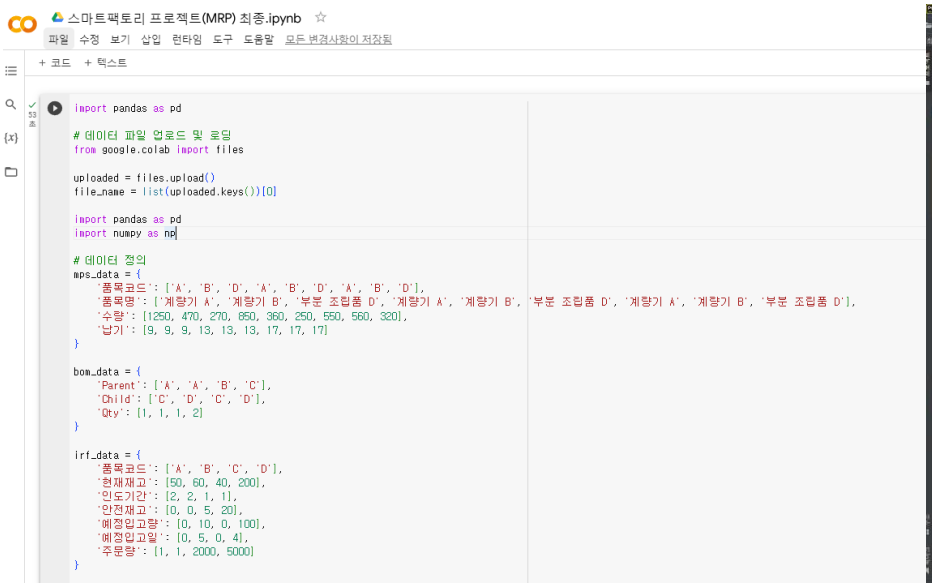
스마트 팩토리 플랫폼과 설계
MRP 프로젝트

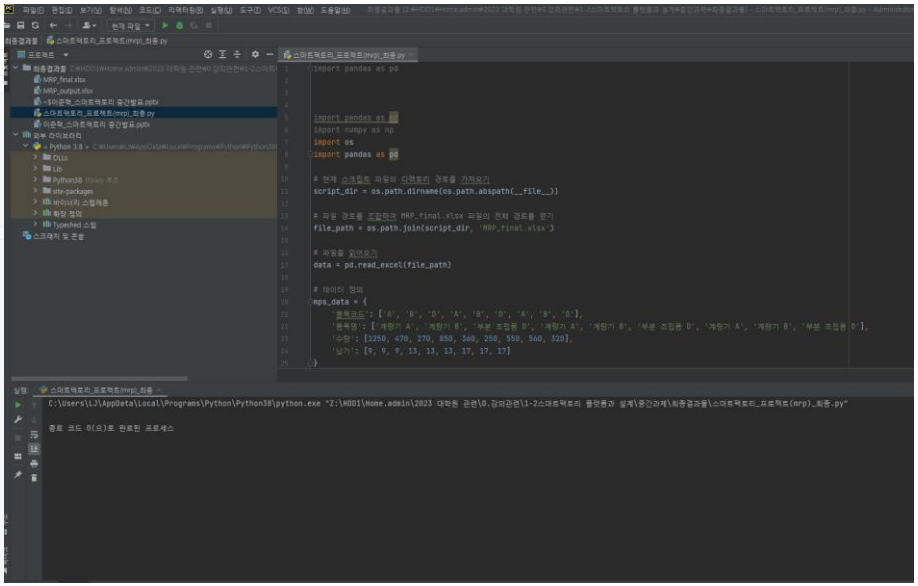
이 준 혁 (2023250419)
산 업 인 공 지 능 학 과

Dept. of Industrial AI

실행 프로그램

파일 경로 문제로 코랩에서 직접 불러오는 방법으로 실행
파이참에서 경로 지정 방법으로 실행





데이터 불러오기

```python
import pandas as pd
import numpy as np
import os
import pandas as pd

# 현재 스크립트 파일의 디렉토리 경로를 가져오기
script_dir = os.path.dirname(os.path.abspath(__file__))

# 파일 경로를 조합하여 MRP_final.xlsx 파일의 전체 경로를 얻기
file_path = os.path.join(script_dir, 'MRP_final.xlsx')

# 파일을 읽어오기
data = pd.read_excel(file_path)
```

1. import 문을 사용하여 필요한 라이브러리인 pandas, numpy, os 를 불러오기.

2. script_dir 변수를 사용하여 현재 스크립트 파일의 디렉토리 경로를 얻기

3. os.path.join() 함수를 사용하여 script_dir과 'MRP_final.xlsx 파일 이름을 결합하여 업로드 해야 하는 파일 전체 경로 가져오기

4. pd.read_excel() 함수를 사용하여 file_path에서 MRP_final.xlsx 파일을 읽기

MRP 에 사용할 MPS 데이터 정의

```
mps_data = {
    '품목코드': ['A', 'B', 'D', 'A', 'B', 'D', 'A', 'B', 'D'],
    '품목명': ['계량기 A', '계량기 B', '부분 조립품 D', '계량기 A', '계량기 B', '부분 조립품 D', '계량기 A', '계량기 B', '부분 조립품 D'
    '수량': [1250, 470, 270, 850, 360, 250, 550, 560, 320],
    '납기': [9, 9, 9, 13, 13, 13, 17, 17, 17]
}
```

품목코드	품목명	수량	납기
A	계량기 A	1,250	9
B	계량기 B	470	9
D	부분 조립품 D	270	9
A	계량기 A	850	13
B	계량기 B	360	13
D	부분 조립품 D	250	13
A	계량기 A	550	17
B	계량기 B	560	17
D	부분 조립품 D	320	17

MRP 프로세스를 위한 초기 입력 데이터 정의로,
각 품목코드, 품목명, 수량, 납기로 나뉘어진 열을 정의함

MRP 에 사용할 BOM, IRF 데이터 정의

```python
bom_data = {
    'Parent': ['A', 'A', 'B', 'C'],
    'Child': ['C', 'D', 'C', 'D'],
    'Qty': [1, 1, 1, 2]
}

irf_data = {
    '품목코드': ['A', 'B', 'C', 'D'],
    '현재재고': [50, 60, 40, 200],
    '인도기간': [2, 2, 1, 1],
    '안전재고': [0, 0, 5, 20],
    '예정입고량': [0, 10, 0, 100],
    '예정입고일': [0, 5, 0, 4],
    '주문량': [1, 1, 2000, 5000]
}
```

Parent	Child	Qty
A	C	1
A	D	1
B	C	1
C	D	2

품목코드	현재재고	인도기간	안전재고	예정입고량	예정입고일	주문량
A	50	2	0			1
B	60	2	0	10	5	1
C	40	1	5			2000
D	200	1	20	100	4	5000

MPS 정의와 마찬가지로 각 열의 항목으로 나누어 정의

Pandas 사용해서 데이터프레임 생성

```
43  # 데이터프레임 생성
44  mps_df = pd.DataFrame(mps_data)
45  bom_df = pd.DataFrame(bom_data)
46  irf_df = pd.DataFrame(irf_data)
47
```

Python 라이브러리인 Pandas를 사용하여 데이터프레임을 생성
데이터프레임: 표 형식의 데이터 구조로, 데이터를 조직화하고 분석하는 데 유용함.

Pd.DataFrame() 함수: 딕셔너리 형태의 데이터를 표 형식의 데이터프레임으로 변환.
-MPS 의 경우 품목코드, 품목명, 수량, 납기 열을 가지는 데이터프레임이 생성
 - 생성된 데이터프레임은 `mps_df` 변수에 저장됩니다.

결과데이터 초기화

```
48      # 결과 테이블 초기화
49      result_data = []
```

프로세스 결과 저장을 위하여 초기화 진행

MRP결과 수행

```
# 각 주차에 대해 MRP 계산
for week in range(4, 18):
    for item in ['A', 'B', 'C', 'D']:
```

for week in range(4, 18):`를 통해 4주에서 17주까지의 주차에 대한 반복 루프를 설정 MRP 계산을 해당 주차별로 수행

for item in ['A', 'B', 'C', 'D']:`를 통해 'A', 'B', 'C', 'D' 품목에 대한 반복 루프를 설정합니다. 각 주차와 품목에 대한 MRP 계산을 수행하기 위한 루프

MRP결과 수행

```
53      for item in ['A', 'B', 'C', 'D']:
54          item_mps = mps_df[(mps_df['품목코드'] == item) & (mps_df['납기'] == week)]
55          item_irf = irf_df[irf_df['품목코드'] == item]
56
57          if not item_mps.empty:
58              demand = item_mps['수량'].sum()
59          else:
60              demand = 0
```

해당 주차 및 품목에 대한 MPS (Master Production Schedule) 데이터를 추출

만약 item_mps 데이터프레임에 값이 있으면, 해당 주차와 품목에 대한 총 소요량 demand 계산.(해당 주차에 필요한 제품 또는 부품의 총 수량)

만약 해당 주차에 MPS 데이터가 없다면 demand는 0으로 설정

MRP결과 수행

```
62          scheduled_receipts = item_irf['예정입고량'].values[0]
63          projected_inventory = item_irf['현재재고'].values[0] + scheduled_receipts
64          net_requirements = demand - projected_inventory
65
66          if net_requirements > 0:
67              planned_orders = net_requirements
68          else:
69              planned_orders = 0
70
71          planned_receipts = max(net_requirements, 0)
```

scheduled_receipts 변수에는 IRF 데이터프레임에서 해당 주차와 품목에 대한 예정 입고량을 가져오기

projected_inventory 변수에는 예상 재고를 계산
(현재 재고와 예정 입고량을 합산한 값)

net_requirements 변수에는 총 소요량과 예상 재고를 비교하여 순 소요량을 계산
(해당 주차에 생산해야 하는 부품 또는 제품의 양)

MRP결과 수행

```
62          scheduled_receipts = item_irf['예정입고량'].values[0]
63          projected_inventory = item_irf['현재재고'].values[0] + scheduled_receipts
64          net_requirements = demand - projected_inventory
65
66          if net_requirements > 0:
67              planned_orders = net_requirements
68          else:
69              planned_orders = 0
70
71          planned_receipts = max(net_requirements, 0)
```

planned_orders 변수에는 순 소요량이 0보다 크면 순 소요량을, 그렇지 않으면 0을 집어넣어 계획 발주량을 계산

Planned_receipts 변수에는 순 소요량이 0 이상인 경우 순 소요량을, 그렇지 않으면 0을 할당하여 계획 발주량을 계산

결과 업데이트

```
        # 결과 테이블 업데이트
        result_data.append({'품목코드': item, '주차': week, '구분': '총소요량', '값': demand})
        result_data.append({'품목코드': item, '주차': week, '구분': '예정입고', '값': scheduled_receipts})
        result_data.append({'품목코드': item, '주차': week, '구분': '예상재고', '값': projected_inventory})
        result_data.append({'품목코드': item, '주차': week, '구분': '순소요량', '값': net_requirements})
        result_data.append({'품목코드': item, '주차': week, '구분': '계획수주', '값': planned_orders})
        result_data.append({'품목코드': item, '주차': week, '구분': '계획발주', '값': planned_receipts})
```

각 주차, 품목 및 결과 구분에 해당하는 정보와 계산된 값들을 result_data 리스트에 추가하기

결과 출력/저장

```
81  # 결과 데이터프레임 생성
82  result_df = pd.DataFrame(result_data)
83
84  # 테이블 출력
85  result_pivot = result_df.pivot(index=['품목코드', '구분'], columns='주차', values='값')
86  result_pivot.reset_index(inplace=True)
87
88  # 결과 데이터프레임을 엑셀 파일로 저장
89  result_pivot.to_excel('MRP_output.xlsx', index=False)
90
91  # 결과 데이터프레임을 바로 출력
92  result_pivot
```

result_df = pd.DataFrame(result_data)을 통해 MRP 계산 결과를 저장한result_data 리스트를 Pandas 데이터프레임으로 변환

result_df.pivot()` 함수를 사용하여 결과 데이터프레임을 피봇함(표형식 정리)

index=['품목코드', '구분'], columns='주차', values='값'`은 데이터를 피벗할 때 어떤 열이 인덱스, 컬럼, 값으로 사용될지를 정의

결과 출력/저장

```
81  # 결과 데이터프레임 생성
82  result_df = pd.DataFrame(result_data)
83
84  # 테이블 출력
85  result_pivot = result_df.pivot(index=['품목코드', '구분'], columns='주차', values='값')
86  result_pivot.reset_index(inplace=True)
87
88  # 결과 데이터프레임을 엑셀 파일로 저장
89  result_pivot.to_excel('MRP_output.xlsx', index=False)
90
91  # 결과 데이터프레임을 바로 출력
92  result_pivot
```

result_pivot.to_excel('MRP_output.xlsx', index=False)을 이용해 결과 데이터프레임을 'MRP_output.xlsx'라는 엑셀 파일로 저장

result_pivot 데이터프레임은 화면에 출력

결과(코랩 피봇출력)

```
# 결과 데이터프레임을 바로 출력
result_pivot
```

파일 선택 복사본 MRP_입력정보2.xlsx

- **복사본 MRP_입력정보2.xlsx**(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 16659 bytes, last modified: 2023. 10. 25. - 100% done

Saving 복사본 MRP_입력정보2.xlsx to 복사본 MRP_입력정보2.xlsx

1 to 24 of 24 entries Filter

index	품목코드	구분	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	A	계획발주	0	0	0	0	0	1200	0	0	0	800	0	0	0	500
1	A	계획수주	0	0	0	0	0	1200	0	0	0	800	0	0	0	500
2	A	순소요량	-50	-50	-50	-50	-50	1200	-50	-50	-50	800	-50	-50	-50	500
3	A	예상재고	50	50	50	50	50	50	50	50	50	50	50	50	50	50
4	A	예정입고	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	A	총소요량	0	0	0	0	0	1250	0	0	0	850	0	0	0	550
6	B	계획발주	0	0	0	0	0	400	0	0	0	290	0	0	0	490
7	B	계획수주	0	0	0	0	0	400	0	0	0	290	0	0	0	490
8	B	순소요량	-70	-70	-70	-70	-70	400	-70	-70	-70	290	-70	-70	-70	490
9	B	예상재고	70	70	70	70	70	70	70	70	70	70	70	70	70	70
10	B	예정입고	10	10	10	10	10	10	10	10	10	10	10	10	10	10
11	B	총소요량	0	0	0	0	0	470	0	0	0	360	0	0	0	560
12	C	계획발주	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	C	계획수주	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	C	순소요량	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40
15	C	예상재고	40	40	40	40	40	40	40	40	40	40	40	40	40	40
16	C	예정입고	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	C	총소요량	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	D	계획발주	0	0	0	0	0	0	0	0	0	0	0	0	0	20
19	D	계획수주	0	0	0	0	0	0	0	0	0	0	0	0	0	20
20	D	순소요량	-300	-300	-300	-300	-300	-30	-300	-300	-300	-50	-300	-300	-300	20
21	D	예상재고	300	300	300	300	300	300	300	300	300	300	300	300	300	300
22	D	예정입고	100	100	100	100	100	100	100	100	100	100	100	100	100	100
23	D	총소요량	0	0	0	0	0	270	0	0	0	250	0	0	0	320

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

결과(파이참 엑셀로 출력)

품목코드	구분	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	계획발주	0	0	0	0	0	1200	0	0	0	800	0	0	0	500
A	계획수주	0	0	0	0	0	1200	0	0	0	800	0	0	0	500
A	순소요량	-50	-50	-50	-50	-50	1200	-50	-50	-50	800	-50	-50	-50	500
A	예상재고	50	50	50	50	50	50	50	50	50	50	50	50	50	50
A	예정입고	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	총소요량	0	0	0	0	0	1250	0	0	0	850	0	0	0	550
B	계획발주	0	0	0	0	0	400	0	0	0	290	0	0	0	490
B	계획수주	0	0	0	0	0	400	0	0	0	290	0	0	0	490
B	순소요량	-70	-70	-70	-70	-70	400	-70	-70	-70	290	-70	-70	-70	490
B	예상재고	70	70	70	70	70	70	70	70	70	70	70	70	70	70
B	예정입고	10	10	10	10	10	10	10	10	10	10	10	10	10	10
B	총소요량	0	0	0	0	0	470	0	0	0	360	0	0	0	560
C	계획발주	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	계획수주	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	순소요량	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40	-40
C	예상재고	40	40	40	40	40	40	40	40	40	40	40	40	40	40
C	예정입고	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	총소요량	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	계획발주	0	0	0	0	0	0	0	0	0	0	0	0	0	20
D	계획수주	0	0	0	0	0	0	0	0	0	0	0	0	0	20
D	순소요량	-300	-300	-300	-300	-300	-30	-300	-300	-300	-50	-300	-300	-300	20
D	예상재고	300	300	300	300	300	300	300	300	300	300	300	300	300	300
D	예정입고	100	100	100	100	100	100	100	100	100	100	100	100	100	100
D	총소요량	0	0	0	0	0	270	0	0	0	250	0	0	0	320

감사합니다