IE 442 – ENTERPRISE INFORMATION SYSTEMS MODELING

MRP IMPLEMENTATION PROJECT

MURAT ÖZTÜRK

2019402093

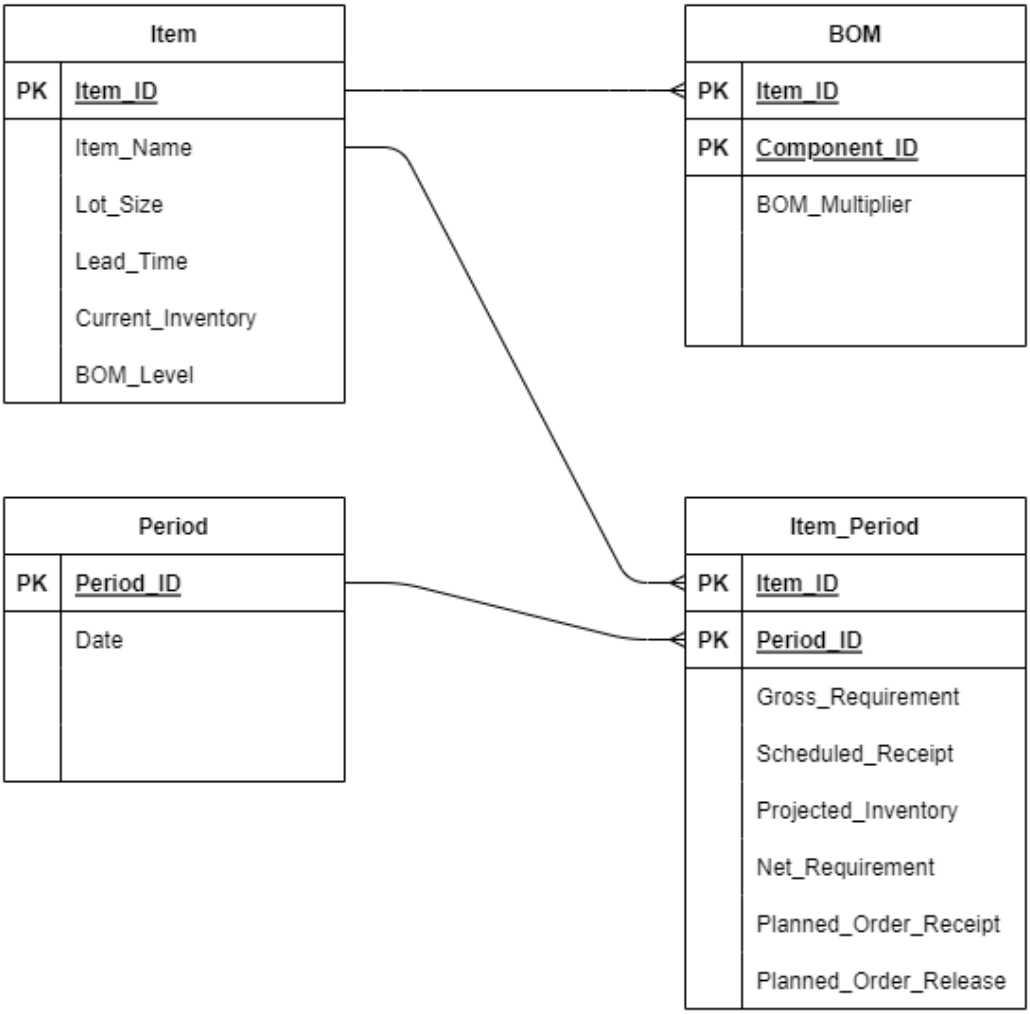
Introduction

The objective of this project was to build a simple Material Requirements Planning (MRP) database with the main algorithms. MRP is a system that deals with inventory and supply management for enterprises. For this project, python and the sqlite3 module will be used. Sqlite3 module is the python package for SQLite. SQLite is a library that provides a relational database management system.

In the project, first, the tables will be created according to the ER diagram. Then, the initial data will be inserted to the tables. After that, the bill of materials (BOM) levels will be calculated for the items. Finally, the MRP calculations will be performed, and the result will be compared with the hand calculated ones.

Note: In this project, backlogs are allowed. Backlogs are shown as putting a negative value in to the projected inventory. However, the data provided to test doesn't require backlogs, thus, this will not be observed.

ER Diagram



Item

Item information are held at the “Item” table. Each item has a unique Item_ID, which is the primary key.

BOM

Items can have a relation with each other, such as being a component of each other. This relations information is held at the “BOM” table. There exists a unique pair of Item_ID and Component_ID as the primary key.

Period

“Period” table holds the date information for each period. Period_ID is the primary key. Period_ID starts at 1 and increases one by one till there are enough periods to represent the time span we want to implement MRP.

Item_Period

“Item_Period” table has unique pairs of Item_ID and Period_ID for the primary key. This table hold the information of the gross requirements, scheduled receipts, etc. for each item and period.

MRP Algorithm

Update BOM Levels

The function “sql_update_bom_levels” calculates BOM levels for all items. This is done in a single SQL query that uses WITH RECURSIVE.

Calculate Planned Order Receipt for BOM Level and Period

The function “sql_calc_mrp_planned_order_receipt” calculates projected inventory, net requirement, and planned order receipt for a single period for all items in the given BOM level. This is done in a single SQL query.

Calculate Planned Order Releases for BOM Level

The function “sql_calc_mrp_planned_order_releases” calculates the planned order releases for all items in the given BOM level. This is done in a single SQL query.

Calculate MRP for BOM Level

The function “sql_calc_mrp” calculates the MRP table for all periods and all the items in the given BOM level.

First, this function retrieves the maximum period number.

```
for period_ID in range(1, period_max + 1):
    sql_calc_mrp_planned_order_receipt(bom_level, period_ID)

sql_calc_mrp_planned_order_releases(bom_level)
```

The periods must be updated by order since the projected inventory from the last period is used in the calculations for the current period.

After all the periods are updated, then, planned order releases are calculated.

Update Component Gross Requirements

The function “sql_calc_component_gross_requirement” calculates dependent gross requirements from the planned order releases of the parent of this item. This is done in a single SQL query.

Calculate All MRP Table

The function “calc_all_mrp_table” calculates all the MRP table.

First, the function retrieves the maximum BOM level.

```
for bom_level in range(bom_level_max + 1):
    sql_calc_mrp(bom_level)
    sql_calc_component_gross_requirement(bom_level)
```

The items in different BOM levels should be calculated by order since the planned order releases of the previous BOM levels affect the current BOM level items’ gross requirements.

The Results

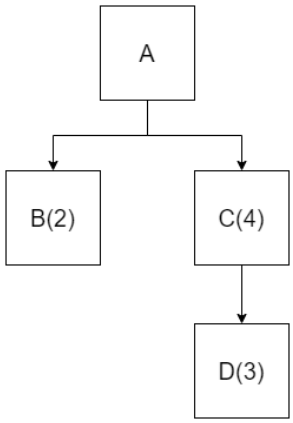
Given Data

Item_ID	Item_Name	Lot_Size	Lead_Time	Current_Inventory	BOM_Level
1	A	10	0	8	
2	B	25	3	50	
3	C	35	1	50	
4	D	100	1	200	

Item_ID	Component_ID	BOM_Multiplier
1	2	2
1	3	4
3	4	3

Item_ID	Period_ID	Gross_Requirement	Scheduled_Receipt	Projected_inventory	Net_Requirement	Planned_Order_Receipt	Planned_Order_Release
1	1	6	0				
1	2	4	10				
1	3	15	0				
1	4	9	0				
1	5	7	0				
2	1	6	25				
2	2	4	0				
2	3	22	0				
2	4	10	0				
2	5	3	0				
3	1	12	0				
3	2	1	0				
3	3	20	35				
3	4	30	0				
3	5	6	0				
4	1	15	0				
4	2	30	0				
4	3	50	0				
4	4	10	0				
4	5	0	0				

BOM Diagram



Project Calculation Results

	Item_ID	Period_ID	Gross_Requirement	Scheduled_Receipt	Projected_Inventory	Net_Requirement	Planned_Order_Receipt	Planned_Order_Release
0	1	1	6	0	2	0	0	0.0
1	1	2	4	10	8	0	0	0.0
2	1	3	15	0	3	7	10	10.0
3	1	4	9	0	4	6	10	10.0
4	1	5	7	0	7	3	10	10.0
5	2	1	6	25	69	0	0	25.0
6	2	2	4	0	65	0	0	25.0
7	2	3	42	0	23	0	0	NaN
8	2	4	30	0	18	7	25	NaN
9	2	5	23	0	20	5	25	NaN
10	3	1	12	0	38	0	0	0.0
11	3	2	1	0	37	0	0	0.0
12	3	3	60	35	12	0	0	58.0
13	3	4	70	0	0	58	58	46.0
14	3	5	46	0	0	46	46	NaN
15	4	1	15	0	185	0	0	0.0
16	4	2	30	0	155	0	0	100.0
17	4	3	224	0	31	69	100	117.0
18	4	4	148	0	0	117	117	0.0
19	4	5	0	0	0	0	0	NaN

Hand Calculations

	Item_ID	Period_ID	Gross_Requirement	Scheduled_Receipt	Projected_inventory	Net_Requirement	Planned_Order_Receipt	Planned_Order_Release
0	1	1	6	0	2	0	0	0
1	1	2	4	10	8	0	0	0
2	1	3	15	0	3	7	10	10
3	1	4	9	0	4	6	10	10
4	1	5	7	0	7	3	10	10
5	2	1	6	25	69	0	0	25
6	2	2	4	0	65	0	0	25
7	2	3	22	0	23	0	0	0
8	2	4	10	0	18	7	25	0
9	2	5	3	0	20	5	25	0
10	3	1	12	0	38	0	0	0
11	3	2	1	0	37	0	0	0
12	3	3	20	35	12	0	0	58
13	3	4	30	0	0	58	58	46
14	3	5	6	0	0	46	46	0
15	4	1	15	0	185	0	0	0
16	4	2	30	0	155	0	0	100
17	4	3	50	0	31	69	100	117
18	4	4	10	0	0	117	117	0
19	4	5	0	0	0	0	0	0

The results are correct. Test passes.