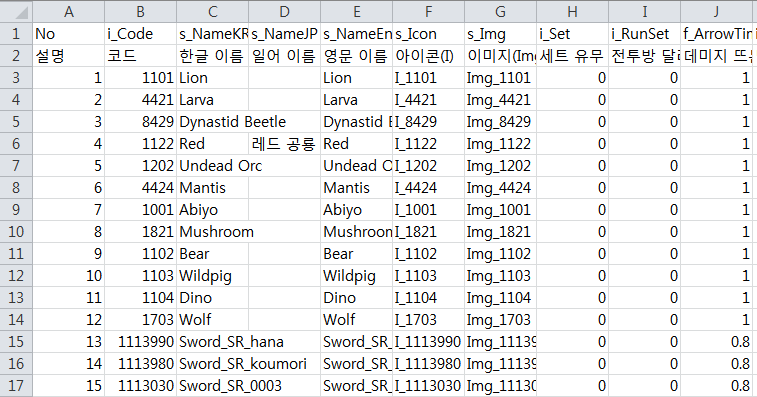
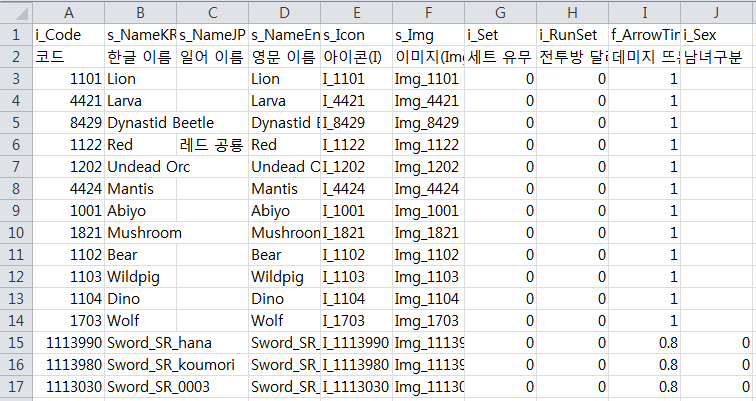
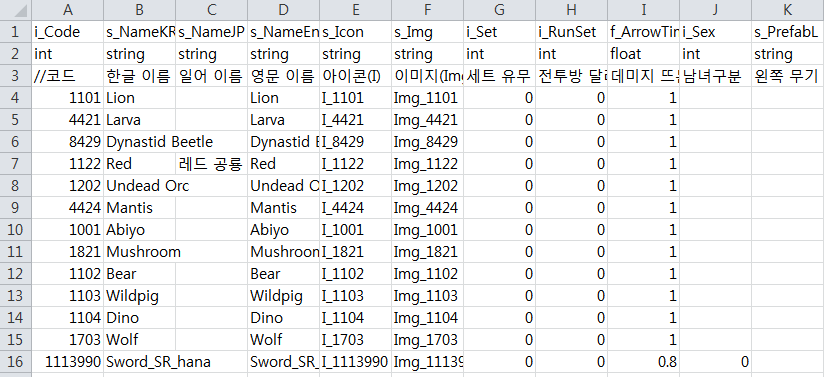
테이블 구조

1. Key가 되는 필드가 맨 왼쪽에 자리해야 합니다.

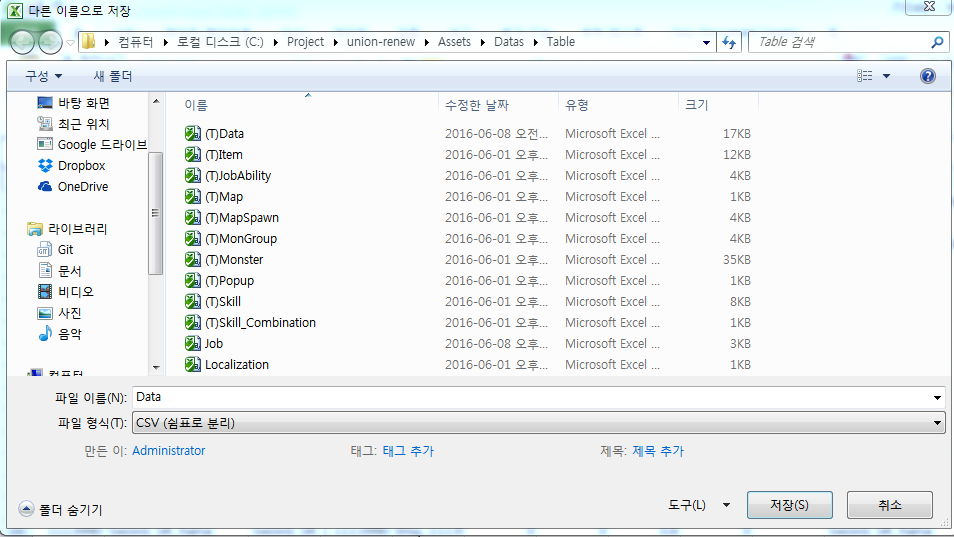




1. 각 항목의 데이터 타입을 알아야 합니다. 더불어 설명부분에는 1번 항목에 “//”를 넣어주세요

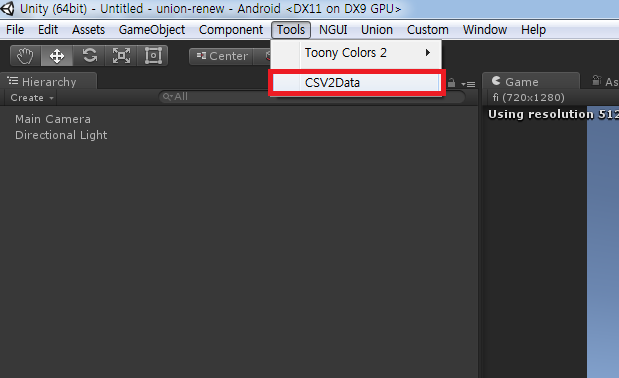


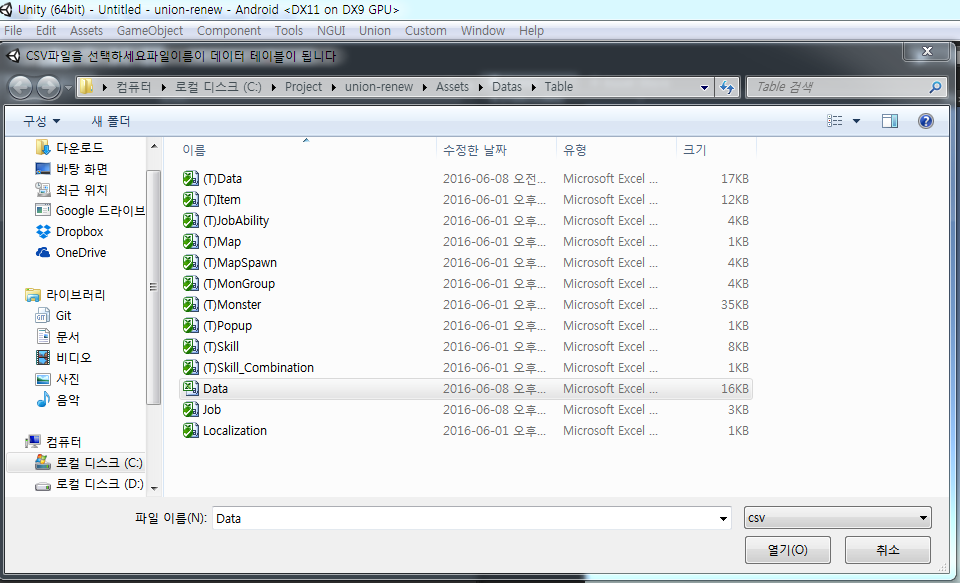
1. 파일이름의 (T)부분을 제외하고 저장합니다. 파일명의 앞자리는 대문자를 권장합니다.



1. 데이터가 준비되면 유니티의 툴에서 변환을 준비합니다.

창이 뜨면서 변환된 파일을 찾아 선택합니다.

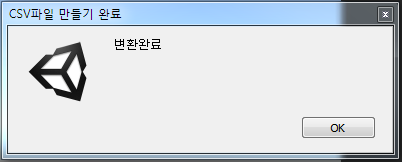




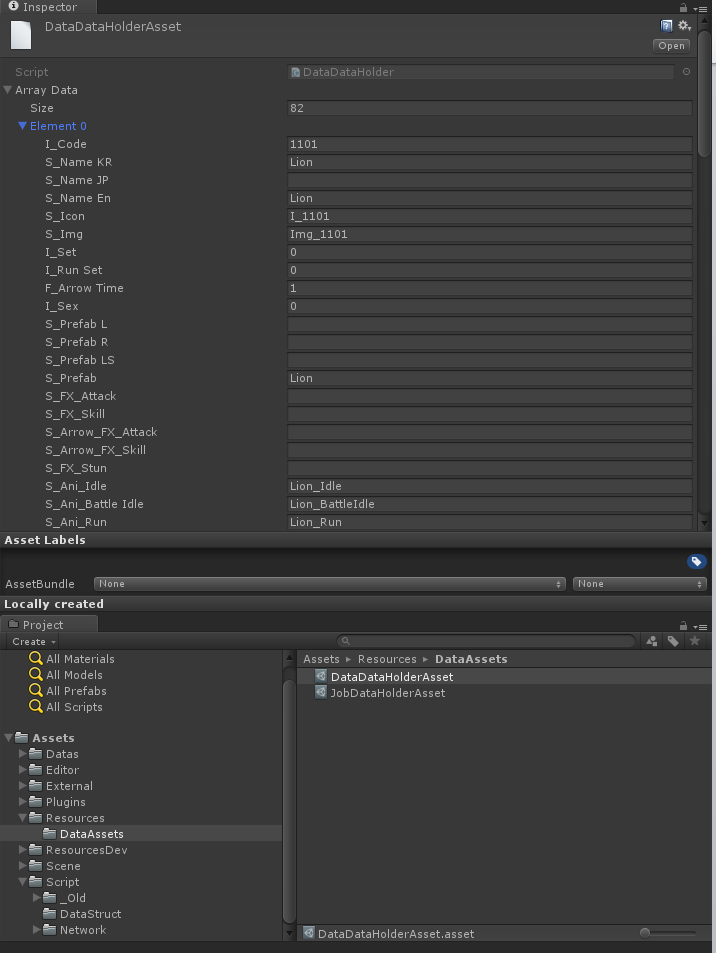
이때 에러가 발생할 수 있는데 이는 이 과정에서 자동으로 소스에서 사용할 데이터 구조를 만들고 이 데이터 구조가 아직 반영이 안된 상태일 경우입니다.

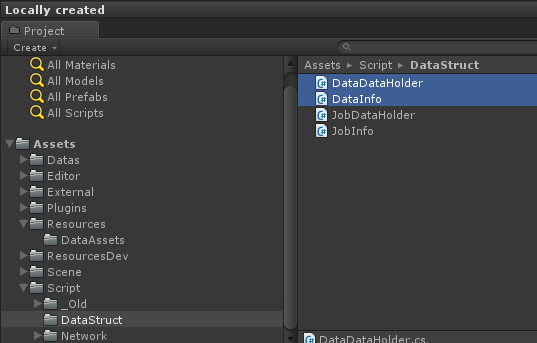
이럴 경우 에디터를 한번 클릭하면 새로 추가된 소스가 추가되며 다시한번 하시면 이 과정을 반복해 주세요

1. 다음과 같이 에러 메시지 없이 완료창이 나오면 성공입니다.



1. 완성이 되면 결과로 다음과 같은 파일들이 나옵니다.

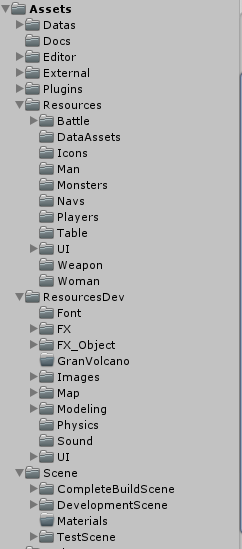




만들어진 DataDataHolerAsset은 유니티에서 바로 읽을 수 있는 데이터 형태에 에디터에서 바로 고칠 수 있는 구조로 되어 있어 간단하게 스탯을 바꿔 테스트할 경우 유용합니다.

주의 사항은 여러 번 만들 경우 DataDataHolerAsset(1) DataDataHolerAsset(2)와 같이 여러 파일들이 만들어지니 확인을 부탁드립니다.

**-폴더구조**



Datas – 리펙토링전 Resources에 있던 자료들로 필요한 것들만 Resources로 옮기며 제거중

Editor – 에디터상에서 돌아가는 Unity기본기능을 확장하는 편의툴들이 모여있음

External –외부의 Asset들을 모아놓은 폴더

Resources – 게임중에

