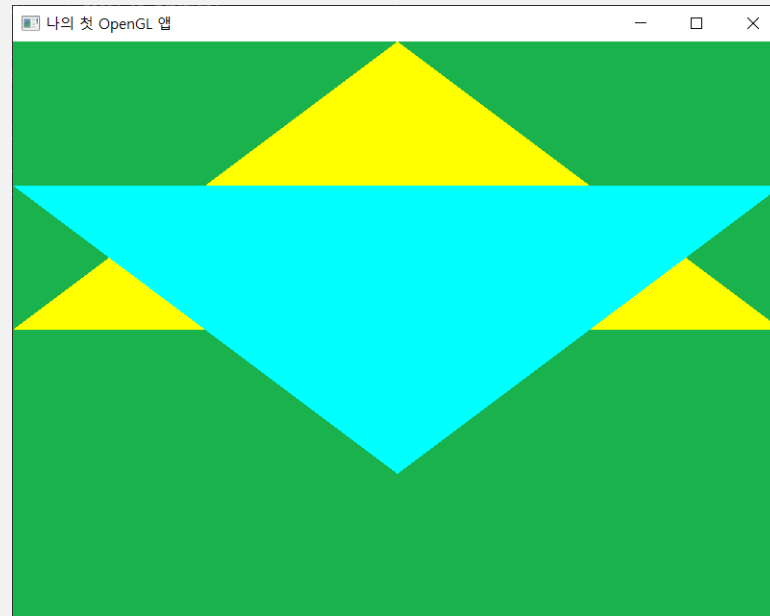
    def initializeGL(self) :
        glClearColor(0.1, 0.7, 0.3, 1.0)

    def resizeGL(self, w: int, h: int) :
        pass

    def paintGL(self):
        pass
```

그림 그리기

```
def paintGL(self):  
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)  
  
    glBegin(GL_TRIANGLES)  
    glColor3f(1, 1, 0)  
    glVertex3f(-1, 0, 0)  
    glVertex3f( 1, 0, 0)  
    glVertex3f( 0, 1, 0)  
    glColor3f(0, 1, 1)  
    glVertex3f(-1, 0.5, 0)  
    glVertex3f( 1, 0.5, 0)  
    glVertex3f( 0, -0.5, 0)  
    glEnd()
```



You are now a graphics programmer!