

프로젝트 기간  
2022-12-21 ~ 2022-12-27

## SMART FACTORY MINI PROJECT

팀명 : 코뿡  
팀원 : 이종원 최영재

Source - [https://github.com/JongWon112/SmartFactory\\_MiniPrj\\_FinalInspection.git](https://github.com/JongWon112/SmartFactory_MiniPrj_FinalInspection.git)

# 목차

1. 프로젝트 주제
2. 개발환경
3. 개발 산출물
4. 주요화면 및 코딩

# 1.프로젝트 주제

출하검사를 통한  
제품 이력 관리



## 2.개발환경



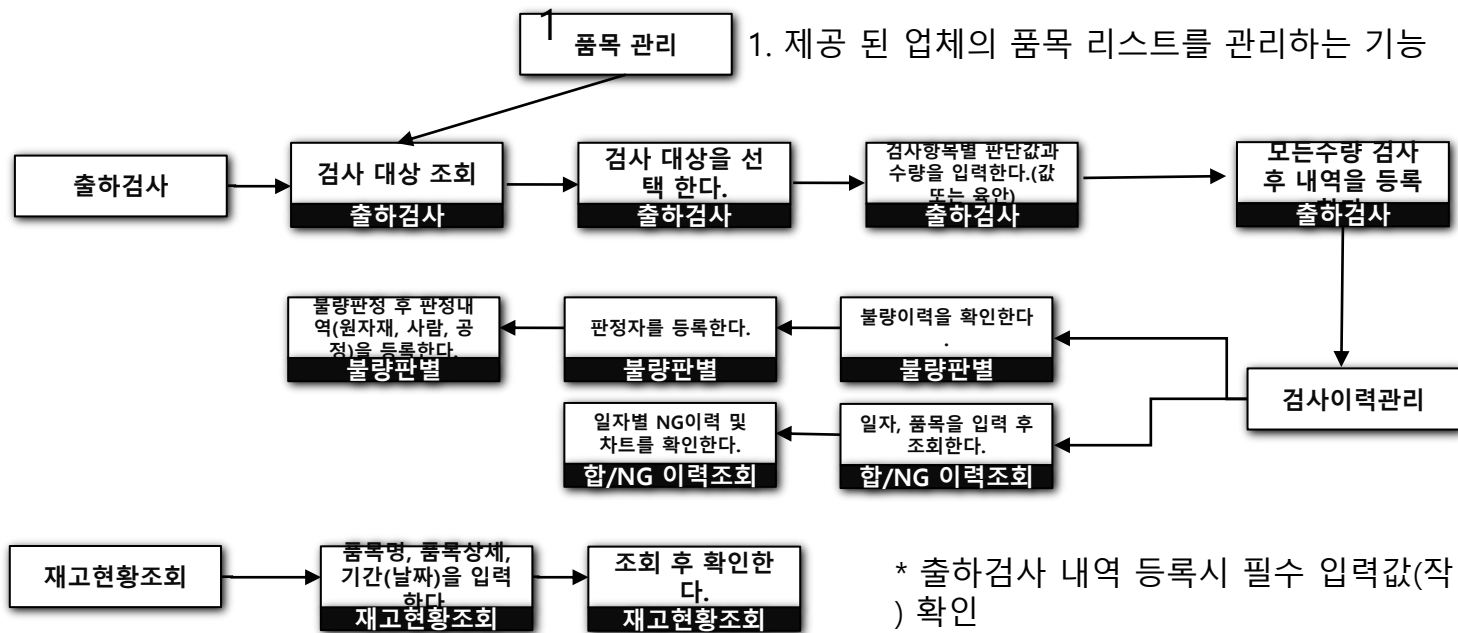
## 개발 산출물 - 일정 계획 관리

발 주 자		KDT_부산	프로젝트 명									작성일	2022-12-21	
Proj 명	제품 출하 검사 시스템											작성자	이종원, 최영재	
			YEAR	2022년										
항 목			MONTH	12									비고	
			DAY	21. PM	22. AM	22. PM	23 AM	23 PM	26 AM	26 PM	27 AM	27 PM		
PM 프로젝트 관리			가중치(%)											
1. 분석 및 설계			50											
1) 프로젝트 과제 선정			5											
2) 요구사항정의			10											
3) TO-BE 산출			5											
4) Application 설계 및 테이블 설계			30											
2. 구현			30											
1) 관리자 프로그램 구현			20											
5) 단위 테스트			10											
3. 테스트			20											
1) 가동 테스트			15											
2) 추가 요청사항 검토 및 적용			5											
TOTAL			100											

# 개발 산출물 - 요구사항 정의

요구사항 ID	요구사항명	상세내역	요청일	진행자	완료일
A-01	출하검사를 위한 현장 프로그램	검사항목 리스트를 기준으로 검사 내역을 등록한다.	2022-12-22	이종원	2022-12-26
		검사항목 리스트는 모든 품목에 대해 동일하게 적용한다.	2022-12-22	이종원	2022-12-26
		검사품목은 완제품(FERT)만을 검사품목으로 한다.	2022-12-22	이종원	2022-12-26
		양품여부는 육안 또는 값으로 작업자가 판단 후 총 검사 수량 중 양품인 제품의 수량, 불량품인 제품의 수량을 등록한다.	2022-12-22	이종원	2022-12-26
		검사 내역 중 NG 결과가 하나라도 있을경우 불량으로 판정한다		이종원	2022-12-26
		작업자, 검사수량, 양품수량, 불량품 수량 은 필수로 입력되어야 한다.	2022-12-22	이종원	2022-12-26
A-02	검사 이력 관리	현장에서 등록 한 불량미력을 확인		이종원	2022-12-27
		불량 내역을 바탕으로 불량판정(사람, 공정, 원자재)후 판정된 내역을 등록한다.	2022-12-22	이종원	2022-12-27
		불량판정기준, 판정자는 필수로 입력되어야 한다.	2022-12-22	이종원	2022-12-27
		합격/불량 이력 조회 (날짜, 검사코드, 품목), 불량 이력은 불량판정 기준이 확인 되어야 한다.	2022-12-22	이종원	2022-12-27
		일자별, 품목별 불량 내역에 대한 차트가 필요하다.	2022-12-22	이종원	2022-12-27
A-03	제품창고 입고이력 및 재고현황	날짜별 제품창고 입고 이력 조회(날짜, 품목명, 품목상세)한다.	2022-12-22	최영재	2022-12-23
		제품 창고의 재고 현황을 조회(품목명, 품목상세)한다.	2022-12-22	최영재	2022-12-23
		조회 품목으로 완제품(FERT)인것만 조회한다.	2022-12-22	최영재	2022-12-23
A-04	품목 관리	제공하는 품목의 리스트를 기준으로 품목 을 관리 한다.		이종원	2022-12-23

# 개발 산출물 - TO-BE Process



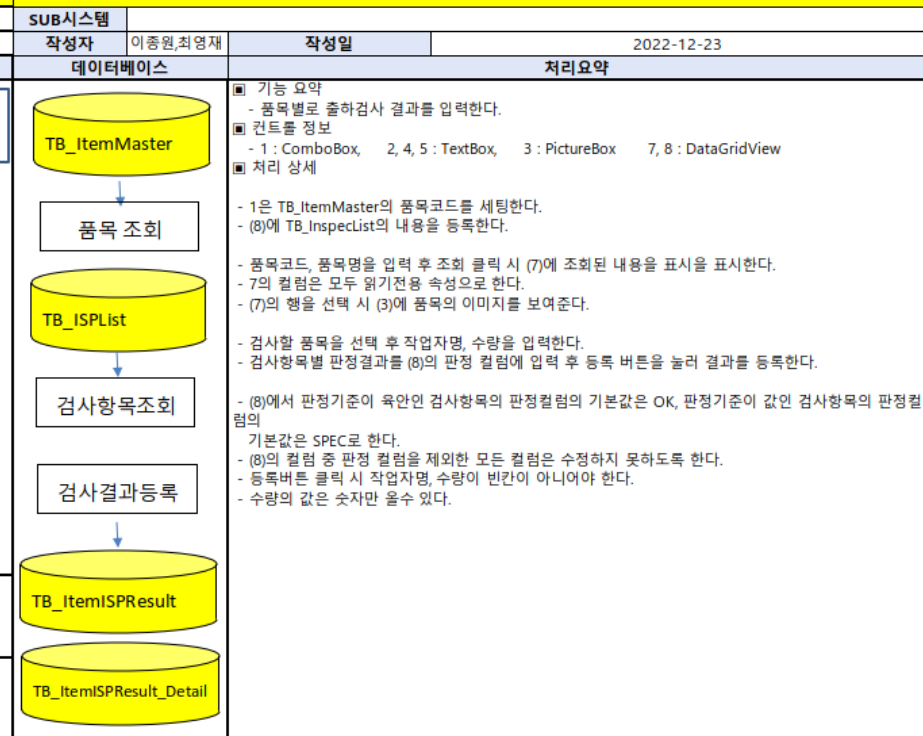
- \* 출하검사 내역 등록시 필수 입력값(작업자, 양품/NG, 수량) 확인
- \* 불량 판별시 필수 입력값(판정자, 판정내역) 확인
- \* 불량 판정 내역으로 원자재 / 사람 / 공정 으로 구분한다.

# 개발 산출물 - Application 설계서

입출력 레이아웃 (공통)

시스템		출하검사 시스템	
프로그램ID	VICE_ISP	프로그램명	출하검사 현장프로그램
화면 레이아웃			
<div> <div> <div>품목코드</div> <div>1</div> </div> <div> <div>품목명</div> <div>2</div> </div> <div> <div>조회</div> <div>등록</div> </div> </div> <div> <div> <div>품목코드</div> <div>품목명</div> <div>품목유형</div> </div> <div>7</div> </div> <div> <div>3</div> </div> <div> <div> <div>작업자명</div> <div>4</div> </div> <div> <div>수량</div> <div>5</div> </div> </div> <div> <div> <div>항목코드</div> <div>검사항목</div> <div>판정기준</div> <div>규격하한</div> <div>규격상한</div> <div>규격기준</div> <div>판정</div> </div> <div>8</div> </div>			
조회데이터블	TB_ItemMaster, TB_IspList		
갱신데이터블	TB_ItemISPResult, TB_ItemISPResult_Detail		

APPLICATION 설계





# 개발 산출물 - Application 설계서

입출력 레이아웃 (상세)

APPLICATION 설계

시스템	출하검사 시스템			SUB시스템			
프로그램ID		프로그램명	출하검사 현장프로그램	작성자	이종원,최영재	작성일	

처리요약

■ SELECT(TB\_ItemMaster)  
저장프로시저명 : SP\_ITEMMASTER\_S1

- ITEMCODE  
- ITEMNAME  
- ITEMTYPE

조회조건  
품목코드 - ITEMCODE  
품목명 - ITEMNAME

- 조회된 내용을 7에 등록한다.

검사품목조회

■ SELECT(TB\_ISPList)  
저장프로시저명 : SP\_ISPList\_S1  
- ISPCODE : 항목코드  
- ISPTYPE : 검사항목유형  
- ISPOINT : 판정기준  
- ISL : 규격하한  
- USL : 규격상한  
- SPEC : 규격기준

조회조건 : X

- 조회된 내용을 8에 등록한다.

검사항목조회

■ INSERT(TB\_ItemISPResult)  
저장프로시저명 : SP\_ItemISPResult\_I1

- SEQ : 검사순번  
- ISDATE : 검사일자  
- ITEMCODE : 품목코드  
- TRESULT : 종합판정결과  
- CHKER : 검사자  
- AMOUNT : 검사수량

- 등록버튼 클릭 시 실행

검사이력등록

■ INSERT(TB\_ItemISPResult\_Detail)  
저장프로시저명 : SP\_ItemISPResultDetail\_I1

- SEQ : 검사순번  
- ISPTYPE : 검사항목유형  
- IRESULT : 개별판정결과  
- IVALUE : 실제 기입값

- 등록버튼 클릭시 실행

검사이력 등록

저장프로시저명 : SP\_SET\_ITEMHOUSE

- ITEMCODE  
- AMOUNT  
- LASTINDATE  
- LASTCHKER

- 재고현황(TB\_ItemHouse)을 업데이트한다.

재고현황  
업데이트

# 개발 산출물 - Application 설계서

## 입출력 레이아웃 (공통)

시스템		출하검사 시스템	
프로그램ID	ISP_History	프로그램명	출하검사 이력관리

화면 레이아웃									
품목코드	1	품목명	2	검사기간	3	~	4		
귀책사유	7								

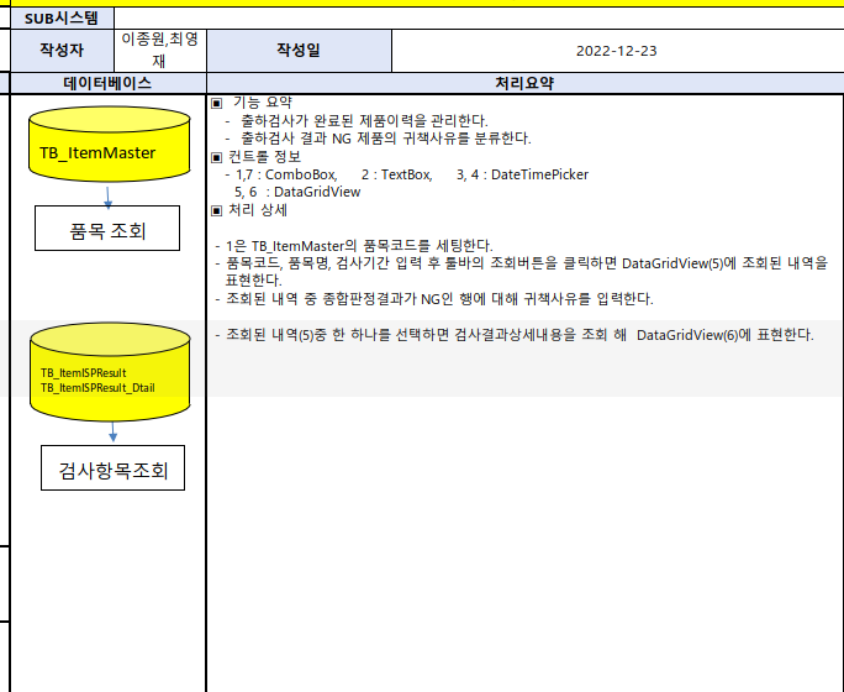
  

검사일	순번	품목코	품목명	수량	종합판정결	검사자	귀책사유	검사코드	검사항목	판정기준	LSL	USL	SPEC	기입값	개별판정결과
5							6								

조회테이블	TB_ItemMaster, TB_ItemISpResult, TB_ItemISpResult_Dtail
갱신테이블	TB_ItemISpResult_Dtail

## APPLICATION 설계



# 개발 산출물 - Application 설계서

입출력 레이아웃 (상세)

## APPLICATION 설계

시스템				SUB시스템			
프로그램ID		프로그램명		작성자		작성일	

## 처리요약

☒ SELECT(TB\_ItemSPResult)

저장프로시저명 : SP\_ItemSPResult\_S1

```

- ISPDATE      : 검사일자
- SEQ         : 순번
- ITEMCODE    : 품목코드
- ITEMNAME     : 품목명
- AMOUNT      : 수량
- RESULT      : 종합판정결과
- CHKER       : 검사자
- FREASON     : 귀책사유

조화조건
품목코드 - ITEMCODE
품목명   - ITEMNAME
검사기간 - ISPDATE

- 품목명은 ItemMaster 테이블과 JOIN

```

조회된 내용을 5에 등록한다

## 검사이력 해더 조회

☒ SELECT(TB\_ItemISPResult\_Detail)

저장프로시저명 : SP\_ItemSPResultDetail\_S1

- ISPCODE : 검사코드
- ISPTYPE : 검사항목
- ISPPPOINT : 판정기준
- LSL : LSL
- USL : USL
- SPEC : SPEC
- IVALUE : 실제 기입값
- IRESULT : 개별판정

- 테이블 5의 순번(SEQ)

-조회된 내용을 6에 등록한다.

## 검사이력 바디 조회

☐ UPDATE(TB\_ItemISPResult)

저장 프로시저명 : SP\_ItemISPRResult\_U1

- NGREASON : 귀책사유

수정조건  
- SEQ : 순번


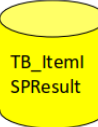
## 물량원의 등락

# 개발 산출물 - Application 설계서

## 입출력 레이아웃 (공통)

시스템	출하검사 시스템		
프로그램ID	Item_FReason	프로그램명	일자별 귀책사유별 불량현황
화면 레이아웃			
품목코드	1	검사기간	2 ~ 3
		귀책사유	4
5		6	
조회데이터	TB_ItemMaster, TB_ItemSPResult		
갱신데이터			

## APPLICATION 설계

SUB시스템			
작성자	이종원,최영재	작성일	2022-12-23
데이터베이스	처리요약		
 		<p>■ 기능 요약</p> <ul style="list-style-type: none"> <li>- 출하검사가 완료된 제품이력을 관리한다.</li> <li>- 출하검사 결과 NG 제품의 귀책사유를 분류한다.</li> </ul> <p>■ 컨트롤 정보</p> <ul style="list-style-type: none"> <li>- 1, 4 : ComboBox    3, 4 : DateTimePicker</li> <li>- 5, 6 : Chart</li> </ul> <p>■ 처리 상세</p> <ul style="list-style-type: none"> <li>- 품목코드, 검사기간, 귀책사유를 입력 후 톨바의 조회버튼을 클릭하면 검사기간동안 양품/NG 이력이 파이차트로 5에 표현되고 일자별 불량 내역 차트가 6에 조회된다.</li> </ul>	
일자별 귀책사유 조회			

# 개발 산출물 - Application 설계서

입출력 레이아웃 (상세)

APPLICATION 설계

시스템				SUB시스템			
프로그램ID		프로그램명		작성자		작성일	

처리요약

■ SELECT(TB\_ItemISPRResult)  
저장프로시저명 : SP\_ItemISPRResult\_PiChartData

- ISPDATE : 검사일자  
- ITEMCODE : 품목코드  
- ITEMNAME : 품목명  
- AMOUNT : 수량  
- TRESULT : 종합판정결과  
- FREASON : 귀책사유  
조회조건  
품목코드 - ITEMCODE  
검사기간 - ISPDATE

합격/NG 수량  
조회

■ SELECT(TB\_ItemISPRResult)  
저장프로시저명 : SP\_ItemISPRResult\_ColumnhartaData

- ISPDATE : 검사일자  
- ITEMCODE : 품목코드  
- ITEMNAME : 품목명  
- AMOUNT : 수량  
- TRESULT : 종합판정결과  
- FREASON : 귀책사유  
조회조건  
품목코드 - ITEMCODE  
검사기간 - ISPDATE  
귀책사유 - FREASON


일자별 불량  
조회

# 개발 산출물 - Application 설계서

## 입출력 레이아웃 (공통)

시스템		출하검사 시스템					
프로그램ID	ItemHouse		프로그램명	제품창고현황			
화면 레이아웃							
품목코드	1	품목명	2	입고일자	3	~	4
5	품목코드	품목명	입고수량	입고자	입고일자		
조회테이블			TB_ItemHouse				
갱신테이블							

## APPLICATION 설계

SUB시스템		2022-12-19	
작성자	이종원, 최영재	작성일	
데이터베이스		처리요약	
		<ul style="list-style-type: none"> <li>■ 기능 요약 <ul style="list-style-type: none"> <li>- 제품창고에 양품의 제품을 저장한다</li> </ul> </li> <li>■ 컨트롤 정보 <ul style="list-style-type: none"> <li>- 1 : ComboBox 2 : TextBox 3, 4 : DateTimePicker 5 : DataGridView</li> </ul> </li> <li>■ 처리 상세 <ul style="list-style-type: none"> <li>- 품목코드, 품목명, 입고일자를 입력 후 통바의 조회버튼을 클릭하면 DataGridView(5)에 조회된 내용을 표현한다.</li> </ul> </li> </ul>	

# 개발 산출물 - Application 설계서

입출력 레이아웃 (상세)					APPLICATION 설계			
시스템	EZ DEV				SUB시스템	기준정보		
프로그램ID	Form06_CustMaster	프로그램명	거래처마스터	작성자	동상현	작성일	2022-12-19	
처리요약								
<div>■ SELECT(TB_ItemHouse) 저장프로시저명 : SP_ItemHouse_S1  ITEMCODE : 품목코드 ITEMNAME : 품목명 AMOUNT INDATE CHKER  조회조건 품목코드 ITEMCODE 품목명 ITEMNAME LASTINDATE 입고일자</div> <div>제품 재고 조회</div>								

# 개발 산출물 - 테이블 기술서

테이블 기술서					테이블명	TB_ItemMaster
설 명	업체에서 제공된 품목정보				작성일	2022-12-22
No	Attribute	Data Type	NotNull	Key	Default	Description
1	ITEMCODE	VARCHAR(20)	O	PK		품목코드
2	ITEMNAME	VARCHAR(30)	O			품목명
3	ITEMTYPE	VARCHAR(5)	O			품목타입
4	ITEMIMAGE	image				품목 이미지
비고						

검사 품목 리스트

테이블 기술서					테이블명	TB_IspList
설 명	업체에서 제공된 출하검사항목 리스트				작성일	2022-12-22
No	Attribute	Data Type	NotNull	Key	Default	Description
1	ISPCODE	VARCHAR(20)	O	PK		검사항목코드
2	ISPTYPE	VARCHAR(20)	O			검사항목유형
3	ISPPPOINT	VARCHAR(10)	O			판정기준(값 OR 육안)
4	LSL	INT				규격하한
5	USL	INT				규격상한
6	SPEC	INT				표준규격
비고						

검사항목 리스트



# 개발 산출물 - 테이블 기술서

테이블 기술서					테이블명	TB_ItemISPResult
설 명	출하검사 결과테이블				작성일	2022-12-22
No	Attribute	Data Type	NotNull	Key	Default	Description
1	SEQ	INT	O	PK		검사순번
2	ISPDATE	DATETIME	O			검사일자
3	ITEMCODE	varchar(20)	O	FK		검사품목코드
4	TRESULT	VARCHAR(10)	O			종합판정(OK / NG)
5	CHKER	VARCHAR(10)	O			검사자
6	AMOUNT	INT	O			검사수량
7	FREASON	VARCHAR(15)				귀책사유(사람, 공정, 원자재)

검사결과 해더 테이블

테이블 기술서					테이블명	TB_ItemISPResult_Dtail
설 명	출하검사 상세결과테이블				작성일	2022-12-22
No	Attribute	Data Type	NotNull	Key	Default	Description
1	SEQ	INT	O	PK		검사순번
2	ISPCODE	VARCHAR(20)	O	PK, FK		검사항목유형
3	IRESULT	VARCHAR(10)	O			개별판정(OK / NG)
4	IVALUE	VARCHAR(10)				실제 기입 값

검사 결과 바디 테이블

# 개발 산출물 - 테이블 기술서

테이블 기술서					테이블명	TB_ItemHouse
설 명	제품창고 현황 테이블				작성일	2022-12-22
No	Attribute	Data Type	NotNull	Key	Default	Description
1	ITEMCODE	VARCHAR(20)	O	PK,FK		품목코드
3	AMOUNT	INT				품목수량
	LASTINDATE	DATE				입고일자
4	LASTCHKER	VARCHAR(10)				검사자(입고자)
비고						

제품 재고 테이블

# 개발 산출물 - 개발리스트

대분류	화면명	화면ID	개발자	SP	SP 설명	Table	CRUD
기준정보	출하검사 현장프로그램	VICE_JSP	이종원	SP_ITEMMASTER_S1	품목 리스트 조회	TB_ItemMaster	R
				SP_ISPList_S1	검사 항목 조회	TB_ISPList	R
				SP_ItemISpResult_I1	출하검사 결과 입력	TB_ItemISpResult	C
				SP_ItemISpResultDetail_I1	출하검사 결과 입력	TB_ItemISpResultDetail	C
				SP_SET_ITEMHOUSE	재고현황 업데이트	TB_ItemHouse	C,U
	출하검사 이력 조회	ISP_History	이종원	SP_ItemISpResult_S1	출하검사 이력 조회	TB_ItemMaster, TB_ItemISpResult	R
				SP_ItemISpResultDetail_S1	출하검사 이력 상세 조회	TB_ItemISpResult_Detail, TB_ISPList	R
				SP_ItemISpResult_U1	NG이력 별 귀책사유 입력	TB_ItemISpResult	U
	완제품 창고 현황	ItemHouse	최영재	SP_ItemHouse_S1	제품창고 현황 조회	TB_ItemHouse, TB_ItemMaster	R
	일자별 불량 현황	Item_FReason	이종원	SP_ItemISpResult_PiChartData	검색 기간동안 불량률 파이차트	TB_ItemISpResult	R
				SP_ItemISpResult_ColumnhartData	검색기간동안 일자별 귀책사유 현황	TB_ItemISpResult	R

# 주요화면 및 코딩 - 현장 프로그램

출하검사 현장프로그램

품목조회

품목코드 ALL

품목명 md

검사품목조회

품목코드	품목명	품목유형
64314-3X00...	MD REINF DA...	FERT
64314-3X00...	MD REINF DA...	FERT

검사

검사코드	검사항목	판정기준	규격하한	규격상한	표준규격	판정
IPT001	기포발생	육안				OK
IPT002	내폭	값	20	25	23	23
IPT003	비드 끊김	육안				NG
IPT004	비드이탈	육안				OK
IPT005	안착불량	값	2	6	4	4
IPT006	용접 위치	육안				OK
IPT007	토크값	값	30	50	40	40
IPT008	홀위치	육안				OK
IPT009	GAP 불량	값	1	3	2	2

작업자명 테스트

수량 200

등록

등록하였습니다

ItemMaster  
테이블에서 조회  
한 완제품 목록

작업자명, 수량 입력 후 등록

- TB\_ItemISPResult
- TB\_ItemISPResult\_Detail
- TB-ItemHouse

기본값 세팅, 판정기준에 따라 입력

# 주요화면 및 코딩 - 현장 프로그램

출하검사 결과 종합 검사 결과를 확인

- 개별검사 결과 NG가 하나라도 발생되면

종합결과 -> NG

DBHelper를 이용해서 프로시저 세팅  
및 실행

```
#region < 1. 출하검사 종합결과 저장 TB_ItemSPResult>
for (int i = 0; i < dgvISPList.Rows.Count; i++)
{
    //값입력인지 ok/ng 입력인지 확인
    if (dgvISPList.Rows[i].Cells["SPEC"].Value.ToString() == "") // ok / ng 입력
    {
        if (dgvISPList.Rows[i].Cells["VALUE"].Value.ToString().ToUpper() == "NG")
        {
            ISPchkValue = "NG";
            break;
        }
    }
    else // 값
    {
        inputValue = Convert.ToInt32(dgvISPList.Rows[i].Cells["VALUE"].Value);
        lsValue = Convert.ToInt32(dgvISPList.Rows[i].Cells["LSL"].Value);
        usValue = Convert.ToInt32(dgvISPList.Rows[i].Cells["USL"].Value);
        if (inputValue < lsValue || inputValue > usValue)
        {
            ISPchkValue = "NG";
            break;
        }
    }
}

//DBHelper를 이용한 프로시저 실행
helper.SetChangeSP("SP_ItemSPResult_11");
helper.SetChangeSP_Param("@ITEMCODE", iTemCode);
helper.SetChangeSP_Param("@TRESULT", ISPchkValue);
helper.SetChangeSP_Param("@CHKR", chkcr);
helper.SetChangeSP_Param("@AMOUNT", amount);

helper.ChangeSP_Run();
}
#endregion
```

```
참조 4개
public void Close()...

//조회용 저장 프로시저 세팅
참조 4개
public void SetSelectSP(string SP_Name)...
//조회용 저장 프로시저 파라미터 세팅
참조 13개
public void SetSelectSP_Param(string variableName, string value)...
//조회용 저장 프로시저 실행
참조 4개
public void SelectSP_Run(DataTable dtTemp)...

참조 2개
public void SelectSql(string sql, DataTable dtTemp)...
//저장용 저장 프로시저 세팅
참조 1개
public void SetChangeSP(string SP_Name)...
//저장용 저장 프로시저 파라미터 세팅
참조 2개
public void SetChangeSP_Param(string variableName, string value)...
//저장용 저장 프로시저 실행
참조 1개
public void ChangeSP_Run()...

//수정, 삭제, 등록 프로시저 변경 및 파라미터 초기화 -> 병합 트랜잭션 방지...
참조 1개
public void setSPNameNew(string spName)...
```

# 주요화면 및 코딩 - 현장 프로그램

```
#region < 2. 출하검사 건사항목별 결과 저장 TB_ItemSPResultDetail >
//검사 품목별 각각의 판단값 입력
for (int i = 0; i < dgvISPList.Rows.Count; i++)
{
    ispCode = dgvISPList.Rows[i].Cells["ISPCODE"].Value.ToString();
    //값입력인지 ok/ng입력인지 확인
    if (dgvISPList.Rows[i].Cells["SPEC"].Value.ToString() == "") // ok / ng 입력
    {
        iValue = dgvISPList.Rows[i].Cells["IVALUE"].Value.ToString().ToUpper();
        iResult = iValue;
    }
    else // 값 입력
    {
        inputValue = Convert.ToInt32(dgvISPList.Rows[i].Cells["IVALUE"].Value);
        lsIValue = Convert.ToInt32(dgvISPList.Rows[i].Cells["LSL"].Value);
        usIValue = Convert.ToInt32(dgvISPList.Rows[i].Cells["USL"].Value);
        iValue = Convert.ToString(inputValue);

        if (inputValue < lsIValue || inputValue > usIValue)
        {
            iResult = "OK";
        }
        else iResult = "NG";
    }
}

//SP 실행
//helper = new DBHelper();
helper.setSPNameNew("SP_ItemSPResultDetail_11");
helper.SetChangeSP_Param("@ISPCODE", ispCode);
helper.SetChangeSP_Param("@IRESULT", iResult);
helper.SetChangeSP_Param("@IVALUE", iValue);
helper.ChangeSP_Run();
}
```

출하검사 결과 개별 검사 결과를 등록

```
#region < 3. 출하검사 결과 OK인 제품 제품창고 테이블 저장 TB_ItemHouse >
//제고창고 insert or update
if (ISPChkValue == "OK")
{
    //helper = new DBHelper();
    helper.setSPNameNew("SP_SET_ITEMHOUSE");
    helper.SetChangeSP_Param("@ITEMCODE", itemCode);
    helper.SetChangeSP_Param("@AMOUNT", amount);
    helper.SetChangeSP_Param("@LASTCHKER", chker);
    helper.ChangeSP_Run();
}

#endregion

helper.sTran.Commit();
MessageBox.Show("등록하였습니다.");
```

출하검사 종합결과 OK 인 제품을  
제품창고 테이블에 등록

# 주요화면 및 코딩 - 현장 프로그램

```
--
-- Author:      이종원
-- Create date: 2022-12-27
-- Description: 출하검사 메인테이블 업데이트
--
ALTER PROCEDURE [dbo].[SP_ItemISPResult_I1]
    @ITEMCODE VARCHAR(20),
    @TRESULT VARCHAR(10),
    @CHKER VARCHAR(15),
    @AMOUNT INT,

    @LANG VARCHAR(10), -- 언어
    @RS_CODE VARCHAR(1) OUT, -- 조회결과를 리턴
    @RS_MSG VARCHAR(100) OUT -- 메시지 리턴
AS
BEGIN
    INSERT INTO TB_ItemISPResult VALUES
        (GETDATE(), @ITEMCODE, @TRESULT, @CHKER, @AMOUNT, NULL);
END

--
-- Author:      이종원
-- Create date: 2022-12-26
-- Description: 출하검사 개별결과값 입력
--
ALTER PROCEDURE [dbo].[SP_ItemISPResultDetail_I1]
    @ISPCODE VARCHAR(20),
    @IREULT VARCHAR(10),
    @IVALUE VARCHAR(10),

    @LANG VARCHAR(10), -- 언어
    @RS_CODE VARCHAR(1) OUT, -- 조회결과를 리턴
    @RS_MSG VARCHAR(100) OUT -- 메시지 리턴
AS
BEGIN
    --SEQ 값 체크
    DECLARE @SEQ INT
    SELECT @SEQ = MAX(SEQ)
    FROM TB_ItemISPResult

    -- INSERT
    INSERT INTO TB_ItemISPResult_Detail (SEQ, ISPCODE, IRESULT, IVALUE)
    VALUES (@SEQ, @ISPCODE, @IREULT, @IVALUE);
END
```

```
--
-- Author:      이종원
-- Create date: 2022-12-26
-- Description: 제품창고 현황 업데이트
--
ALTER PROCEDURE [dbo].[SP_SET_ITEMHOUSE]
    @ITEMCODE VARCHAR(20),
    @AMOUNT INT,
    @LASTCHKR VARCHAR(10),

    @LANG VARCHAR(10), -- 언어
    @RS_CODE VARCHAR(1) OUT, -- 조회결과를 리턴
    @RS_MSG VARCHAR(100) OUT -- 메시지 리턴
AS
BEGIN
    -- INSERT 할지 UPDATE 할지 결정
    DECLARE @FLAG INT
    SELECT @FLAG = COUNT(*)
    FROM TB_ItemHouse
    WHERE LASTINDATE = Convert(date, GETDATE())
    AND ITEMCODE = @ITEMCODE

    -- FLAG 값이 0이면 INSERT 1이면 UPDATE
    IF @FLAG = 0
    BEGIN
        -- INSERT
        INSERT INTO TB_ItemHouse (ITEMCODE, AMOUNT, LASTINDATE, LASTCHKR)
        VALUES (@ITEMCODE, @AMOUNT, GETDATE(), @LASTCHKR)
    END
    ELSE
    BEGIN
        --UPDATE
        UPDATE TB_ItemHouse
        SET AMOUNT = AMOUNT + @AMOUNT
        ,LASTCHKR = @LASTCHKR
        WHERE ITEMCODE = @ITEMCODE
    END
END
```

# 주요화면 및 코딩 - 출하검사 이력조회

체크박스 선택 시 귀책사유를 입력해야하는 데이터 표시

출하검사 이력 조회  
 품목조회

품목코드  품목명

☐ 귀책사유 미입력 데이터 조회

검사기간  ~

귀책사유 콤보박스

귀책사유

종합판정 결과 NG인 행의 귀책사유를 입력하는 콤보박스

검사일자	순번	품목코드	품목명	수량	종합판정 결과	검사자	귀책사유
2022-12-27 오후 12:37	85	64329-2W000E	REINF-DASH LWR SIDE, RH	123	NG	테스트	
2022-12-27 오후 12:12	84	64314-3X900E	MD REINF DASH PNL RHD	200	NG	테스트	
2022-12-27 오전 10:51	83	64331-2S500E	MBR ASSY-DASH CROSS	10	NG	테스트	공정
2022-12-27 오전 10:45	82	64319-2W000E	REINF-DASH LWR SIDE, LH	120	OK	관리자	
2022-12-27 오전 10:44	81	64319-2W000E	REINF-DASH LWR SIDE, LH	100	OK	관리자	
2022-12-27 오전 9:20	74	71618-1R000E	EXTN-QTR INR RR, LH	10	NG	코빙	공정
2022-12-27 오전 9:20	73	71157-D3000E	REINF-FR DR HINGE UPR, LH	20	NG	코빙	사람
2022-12-27 오전 9:19	72	71157-D3000E	REINF-FR DR HINGE UPR, LH	1000	OK	코빙	
2022-12-27 오전 9:19	71	71157-D3000E	REINF-FR DR HINGE UPR, LH	120	OK	코빙	
2022-12-27 오전 9:18	70	71157-D3000E	REINF-FR DR HINGE UPR, LH	120	OK	코빙	

검사코드	검사항목	ISPOINT	규격하한	규격상한	표준규격	실제 기입값	개별판정
IP T001	기포발생	육안				NG	NG
IP T002	내폭	값	20	25	23	23	NG
IP T003	비드 굵기	육안				OK	OK
IP T004	비드미탈	육안				OK	OK
IP T005	안착불량	값	2	6	4	4	NG
IP T006	용접 위치	육안				OK	OK
IP T007	토크값	값	30	50	40	40	NG
IP T008	홀위치	육안				OK	OK
IP T009	GAP 불량	값	1	3	2	2	NG

메인테이블에서 특정 행 선택 시 해당하는 상세 정보 표시



# 주요화면 및 코딩 - 출하검사 이력조회

참조 1개

```
private void dgvISPMMain_CellClick(object sender, DataGridViewCellEventArgs e)
{
    #region < 출하검사 서버 테이블 조회 >
    string seq = dgvISPMMain.CurrentRow.Cells["SEQ"].Value.ToString();

    //DB 객체 생성
    DBHelper helper = new DBHelper();
    //저장프로시저 세팅
    helper.SetSelectSP("SP_ItemISPSResultDetail_S1");
    //파라미터 세팅
    helper.SetSelectSP_Param("@SEQ", seq);
    //sp 실행
    DataTable dtTemp_Sub = new DataTable();
    helper.SelectSP_Run(dtTemp_Sub);

    //그리드에 바인딩
    dgvISPSSub.DataSource = dtTemp_Sub;
    #endregion
}
```

```
-- =====
-- Author:      이종원
-- Create date:  2022-12-23
-- Description:  출하검사 이력 바디조회
-- =====
ALTER PROCEDURE [dbo].[SP_ItemISPSResultDetail_S1]
    @SEQ INT, --순번

    @LANG          VARCHAR(10) ,      -- 언어
    @RS_CODE       VARCHAR(1) OUT,     -- 조회결과를 리턴
    @RS_MSG        VARCHAR(100) OUT   -- 메세지 리턴
AS
BEGIN
    SELECT B.ISPCODE AS ISPCODE
        ,B.ISPTYPE AS ISPTYPE
        ,B.ISPPPOINT AS ISPPPOINT
        ,B.LSL AS LSL
        ,B.USL AS USL
        ,B.SPEC AS SPEC
        ,A.IVALUE
        ,A.IRESULT
    FROM TB_ItemISPSResult_Detail A JOIN TB-ISPList B
        ON B.ISPCODE = A.ISPCODE
    WHERE A.SEQ = @SEQ
END
```

# 주요화면 및 코딩 - 출하검사 이력조회

```
#region < 귀책사유 저장 >
DataTable dtTemp = ((DataTable)dgvISPMain.DataSource).GetChanges();

if (dtTemp.Rows.Count == 0) return;

DBHelper helper = new DBHelper();
helper.SetChangeSP("SP_ItemISPResult_U1");
string seq = string.Empty;
string fReason = string.Empty;
try
{
    foreach (DataRow dr in dtTemp.Rows)
    {
        switch (dr.RowState)
        {
            case DataRowState.Modified:
                seq = dr["SEQ"].ToString();
                fReason = dr["FREASON"].ToString();
                helper.setSPNameNew ("SP_ItemISPResult_U1");
                helper.SetChangeSP_Param("@SEQ", seq);
                helper.SetChangeSP_Param("@FREASON", fReason);

                helper.ChangeSP_Run();
                break;
        }
    }

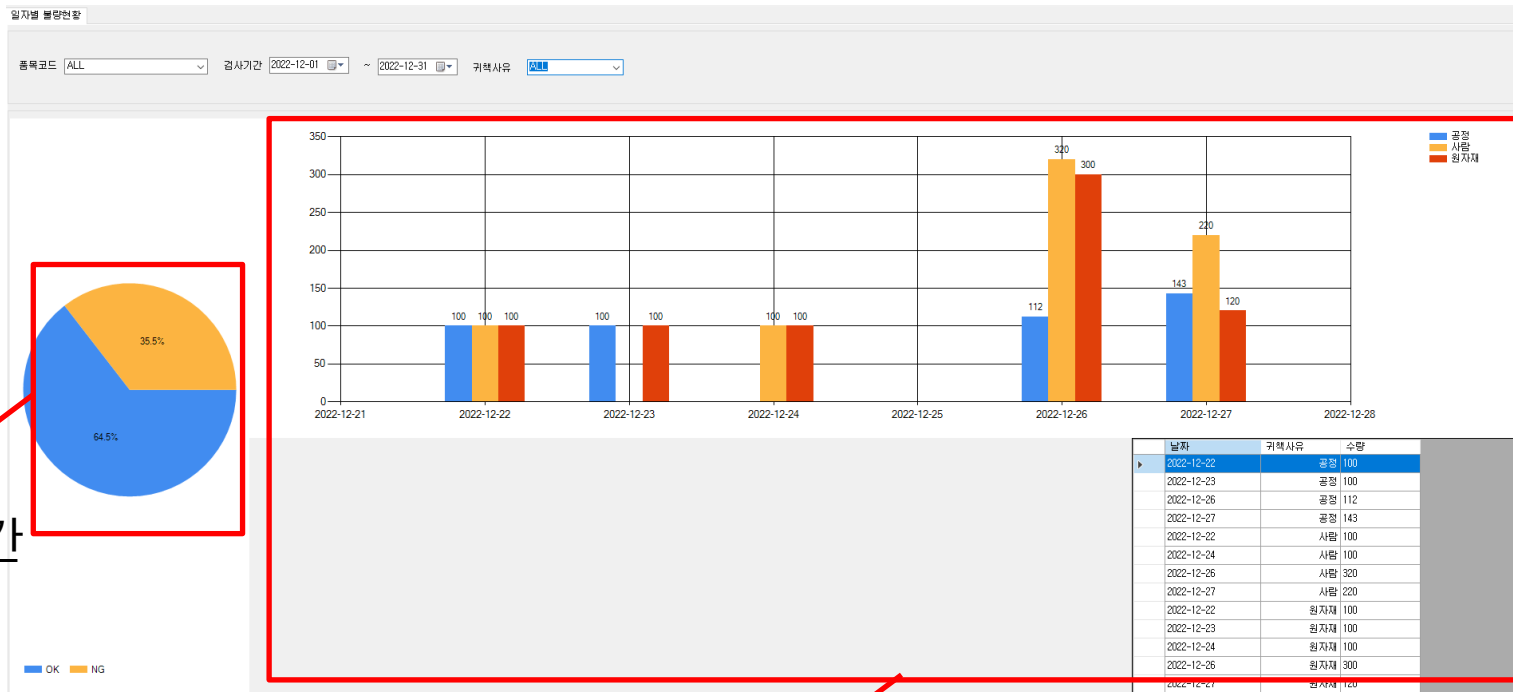
    helper.sTran.Commit();
    MessageBox.Show("저장하였습니다.");
}
}
```

```
-- =====
-- Author:      이종원
-- Create date: 2022-12-26
-- Description: 출하검사 귀책사유 업데이트
-- =====

ALTER PROCEDURE [dbo].[SP_ItemISPResult_U1]
    @SEQ INT,
    @FREASON VARCHAR(15),

    @LANG      VARCHAR(10),
    @RS_CODE   VARCHAR(1)   OUTPUT,
    @RS_MSG    VARCHAR(100) OUTPUT
AS
BEGIN
    UPDATE TB_ItemISPResult
        SET FREASON = @FREASON
    WHERE SEQ = @SEQ
END
```

# 주요화면 및 코딩 - 일자별 불량 현황



조회 기간 동안 일자별 불량 발생 건수

# 주요화면 및 코딩 - 일자별 불량 현황(파이차트)

#region < 조회기간동안 OK/NG 비율을 파이 차트로 표현 >

```
if (dtTemp.Rows.Count == 0)
{
    chart1.Series.Clear();
    return;
}
```

```
int cntOK = 0;
int cntNG = 0;
```

//해당기간동안 OK, NG 수량 합

```
foreach(DataRow dr in dtTemp.Rows)
{
    if(dr["TRESULT"].ToString() == "OK")
    {
        cntOK += Convert.ToInt32(dr["ITEMCNT"].ToString());
    }
    else
    {
        cntNG += Convert.ToInt32(dr["ITEMCNT"].ToString());
    }
}
```

//chart1에 파이차트 세팅

```
chart1.Series.Clear();
Series series = new Series();
series.ChartType = SeriesChartType.Pie;
```

```
chart1.Series.Add(series);
```

```
chart1.Legends[0].Docking = System.Windows.Forms.DataVisualization.Charting.Docking.Bottom;
```

//파이차트를 ok/ng 비율로 표시

```
chart1.Series[0].Label = "#PERCENT(P1)";
string[] Legend = { "OK", "NG" };
```

//ok/ng 수량이 0일 경우 처리

```
if(cntOK == 0)
```

```
{
    chart1.Series[0].Points.Add(cntNG);
    chart1.Series[0].Points[0].LegendText = Legend[1];
}
```

```
else if(cntNG == 0)
```

```
{
    chart1.Series[0].Points.Add(cntOK);
    chart1.Series[0].Points[0].LegendText = Legend[0];
}
```

```
else
```

```
{
```

```
    chart1.Series[0].Points.Add(cntOK);
    chart1.Series[0].Points[0].LegendText = Legend[0];
    chart1.Series[0].Points.Add(cntNG);
    chart1.Series[0].Points[1].LegendText = Legend[1];
}
```

## 주요화면 및 코딩 - 일자별 불량 현황(막대차트)

```
if (Convert.ToString(cboFReason.SelectedValue) == "") //ALL 선택시
{
    #region < 다중 막대 차트 >
    chart2.DataBindCrossTable(dtTemp2.AsEnumerable(), "FREASON", "DATE", "CNT", "");
    for (int i = 0; i < chart2.Series.Count; i++)
    {
        chart2.Series[i].IsVisibleInLegend = true; // 범례 표시
        chart2.Series[i].LegendText = chart2.Series[i].ToString().Split('-')[1].ToString(); // 범례
        chart2.Series[i].SetCustomProperty("PixelPointWidth", "100"); //막대의 폭
        chart2.Series[i].SetCustomProperty("PointWidth", "0");
        //
        //chart2.Series[i].Label = chart2.Series[i].ToString().Split('-')[1].ToString();

        chart2.Series[i].IsValueShownAsLabel = true; // 차트에 값 표시
    }
    #endregion
}
else // 개별 선택시
{
    chart2.DataBindTable(dtTemp2.DefaultView, "DATE");
    chart2.Series[0].Name = Convert.ToString(dtTemp2.Rows[0]["FREASON"]);
    //chart2.Series[1].Name = Convert.ToString(dtTemp2.Rows[1]["FREASON"]);
    //chart2.Series[2].Name = Convert.ToString(dtTemp2.Rows[2]["FREASON"]);
}
```

# 주요화면 및 코딩 - 일자별 불량 현황

```
ALTER PROCEDURE [dbo].[SP_ItemISResult_PiChartData]
    @ITEMCODE VARCHAR(20),
    @SDATE DATE,
    @EDATE DATE,

    @LANG VARCHAR(10), -- 언어
    @RS_CODE VARCHAR(1) OUT, -- 조회결과를 리턴
    @RS_MSG VARCHAR(100) OUT -- 메시지 리턴
AS
BEGIN
    SELECT A.ISPDATE
    ,A.ITEMCODE
    ,A.TRESULT
    ,SUM(A.SUM_CNT) AS ITEMCNT
    FROM
    (
        select CONVERT(DATE,ISPDATE) AS ISPDATE
        ,ITEMCODE
        ,TRESULT
        ,SUM(AMOUNT) AS SUM_CNT
        from TB_ItemISResult
        GROUP BY ITEMCODE, TRESULT, ISPDATE
    ) A
    WHERE CONVERT(DATE, A.ISPDATE) BETWEEN @SDATE AND @EDATE
    AND A.ITEMCODE LIKE '%' + @ITEMCODE + '%'
    GROUP BY ISPDATE, ITEMCODE, TRESULT
END
```

```
ALTER PROCEDURE [dbo].[SP_ItemISResult_ColumnhartData]
    @ITEMCODE VARCHAR(20),
    @SDATE DATE,
    @EDATE DATE,
    @FREASON VARCHAR(15),

    @LANG VARCHAR(10), -- 언어
    @RS_CODE VARCHAR(1) OUT, -- 조회결과를 리턴
    @RS_MSG VARCHAR(100) OUT -- 메시지 리턴
AS
BEGIN
    SELECT DATE
    ,FREASON
    ,SUM(AMOUNT) AS CNT
    FROM (
        SELECT CONVERT(DATE, ISPDATE) AS DATE
        ,ITEMCODE
        ,TRESULT
        ,AMOUNT
        ,FREASON
        FROM TB_ItemISResult
        WHERE FREASON LIKE '%' + @FREASON + '%'
        AND ITEMCODE LIKE '%' + @ITEMCODE + '%'
    ) A
    WHERE DATE BETWEEN @SDATE AND @EDATE
    GROUP BY DATE, FREASON
END
```

# 주요화면 및 코딩 - 제품 참고 현황

제품참고 현황

품목코드: ALL    품목명:    날짜: 2022-12-01 ~ 2022-12-31

	품목코드	품목명	수량	입고일자	마지막입고자
▶	64314-3X000E	MD REINF DA...	120	2022-12-27	코병
	64314-3X900E	MD REINF DA...	120	2022-12-27	코병
	64319-2W000E	REINF-DASH ...	220	2022-12-27	관리자1
	64355-D3000E	REINF DASH L...	50	2022-12-26	검사자3
	64395-B3000E	MBR-DASH C...	150	2022-12-26	검사자2
	71157-D3000E	REINF-FR DR ...	1240	2022-12-27	코병
	71167-D3000E	REINF-FR DR ...	120	2022-12-27	코병
	71178-D3000E	BRKT-FR PLR...	200	2022-12-26	ㅋㅋㅋ
	71189-D3000E	BRKT-FR PLR...	300	2022-12-26	호호호
	71618-1R000E	EXTN-QTR IN...	120	2022-12-27	코병
	71628-1R000E	EXTN-QTR IN...	120	2022-12-26	검사자1
	71687-B8000E	A/CON MTG RH	800	2022-12-26	호호호

TB\_ITEMHOUSE 테이블의 제품별  
재고 수량을 조회

```
ALTER PROCEDURE [dbo].[SP_ItemHouse_S1]

    @ITEMCODE    VARCHAR(30), -- 품목코드
    @ITEMNAME     VARCHAR(30), -- 품목명
    @SINDATE      DATE, -- 조회시작날짜
    @EINDATE      DATE, -- 조회마감날짜

    @LANG         VARCHAR(10),
    @RS_CODE      VARCHAR(1) OUTPUT,
    @RS_MSG       VARCHAR(100) OUTPUT
AS
BEGIN
    SELECT A.ITEMCODE
           ,B.ITEMNAME
           ,A.AMOUNT
           ,A.LASTINDATE
           ,A.LASTCHKR

    FROM TB_ItemHouse A JOIN TB_ItemMaster B
        ON A.ITEMCODE= B.ITEMCODE
    WHERE A.ITEMCODE LIKE '%' + @ITEMCODE + '%'
    AND B.ITEMNAME LIKE '%' + @ITEMNAME + '%'
    AND A.LASTINDATE BETWEEN @SINDATE AND @EINDATE

END
```

Q & A