이 코드를 작성하면서 어려웠던 점 중 하나는 파일의 읽기 및 쓰기 작업을 안전하게 처리하는 것이었습니다. 파일을 읽고 쓰는 과정이 중요했습니다 파일 경로를 절대 경로로 변환하여 사용함으로써 이러한 문제를 최소화하려고 했습니다. 또한, Base64 인코딩과 디코딩 과정에서 데이터 손실이 발생하지 않도록 주의가 필요했습니다. 특히, 파일을 바이너리 모드로 읽고 쓰지 않으면 텍스트 파일로 처리되어 데이터가 손실될 수 있기 때문에, 'rb'와 'wb' 모드를 사용하여 파일을 처리했습니다.

또한, 사용자 입력을 처리하는 과정에서도 사용자의 실수를 방지하기 위해 입력 값을 철저히 검증해야 했습니다. 잘못된 선택을 했을 경우 프로그램이 제대로 작동하지 않을 수 있기 때문에, 올바른 입력 값만을 처리하도록 조건문을 작성하는 데 신경을 썼습니다.

이 과정을 통해 파일 입출력과 Base64 인코딩/디코딩의 원리를 깊이 이해할 수 있었으며, 예외 처리를 통해 프로그램의 안정성을 높이는 방법에 대해서도 생각해 볼 수 있는 계기가 되었습니다. 예를 들어, 파일이 존재하지 않거나 읽기/쓰기 권한이 없는 경우에도 프로그램이 중단되지 않고 적절한 오류 메시지를 출력하도록 개선할 수 있는 여지를 발견했습니다. 이와 같이, 작은 프로그램이라도 세심한 고려와 설계가 필요하다는 점을 배운 유익한 경험이었습니다.

Base64를 C언에서 인코딩 디코딩 할때에는 배열을 사용해서 했기 때문에 어려웠던점이 컸습니다 그런데 파이썬에서는 Base64라는 모듈이 있었다는 걸 알게되었기에 그걸 활용하여 이번 프로그램을 만들 때 굉장히 편리했습니다. 또 학교에서 읽기 쓰기 방법을 배웠던 적이 있었기에 그 방법을 떠올려가며 만들었기에 쉽게 느껴졌습니다 중간에 모르는 부분이 있거나 한 부분은 구글링을 통해 알았습니다.