주간 졸업 프로젝트 수행 내용

인공지능 학습 데이터 셋을 만들기 위해서 많은 시간을 소요 했지만, 다 만들어지지 않았고, 시간이 너무 지체 되는 것 같아서, Python을 활용하여 데이터 셋을 임의적으로 랜덤 함수를 이용해서 만들어봤다.

데이터셋은 입고, 출고, 재고, 불량 데이터를 뽑아냈다.

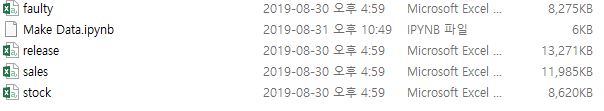
데이터 셋은 Pandas 라이브러리를 활용해서 표를 만들고, numpy 활용하여 데이터 배열을 만들었다.

날짜 계산 라이브러리는 Datetime 라이브러리를 활용하여, 만들었고, 기준점은 많은 데이터를 기준으로 가지고 학습시켜보고 싶어서 2000년도부터 2018년도까지의 데이터를 만들어서 학습을 하기로 잡았다.

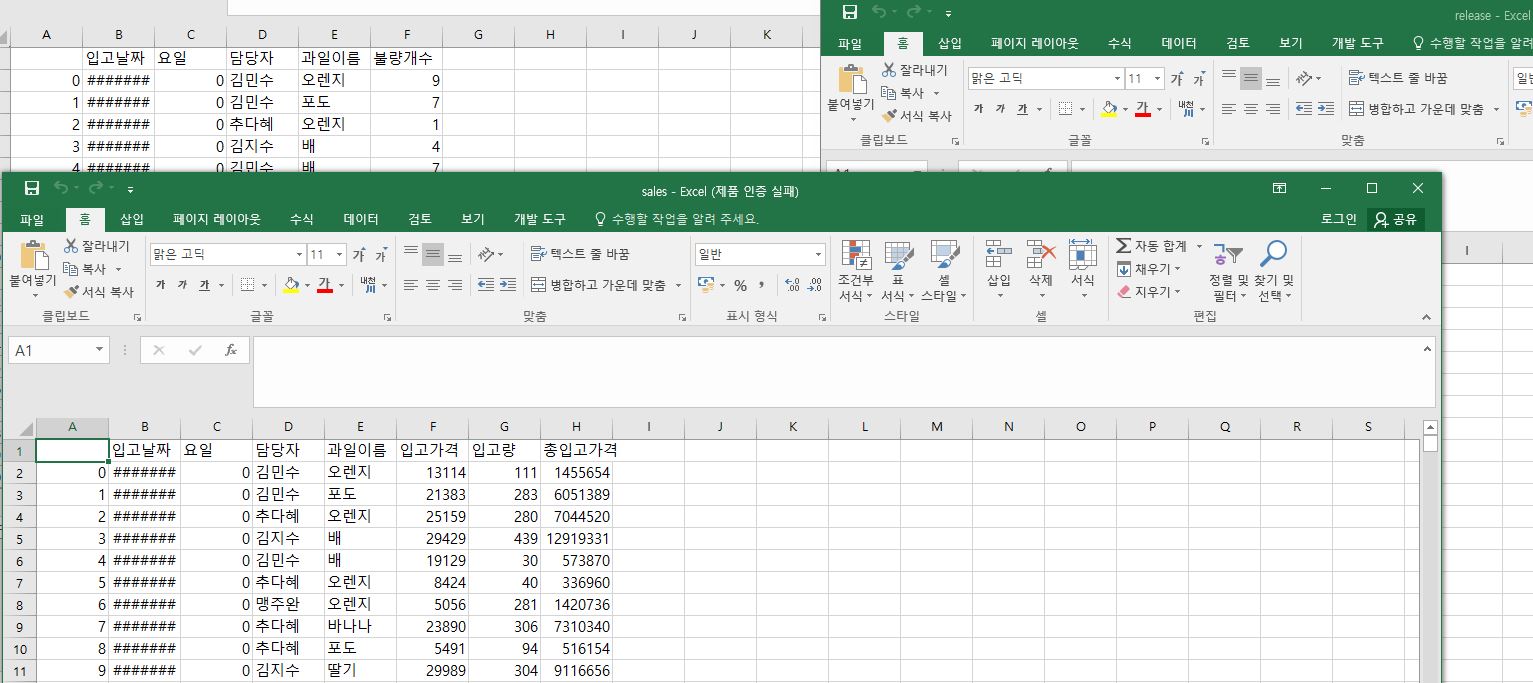
데이터셋 뽑은 코드는 다음과 같다.

|  |
| --- |
| from pandas import Series, DataFrame  import numpy as np  import datetime  import random  name = np.array(['김민수','추다혜','김지수','맹주완'])  fruit = np.array(['바나나','오렌지','사과','딸기','포도','배'])  sales\_data = {'입고날짜' : [], '요일' : [], '담당자' : [], '과일이름' : [], '입고가격' : [], '입고량' : [], '총입고가격' : []}  faulty\_data = {'입고날짜' : [], '요일' : [], '담당자' : [], '과일이름' : [], '불량개수' : []}  release\_data = {'출고날짜' : [], '요일' : [], '담당자' : [], '과일이름' : [], '출고가격' : [], '출고량' : [], '총출고가격' : []}  stock\_data = {'입고날짜' : [], '요일' : [], '담당자' : [], '과일이름' : [], '재고' : []}  today = datetime.date(2000,1,1)  while True:  nowTuple = today.timetuple()  #월  year = nowTuple.tm\_year  #요일  weekday = nowTuple.tm\_wday  #년도가 2019년이라면 break  if(year == 2019):  break  elif (weekday == 6 or weekday == 5):  today = today + datetime.timedelta(days=1)  else:  #rand\_day\_sales\_volume 판매량  #rand\_box 입고박스  #today 날짜  #rand\_price 입고가격  #all\_price 입고가격 \* 입고박스  #faulty\_box 불량율  rand\_day\_sales\_volume = random.randrange(1,100)  for j in range(1,rand\_day\_sales\_volume):  name\_temp = np.random.choice(name)  fruit\_temp = np.random.choice(fruit)  #입고  rand\_box = random.randrange(1,600)  rand\_price = random.randrange(5000,30000)  all\_price = rand\_box \* rand\_price  sales\_data['입고날짜'].append(today)  sales\_data['요일'].append(weekday)  sales\_data['담당자'].append(name\_temp)  sales\_data['과일이름'].append(fruit\_temp)  sales\_data['입고가격'].append(rand\_price)  sales\_data['입고량'].append(rand\_box)  sales\_data['총입고가격'].append(all\_price)  #불량  faulty\_box = random.randrange(0,10)  faulty\_data['입고날짜'].append(today)  faulty\_data['요일'].append(weekday)  faulty\_data['담당자'].append(name\_temp)  faulty\_data['과일이름'].append(fruit\_temp)  faulty\_data['불량개수'].append(faulty\_box)  #출고  release\_data['출고날짜'].append(today)  release\_data['요일'].append(weekday)  release\_data['담당자'].append(name\_temp)  release\_data['과일이름'].append(fruit\_temp)  release\_price = rand\_price + rand\_price \* 0.1  release\_data['출고가격'].append(release\_price)  release\_box = rand\_box - faulty\_box  if (release\_box <= 1):  rand\_release\_box = 1  else:  rand\_release\_box = random.randrange(1,release\_box)  release\_data['출고량'].append(rand\_release\_box)  release\_data['총출고가격'].append(rand\_release\_box \* release\_price)  #재고  stock\_data['입고날짜'].append(today)  stock\_data['요일'].append(weekday)  stock\_data['담당자'].append(name\_temp)  stock\_data['과일이름'].append(fruit\_temp)  stock = rand\_box - faulty\_box - rand\_release\_box  stock\_data['재고'].append(stock)    today = today + datetime.timedelta(days=1)  #CSV 파일 출력  sales\_frame = DataFrame(sales\_data)  sales\_frame.to\_csv('sales.csv',mode='w',encoding='ms949')  faulty\_frame = DataFrame(faulty\_data)  faulty\_frame.to\_csv('faulty.csv',mode='w',encoding='ms949')  release\_frame = DataFrame(release\_data)  release\_frame.to\_csv('release.csv', mode='w', encoding='ms949')  stock\_frame = DataFrame(stock\_data)  stock\_frame.to\_csv('stock.csv',mode='w',encoding='ms949') |

뽑은 파일 결과



입고, 출고, 재고, 불량 데이터 CSV파일



뽑아낸 파일 결과

다음주 할 일

1. DB 모델링을 다시 하고, 뽑아낸 데이터를 DB에 삽입하기

2. 인공지능 모델과 DB 연동하여 학습 한번 시켜보기