

# 1. PyQt 처음 감 잡기

2022년 6월 10일 금요일 오후 10:38

- 대략 이 문서 한번 쪽 읽어보고 폴더 0에서 4까지 한번 쪽 훑어 보면 대략 감 잡히는걸 목표로 정리함
- PyQt란?
  - o Qt에 대한 python binding. 뭐 Qt dll들에 대한 python binding이라는 정도로 이해하면 되지 않을까
  - o 그냥 Qt를 python을 이용해 빠르고 간편하게 짜고 싶을때 쓰면 될거라고 생각한다
  - o Distribution문제는 강좌를 좀 더 보고 정리하는 걸로
- PyQt설치
  - o pip install로 설치할 수 있다. 2022/09/10 시점에서는 이렇게
    - pip install PyQt6
  - o Qt Designer라고 UI에 대한 visual editor가 있는데 이건 Qt Designer를 그냥 다운 받아 설치하면 된다
    - 강좌에서는 이것도 pip (pip install pyqt6-tools) 로 설치할 수 있다고 하는데 나는 해보니 에러나고 안됨
      - 아마도 나는 python 3.10쓰고 강좌는 3.9를 써서 생기는 문제인 듯 한데, 뭐 그냥 따로 설치하면 되지...
- Qt Window type classes
  - o QMainWindow
    - Default layout이 처음부터 존재함
      - 예: statusBar, menuBar 같은거
- QDialog
  - o Short term task / brief communication에 유리
  - o 그냥 간단한 메시지나 보여주거나 간단한 인풋 받는 정도로 쓰기 좋다 정도인듯
  - o 그냥 이렇게 쓰기 만만한 템플릿으로 만들어 둔건가 봄
- QWidget
  - o 설명에 따르면 QWidget이 모든 Window의 base class가 된다. 사실 고급 개발을 할거면 QWidget에서 시작해야 할 것 같다
  - o 강좌 예제도 QWidget 상속 받아서 custom ui element 만드는 식임
- Qt Designer
  - o UI를 코드로 짜려면 아무래도 직관적이지 못하니까 Qt Designer를 따로 만든 거인것 같다
  - o 껌데기를 쉽게 만들려면 이걸로 만드는데 좋은 듯
    - Menu의 Form -> Preview를 통해 실제 실행 했을 때 어떻게 생기게 되는지도 바로 볼 수 있음
  - o Qt Designer로 만든 결과(.ui파일)를 사용하는 방법은 크게 두가지가 있음
    1. .ui 파일을 .py파일로 변환
      - pyuic6 -x qd\_result.ui -o converted\_py.py
    2. .ui 파일을 로드

```
import sys
from PyQt6 import uic

class UI(QWidget):
    def __init__(self):
        super().__init__()
        uic.loadUi("qd_result.ui", self)

app = QApplication(sys.argv)
window = UI()
window.show()
app.exec()
```