

한국스마트정보교육원

39 기 1 팀 프로젝트 완료 보고서

프로젝트명
의류제조업 품질관리를 강화한 스마트팩토리 생산관리 시스템

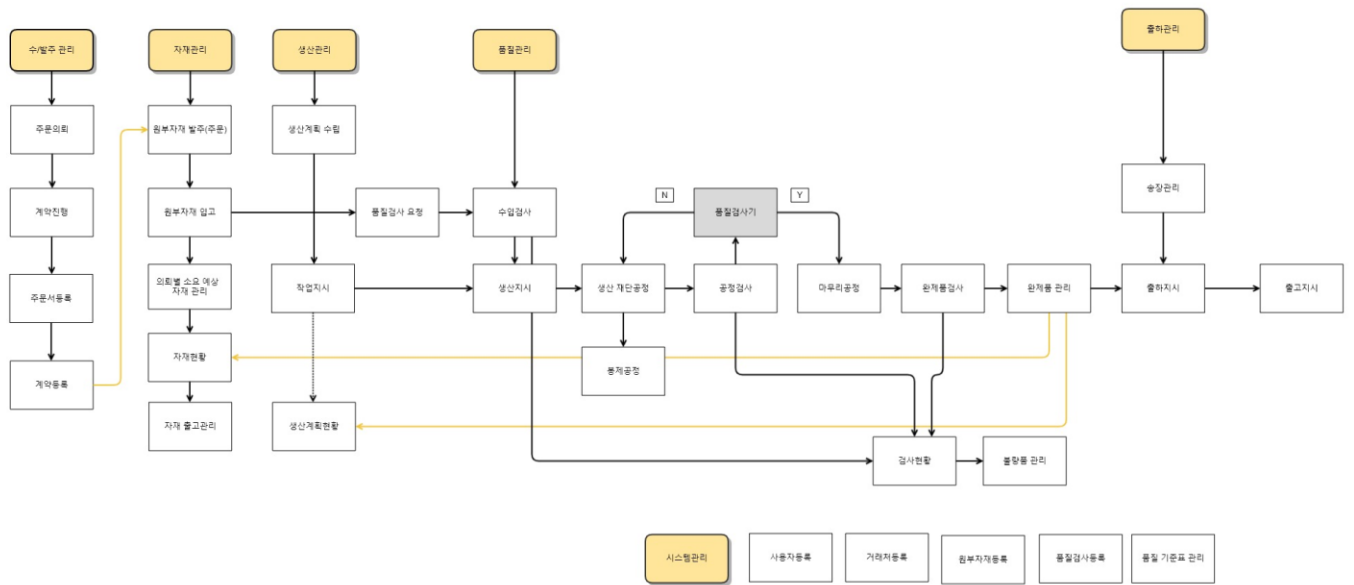
팀 원	박민아
	김다미
	이한빛
	한보람
작성일	2021 년 7 월 27 일

호스팅	http://sodak0127.cafe24.com/
GitHub	https://github.com/mina0596/clothing_smartFactory_ME_S

목차
1 . 프로젝트 기본정보
1.1 시스템 흐름도
2 . 기능적 요구사항 정의서
3 . 업무분담(담당업무)
4 . 설계사양서
4.1 ERD 물리모델
4.2 테이블 구조
4.3 네이밍 규칙
4.4 디렉토리 구조
4.5 패키지 구조
5 . 개발환경
6. 어플리케이션
6.1 어플리케이션 프로시저
6.2 서버 및 기타 설정
6.3 주요 기능별 정리

1. 프로젝트 기본정보

프로젝트명	의류제조업 품질관리를 강화한 스마트팩토리 생산관리 시스템	
개발기간	■ 2021 년 05 월 26 일 - 2021 년 07 월 26 일 (80 일)	
	2021 년 05 월 02 일 ~ 2021 년 05 월 14 일 (12 일)	1 프로젝트 주제 선정 및 사전조사
	2021 년 05 월 15 일 ~ 2021 년 06 월 08 일 (24 일)	1 시스템 구조도 작성 1 설계자료 취합, 수정 및 보완 1 ERD 작업 및 메서드, 패키지 정리
	2021 년 06 월 09 일 ~ 2021 년 06 월 11 일 (2 일)	1 서버 설정 1 개발환경 설정 1 Hosting 설정
	2021 년 06 월 12 일 ~ 2021 년 07 월 18 일 (36 일)	1 기능 구현(개인 분담 프로그래밍)
	2021 년 07 월 19 일 ~ 2021 년 07 월 22 일 (3 일)	1 테스트, 오류 수정 및 보완 1 미완성 기능 구현
	2021 년 07 월 23 일 ~ 2021 년 07 월 26 일 (3 일)	1 완료보고서 작성
개발목적	<p>한국형 봉제공장 스마트팩토리 상용화가 활발해짐에 따라 이에 부합한 전사관리시스템을 구축할 목적이다.</p> <p>완전자동화 시스템의 최대장점인 생산성과 품질향상에 기여하는 QC 강화 MES 시스템을 구축함으로써 기업경영의 경쟁력을 향상시킨다.</p> <p>기존 전수검사보다 빠른 속도로 검사를 실행하여 한 개의 불량품도 납품되지 않게 관리할 수 있는 시스템을 구축한다.</p>	



2. 요구사항 정의서

2.1 기능적 요구사항

No.	기능적 요구사항	관리자	본사사원	공장사원
1	주문서 관리			
	주문서 등록	주문서 등록	주문서 등록	
	주문서 수정	주문서 수정	주문서 수정	
	주문서 삭제	주문서 삭제	주문서 삭제	
	주문서 전체 조회	주문서 전체 조회	주문서 전체 조회	주문서 전체 조회
	상세 주문서 조회	상세 주문서 조회	상세 주문서 조회	상세 주문서 조회
	승인 완료 주문서 조회	승인 완료 주문서 조회	승인 완료 주문서 조회	승인 완료 주문서 조회
2	수주 계약 관리			
	수주 계약 등록	수주 계약 등록	수주 계약 등록	
	수주 계약 수정	수주 계약 수정	수주 계약 수정	
	수주 계약 삭제	수주 계약 삭제	수주 계약 삭제	
	수주 계약 전체 조회	수주 계약 전체 조회	수주 계약 전체 조회	수주 계약 전체 조회
	수주 계약 상세 조회	수주 계약 상세 조회	수주 계약 상세 조회	수주 계약 상세 조회
3	발주 요청 관리			
	발주 요청 등록	발주 요청 등록		발주 요청 등록
	발주 요청 수정	발주 요청 수정		발주 요청 수정
	발주 요청 삭제	발주 요청 삭제		발주 요청 삭제
	발주 요청 전체 조회	발주 요청 전체 조회	발주 요청 전체 조회	발주 요청 전체 조회
4	발주 계약 관리			
	발주 계약 등록	발주 계약 등록		발주 계약 등록
	발주 계약 수정	발주 계약 수정		발주 계약 수정

	발주 계약 삭제	발주 계약 삭제		발주 계약 삭제
	발주 계약 전체 조회	발주 계약 전체 조회	발주 계약 전체 조회	발주 계약 전체 조회
5	자재 입고 관리			
	자재 입고 등록	자재 입고 등록		자재 입고 등록
	자재 입고 수정	자재 입고 수정		자재 입고 수정
	자재 입고 삭제	자재 입고 삭제		자재 입고 삭제
	자재 입고 현황	자재 입고 현황	자재 입고 현황	자재 입고 현황
6	자재 출고 관리			
	자재 출고 등록	자재 출고 등록		자재 출고 등록
	자재 출고 수정	자재 출고 수정		자재 출고 수정
	자재 출고 삭제	자재 출고 삭제		자재 출고 삭제
	자재 출고 전체 조회	자재 출고 전체 조회	자재 출고 전체 조회	자재 출고 전체 조회
7	자재 현황 관리			
	원부자재 현 재고 현황 조회	원부자재 현 재고 현황 조회	원부자재 현 재고 현황 조회	원부자재 현 재고 현황 조회
	원부자재 입고 출고 종합 조회	원부자재 입고 출고 종합 조회	원부자재 입고 출고 종합 조회	원부자재 입고 출고 종합 조회
	입고 출고 정보 수정	입고 출고 정보 수정		입고 출고 정보 수정
	입고 출고 정보 등록	입고 출고 정보 등록		입고 출고 정보 등록
8	생산 계획 관리			
	생산 계획 등록	생산 계획 등록	생산 계획 등록	
	생산 계획 수정	생산 계획 수정	생산 계획 수정	
	생산 계획 삭제	생산 계획 삭제	생산 계획 삭제	
	주간별 생산 계획 조회	주간별 생산 계획 조회	주간별 생산 계획 조회	주간별 생산 계획 조회

	월별 생산 계획 조회	월별 생산 계획 조회	월별 생산 계획 조회	월별 생산 계획 조회
	생산 계획 전체 조회	생산 계획 전체 조회	생산 계획 전체 조회	생산 계획 전체 조회
9 생산 작업 지시 관리				
	생산 작업 지시 등록	생산 작업 지시 등록	생산 작업 지시 등록	
	생산 작업 지시 수정	생산 작업 지시 수정	생산 작업 지시 수정	
	생산 작업 지시 삭제	생산 작업 지시 삭제	생산 작업 지시 삭제	
	생산 작업 지시 조회	생산 작업 지시 조회	생산 작업 지시 조회	생산 작업 지시 조회
10 생산 지시 관리				
	생산 대기중 의뢰품목 조회	생산 대기중 의뢰품목 조회	생산 대기중 의뢰품목 조회	생산 대기중 의뢰품목 조회
	작업 지시별 최초 생산 시작	작업 지시별 최초 생산 시작		작업 지시별 최초 생산 시작
	의뢰 품목별 생산현황 조회	의뢰 품목별 생산현황 조회	의뢰 품목별 생산현황 조회	의뢰 품목별 생산현황 조회
	품목별 생산 공정별 생산 시작	품목별 생산 공정별 생산 시작		품목별 생산 공정별 생산 시작
	품목별 생산 공정별 생산 완료	품목별 생산 공정별 생산 완료		품목별 생산 공정별 생산 완료
11 생산 현황 관리				
	생산 계획별 생산현황 조회	생산 계획별 생산현황 조회	생산 계획별 생산현황 조회	생산 계획별 생산현황 조회
	작업 지시별 생산현황 조회	작업 지시별 생산현황 조회	작업 지시별 생산현황 조회	작업 지시별 생산현황 조회
	의뢰 품목별 생산현황 조회	의뢰 품목별 생산현황 조회	의뢰 품목별 생산현황 조회	의뢰 품목별 생산현황 조회
	품목별 생산현황 조회	품목별 생산현황 조회	품목별 생산현황 조회	품목별 생산현황 조회
12 완제품 관리				
	완제품 조회		공통농기계 조회	공통농기계 조회
12 품질 검사 기준표 관리				
	품질검사 기준표 등록	품질검사 기준표 등록	품질검사 기준표 등록	

	품질검사 기준표 수정	품질검사 기준표 수정	품질검사 기준표 수정	
	품질검사 기준표 삭제	품질검사 기준표 삭제	품질검사 기준표 삭제	
	품질검사 기준표 조회	품질검사 기준표 조회	품질검사 기준표 조회	품질검사 기준표 조회
13	품질 검사 요청 관리			
	품질 검사 요청 등록	품질 검사 요청 등록		품질 검사 요청 등록
	품질 검사 요청 수정	품질 검사 요청 수정		품질 검사 요청 수정
	품질 검사 요청 삭제	품질 검사 요청 삭제		품질 검사 요청 삭제
	품질 검사 요청 조회	품질 검사 요청 조회	품질 검사 요청 조회	품질 검사 요청 조회
14	품질 검사 측정값 관리			
	상세 품질 검사 결과 등록	품질 검사 결과 등록		품질 검사 결과 등록
	상세 품질 검사 결과 수정	품질 검사 결과 수정		품질 검사 결과 수정
	상세 품질 검사 결과 삭제	품질 검사 결과 삭제		품질 검사 결과 삭제
	상세 품질 검사 결과 조회	상세 품질 검사 결과 조회	상세 품질 검사 결과 조회	상세 품질 검사 결과 조회
	품질 검사 최종 결과 조회	품질 검사 최종 결과 조회	품질 검사 최종 결과 조회	품질 검사 최종 결과 조회
	품질 검사 성적서 조회	품질 검사 성적서 조회	품질 검사 성적서 조회	품질 검사 성적서 조회
15	품질 검사 현황 관리			
	실시간 검사현황 조회	실시간 검사현황 조회	실시간 검사현황 조회	실시간 검사현황 조회
	계약별 검사 현황 조회	계약별 검사 현황 조회	계약별 검사 현황 조회	계약별 검사 현황 조회
	품질 검사별 합격률 조회	품질 검사별 합격률 조회	품질 검사별 합격률 조회	품질 검사별 합격률 조회
	품목별 합격률 조회	품목별 합격률 조회	품목별 합격률 조회	품목별 합격률 조회
16	불량품 관리			
	불량품 등록	불량품 등록		불량품 등록
	불량품 수정	불량품 수정		불량품 수정

	불량품 삭제	불량품 삭제		불량품 삭제
	불량품별 처리 조회	불량품별 처리 조회	불량품별 처리 조회	불량품별 처리 조회
17 출하 지시 관리				
	출하 지시 등록	출하 지시 등록	출하 지시 등록	
	출하 지시 수정	출하 지시 수정	출하 지시 수정	
	출하 지시 삭제	출하 지시 삭제	출하 지시 삭제	
	출하 지시 조회	출하 지시 조회	출하 지시 조회	출하 지시 조회
18 송장 관리				
	송장 등록	송장 등록	송장 등록	
	송장 수정	송장 수정	송장 수정	
	송장 삭제	송장 삭제	송장 삭제	
	송장 조회	송장 조회	송장 조회	송장 조회
19 출고 관리				
	출고 정보 등록	출고 정보 등록	출고 정보 등록	
	출고 정보 수정	출고 정보 수정	출고 정보 수정	
	출고 정보 삭제	출고 정보 삭제	출고 정보 삭제	
	출고 정보 조회	출고 정보 조회	출고 정보 조회	출고 정보 조회
20 기초 정보 관리				
	사용자 등록	사용자 등록		
	사용자 수정	사용자 수정		
	사용자 삭제	사용자 삭제		
	개인 사용자 조회	개인 사용자 조회	개인 사용자 조회	개인 사용자 조회
	거래처 등록	거래처 등록	거래처 등록	
	거래처 수정	거래처 수정	거래처 수정	

	거래처 삭제	거래처 삭제	거래처 삭제	
	거래처 조회	거래처 조회	거래처 조회	거래처 조회
	원부자재 등록	원부자재 등록		원부자재 등록
	원부자재 수정	원부자재 수정		원부자재 수정
	원부자재 삭제	원부자재 삭제		원부자재 삭제
	원부자재 조회	원부자재 조회	원부자재 조회	원부자재 조회
	품질 검사 등록	품질 검사 등록	품질 검사 등록	
	품질 검사 수정	품질 검사 수정	품질 검사 수정	
	품질 검사 삭제	품질 검사 삭제	품질 검사 삭제	
	품질 검사 조회	품질 검사 조회	품질 검사 조회	품질 검사 조회
	생산 공정 등록	생산 공정 등록		생산 공정 등록
	생산 공정 수정	생산 공정 수정		생산 공정 수정
	생산 공정 삭제	생산 공정 삭제		생산 공정 삭제
	생산 공정 조회	생산 공정 조회	생산 공정 조회	생산 공정 조회

3. 업무분담(담당 업무)

이 름	담당 업무	업무 내용
박민아	팀장	회의 진행, 일정 관리, 업무 조율
	Hosting	배포 업무 수행
	자재현황	계산된 실시간 재고 현황, 자재 입고/출고 종합 목록
	생산현황	완제품관리, 작업지시, 생산지시, 생산계획별 / 품목별 생산현황 화면 및 기능 구현
	품질검사 합격률	품질검사별 합격률 기간별 조회 화면 및 기능 구현
	출고 관리	출고 제품 CRUD 및 화면구현
	자재 기초 정보, 생산 공정 정보	원부자재, 생산공정 정보 CRUD 및 화면구현
	권한별 로그인	권한별 기능 접근 interceptor 구현
한보람	Hosting	배포 업무 수행
	발주관리	원부자재 발주 주문, 계약 CRUD 및 화면 구현
	품질 검사 현황	의뢰 품목별 품질검사 성적서 조회 / 계약별 품질검사 현황 조회 구현
	출하 관리	출하지시, 송장 발행 CRUD 및 화면 구현
	품질 검사	품질 검사 종류 CRUD 및 화면 구현
	팀 프로젝트 문서 관리	회의 내용 정리, 관리 및 문서 작성
이한빛	수주 관리	주문서 CRUD + 다수의 승인된 의뢰의 수주계약 CRUD 및 화면구현 / 상세 주문서 조회

	자재 출고	자재출고 CRUD 및 화면구현
	생산 현황	작업지시별, 의뢰별 생산 상세 현황 기능 및 화면구현
	불량품관리	불량품 CRUD 및 화면 구현
	사용자, 거래처관리	기초정보관리의 사용자, 거래처 정보 CRUD 및 화면 구현
	페이지 넘버링	페이지 내용물의 페이지징 구현
	Hosting	배포 업무 수행
김다미	자재 입고	원부자재 자재입고 CRUD 및 화면 구현
	생산 계획	생산계획의 CRUD / 주간별, 월별 전체 생산계획 리스트와 달력으로 조회 화면구현
	품질 검사 요청	품질 검사 요청 CRUD 및 화면 구현
	품질 검사 측정값	요청된 품질 검사 중 실시된 검사 측정값을 다수로 입력 + 합격과 불합격 자동 판정되는 CRUD 및 화면 구현
	실시간 품질 검사 현황	실시간으로 품질 검사 요청 건수에 대한 현황 조회 가능한 화면 구현
	CSS	로그인 화면 구현 / 프로젝트 Bootstrap 활용 템플릿 적용 및 모듈 분리
	Hosting	배포 업무 수행

4. 설계사양서

4.1 ERD 물리 & 논리 모드



4.2 테이블 구조

ERP 접근 권한(erp_access_level)			
ERP 접근 레벨 번호(PK)	level_num	INT(5)	NOT NULL
ERP 접근 레벨명	level_name	VARCHAR(10)	NOT NULL

모든사원 목록(employee_info)			
사원 코드(PK)	employee_code	VARCHAR(50)	NOT NULL
ERP 접근 레벨 번호 (FK)	level_num	INT(5)	NOT NULL
사원 ERP 로그인 아이디	employee_id	VARCHAR(50)	NOT NULL
사원 비밀번호	employee_pw	VARCHAR(50)	NOT NULL
사원명	employee_name	VARCHAR(50)	NOT NULL

사원 연락처	employee_tel	VARCHAR(50)	NOT NULL
사원 주소	employee_address	VARCHAR(50)	NOT NULL
사원 이메일	employee_email	VARCHAR(50)	NOT NULL
사원 부서	employee_department	VARCHAR(50)	NOT NULL
사원 직책	employee_position	VARCHAR(50)	NOT NULL
등록일시	employee_info_reg_date	DATETIME	NOT NULL
변경일시	employee_info_update_date	DATETIME	NOT NULL

원부자재 정보(raw_material_info)			
원부자재 코드(PK)	raw_material_code	VARCHAR(50)	NOT NULL
원부자재 구분	raw_material_high_classification	VARCHAR(50)	NOT NULL
원부자재 재료 구분	raw_material_medium_classification	VARCHAR(50)	NOT NULL
원부자재 재료 구분 코드	raw_material_medium_code	VARCHAR(50)	NOT NULL
원부자재 재료명	raw_material_name	VARCHAR(50)	NOT NULL
원부자재 재료 코드	raw_material_sub_code	VARCHAR(50)	NOT NULL
원부자재 재료 색상 코드	raw_material_color_code	VARCHAR(50)	NOT NULL
특징 (패턴 및 모양)	raw_material_feature	VARCHAR(100)	NULL
단위	raw_material_unit	VARCHAR(10)	NOT NULL
등록일시	raw_material_reg_date	DATETIME	NOT NULL

변경일시	raw_material_update_date	DATETIME	NOT NULL
------	--------------------------	----------	----------

원부자재 발주(raw_material_order)			
원부자재 발주 코드(PK)	raw_material_order_code	VARCHAR(50)	NOT NULL
원부자재 코드(FK)	raw_material_code	VARCHAR(50)	NOT NULL
원부자재 거래처 코드(FK)	client_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
수량	raw_material_order_amount	INT(11)	NOT NULL
원부자재 승인상태	raw_material_order_approval	VARCHAR(10)	NOT NULL
주문 일시	raw_material_order_date	DATETIME	NOT NULL
입고예정일	raw_material_expected_date	DATETIME	NOT NULL
등록일시	raw_material_order_reg_date	DATETIME	NOT NULL
변경일시	raw_material_order_update_date	DATETIME	NOT NULL

계약상세목록(contract_group_list)			
계약번호	contract_num	VARCHAR(50)	NOT NULL
발주코드 / 의뢰코드	order_code	VARCHAR(50)	NOT NULL
거래처코드(FK)	client_code	VARCHAR(50)	NOT NULL

계약목록(수주/발주)(contract_info)			
계약번호(PK)	contract_num	VARCHAR(50)	NOT NULL
계약 담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
단위	contracted_unit	VARCHAR(10)	NOT NULL
수량	contracted_amount	INT(11)	NOT NULL
결제조건	contracted_payment_terms	VARCHAR(20)	NOT NULL
수주 / 발주 일자	contracted_date	DATETIME	NOT NULL
총계약금액	contract_total_payment	INT(11)	NOT NULL
계약상태	contract_state	VARCHAR(20)	NOT NULL
계약시작일자	contract_start_date	DATETIME	NOT NULL
계약종료일자	contract_end_date	DATETIME	NOT NULL
등록일시	contract_info_reg_date	DATETIME	NOT NULL
변경일시	contract_info_update_date	DATETIME	NOT NULL

원부자재 입고 / 출고(raw_material_inventory_transaction)			
입/출고 코드(PK)	material_transaction_code	VARCHAR(50)	NOT NULL
원부자재 코드(FK)	raw_material_code	VARCHAR(50)	NOT NULL
원부자재 발주 코드(FK)	raw_material_order_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
입고 / 출고 구분	transaction_category	VARCHAR(10)	NOT NULL
수량	raw_material_transaction_amount	INT(11)	NOT NULL

입고 / 출고 일시	raw_material_transaction_date	DATETIME	NOT NULL
등록일시	raw_material_transaction_reg_date	DATETIME	NOT NULL
변경일시	raw_material_transaction_update_date	DATETIME	NOT NULL

생산 제품 상세 코드(product_code_detail)			
품목코드(PK)	product_code	VARCHAR(50)	NOT NULL
성별에 따른 양복 코드	gender_categorized_code	VARCHAR(50)	NOT NULL
성별에 따른 양복명	gender_categorized_name	VARCHAR(50)	NOT NULL
품목 소분류 코드	detailed_categorized_code	VARCHAR(50)	NOT NULL
품목 소분류명	detailed_categorized_name	VARCHAR(50)	NOT NULL
등록일시	product_code_detail_reg_date	DATETIME	NOT NULL
변경일시	product_code_detail_update_date	DATETIME	NOT NULL

품목별 의뢰 상세 목록(requested_product_detail)			
품목별 의뢰 코드(PK)	requested_product_code	VARCHAR(50)	NOT NULL
품목코드(FK)	product_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
거래처코드(FK)	client_code	VARCHAR(50)	NOT NULL
수주 계약코드	contract_code	VARCHAR(50)	NOT NULL
의뢰코드	product_request_code	VARCHAR(50)	NOT NULL

도식화 첨부파일(jpg)	design_drawing_attachment	VARCHAR(200)	NULL
납기일	requested_due_date	DATE	NOT NULL
납품처 주소	requested_delivery_address	VARCHAR(50)	NOT NULL
납품처 연락처	requested_delivery_tel	VARCHAR(50)	NOT NULL
특이사항	requested_special_note	VARCHAR(50)	NULL
의뢰승인여부	approval_status	VARCHAR(50)	NOT NULL
등록일시	requested_product_detail_reg_date	DATETIME	NOT NULL
변경일시	requested_product_detail_update_date	DATETIME	NOT NULL

생산 품목별 측정 사이즈(required_size_list)			
품목 소분류 측정부위 코드(PK)	required_size_code	VARCHAR(50)	NOT NULL
품목 소분류 코드(FK)	detailed_categorized_code	VARCHAR(50)	NOT NULL
측정부위	measurement_part	VARCHAR(50)	NOT NULL
단위	unit	VARCHAR(10)	NOT NULL
등록일시	required_size_reg_date	DATETIME	NOT NULL
변경일시	required_size_update_date	DATETIME	NOT NULL

의뢰 품목별 측정 사이즈(requested_product_size)			
품목별 의뢰 코드(FK)	requested_product_code	VARCHAR(50)	NOT NULL

품목 소분류 측정부위 코드(FK)	required_size_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
측정값	measured_value	INT(11)	NOT NULL
등록일시	required_size_reg_date	DATETIME	NOT NULL
변경일시	required_size_update_date	DATETIME	NULL

의뢰 품목별 필요 원부자재 목록(used_raw_material)

품목별 의뢰 코드(FK)	requested_product_code	VARCHAR(50)	NOT NULL
원부자재 코드(FK)	raw_material_code	VARCHAR(50)	NOT NULL
입력 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
필요 수량	required_quantity	INT(11)	NOT NULL
첨부파일	used_raw_material_attachment	VARCHAR(200)	NULL
등록일시	used_raw_material_reg_date	DATETIME	NOT NULL
변경일시	used_raw_material_update_date	DATETIME	NULL

작업 지시(work_order)

작업 지시 코드(PK)	product_code	VARCHAR(50)	NOT NULL
품목별 의뢰 코드(FK)	requested_product_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
생산 시작 예정일	expected_production_start_date	DATETIME	NOT NULL
생산 마감 예정일	expected_production_end_date	DATETIME	NOT NULL

등록일시	work_order_reg_date	DATETIME	NOT NULL
변경일시	work_order_update_date	DATETIME	NULL

생산 계획(production_plan)			
생산계획코드(PK)	production_plan_code	VARCHAR(50)	NOT NULL
품목코드(FK)	product_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
계획수량	production_plan_quantity	INT(11)	NOT NULL
단위	production_plan_quantity_unit	VARCHAR(50)	NOT NULL
작업 시작 예정일	expected_production_start_date	DATE	NOT NULL
작업 마감 예정일	expected_production_end_date	DATE	NOT NULL
계획기간	production_planning_period	INT(11)	NOT NULL
비고	production_plan_note	VARCHAR(50)	NULL
등록일시	production_plan_reg_date	DATETIME	NOT NULL
변경일시	production_plan_update_date	DATETIME	NULL

품질검사 대분류 검사 코드분류(high_class_inspection_list)			
대분류 검사코드(PK)	high_class_code	VARCHAR(50)	NOT NULL
대분류 검사명	high_class_name	VARCHAR(50)	NOT NULL
등록일	high_class_reg_date	DATETIME	NOT NULL
변경일	high_class_update_date	DATETIME	NULL

품질검사 대/중분류 검사 코드분류(medium_class_inspection_list)			
대/중분류 검사코드(PK)	high_med_class_code	VARCHAR(50)	NOT NULL
대분류 검사코드(FK)	high_class_code	VARCHAR(50)	NOT NULL
중분류 검사코드	med_class_code	VARCHAR(50)	NOT NULL
중분류 검사명	med_class_name	VARCHAR(50)	NOT NULL
등록일	med_class_reg_date	DATETIME	NOT NULL
변경일	med_class_update_date	DATETIME	NULL

대/중/소분류 검사 코드분류(low_class_inspection_list)			
대/중/소분류 검사코드(PK)	high_med_low_class_code	VARCHAR(50)	NOT NULL
대분류 검사코드(FK)	high_class_code	VARCHAR(50)	NOT NULL
대/중분류 검사코드(FK)	high_med_class_code	VARCHAR(50)	NOT NULL
소분류 검사코드	low_class_code	VARCHAR(50)	NOT NULL
소분류 검사명	low_class_name	VARCHAR(50)	NOT NULL
등록일	low_class_reg_date	DATETIME	NOT NULL
변경일	low_class_update_date	DATETIME	NULL

대/중/소/세부 검사코드분류(sub_class_inspection_list)			
대/중/소/세부 검사코드(PK)	high_med_low_sub_class_code	VARCHAR(50)	NOT NULL
대분류 검사코드(FK)	high_class_code	VARCHAR(50)	NOT NULL
대/중분류 검사코드(FK)	high_med_class_code	VARCHAR(50)	NOT NULL

대/중/소분류 검사코드(FK)	high_med_low_class_code	VARCHAR(50)	NOT NULL
세부검사코드	sub_class_code	VARCHAR(50)	NOT NULL
세부검사명	sub_class_name	VARCHAR(50)	NOT NULL
등록일	low_class_reg_date	DATETIME	NOT NULL
변경일	low_class_update_date	DATETIME	NULL

품질검사 요청(quality_inspection_request)			
품질검사이청 코드(PK)	requested_product_code	VARCHAR(50)	NOT NULL
품목별 의뢰 코드(FK)	requested_product_code	VARCHAR(50)	NOT NULL
품질검사 요청 사원 코드(FK)	quality_inspection_request_employee_code	VARCHAR(50)	NOT NULL
품질검사 승인자 사원 코드(FK)	quality_inspection_request_approved_employee_code	VARCHAR(50)	NOT NULL
품질검사 코드	quality_inspection_code	VARCHAR(50)	NOT NULL
샘플링 여부	sampling_check	VARCHAR(50)	NOT NULL
특이사항	inspectioin_special_request	VARCHAR(50)	NOT NULL
승인여부	inspection_request_approval	VARCHAR(50)	NULL
검사 완료 요청 시각	requested_inspection_complete_date	DATETIME	NOT NULL
검사 요청 시각	inspection_request_date	DATETIME	NOT NULL
승인시각	inspection_request_approval_date	DATETIME	NOT NULL
등록일시	quality_inspection_request_reg_date	DATETIME	NOT NULL
변경일시	quality_inspection_request_update_date	DATETIME	NOT NULL

기준치: 원부자재 이화학검사 원단별 등급 기준치(biochem_fabric_level_standard)

이화학검사 등급 기준치 코드(PK)	biochem_fabric_level_standard_code	VARCHAR(50)	NOT NULL
품질검사 코드(FK)	quality_inspection_code	VARCHAR(50)	NOT NULL
원부자재 코드(FK)	raw_material_code	VARCHAR(50)	NOT NULL
최소치	min_value	INT(11)	NOT NULL
최대치	max_value	INT(11)	NOT NULL
측정등급	measured_level	INT(11)	NOT NULL
등록일시	biochem_inspection_standard_reg_date	DATETIME	NOT NULL
변경일시	biochem_inspection_standard_update_date	DATETIME	NUL

품질검사 기준표(inspection_standard)			
품질검사코드(PK)	quality_inspection_code	VARCHAR(50)	NOT NULL
구분	category	VARCHAR(50)	NOT NULL
최소값	min_value	INT(11)	NULL
최대값	max_value	INT(11)	NULL
측정방법	method	VARCHAR(50)	NULL
측정단위	unit	VARCHAR(50)	NULL
허용 오차 및 등급	standard_tolerance	VARCHAR(50)	NULL
필수요소	required_inspection_check	VARCHAR(50)	NULL
등록일시	level_standard_reg_date	DATETIME	NOT NULL
변경일시	level_standard_update_date	DATETIME	NULL

측정표: 품질 검사 결과 측정값(inspection_result)

세부품질검사 결과 측정값 코드(PK)	inspection_result_code	VARCHAR(50)	NOT NULL
품질검사요청 코드(FK)	quality_inspection_request_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NULL
원부자재코드(FK)	raw_material_code	VARCHAR(50)	NULL
품질검사 코드(FK)	quality_inspection_code	VARCHAR(50)	NOT NULL
측정회차	inspection_measurement_num	INT(11)	NOT NULL
측정 결과값	inspection_measurement_value	INT(11)	NULL
측정 등급 결과	inspection_measurement_level_result	INT(11)	NULL
오차 최소값	min_tolerance	INT(11)	NULL
오차 최대값	max_tolerance	INT(11)	NULL
오차 수치	error_range	INT(11)	NULL
합격/불합격 여부	inspection_pass_check	VARCHAR(50)	NOT NULL
검사 시작 일시	inspection_start_date	DATETIME	NOT NULL
검사 소요 시간	inspection_duration	INT(11)	NOT NULL
검사 완료 일시	inspection_end_date	DATETIME	NOT NULL
등록일시	inspection_result_reg_date	DATETIME	NOT NULL
변경일시	inspection_result_update_date	DATETIME	NULL

최종결과표: 품질검사 최종 결과(inspection_final_result)			
품질검사 결과 측정값 코드(PK)	inspection_result_code	VARCHAR(50)	NOT NULL
품질검사요청 코드(FK)	quality_inspection_request_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL

품질검사 코드(FK)	quality_inspection_code	VARCHAR(50)	NOT NULL
소분류 검사의 최종 합격/불합격 여부	low_cate_final_pass_check	VARCHAR(50)	NOT NULL
검사 최초 시작 일시	inspection_start_date	DATETIME	NOT NULL
검사 총 소요 시간	inpection_duration	INT(11)	NOT NULL
검사 최종 완료 일시	inspection_end_date	DATETIME	NOT NULL
등록일시	final_result_reg_date	DATETIME	NOT NULL
변경일시	final_result_update_date	DATETIME	NULL

검사항목별 결과에 따른 불량품 조치 결정에 대한 테이블(defective_product_management)			
불량품 코드(PK)	defective_product_code	VARCHAR(50)	NOT NULL
품질검사요청 코드(FK)	quality_inspection_request_code	VARCHAR(50)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
품질검사 코드(FK)	quality_inspection_code	VARCHAR(50)	NOT NULL
품질검사 결과 코드(FK)	inspection_result_code	VARCHAR(50)	NOT NULL
불량원인	defect_cause	VARCHAR(50)	NOT NULL
처리방법	defective_product_measure	VARCHAR(50)	NOT NULL
비고	management_note	VARCHAR(50)	NOT NULL
등록일시	defective_product_reg_date	DATETIME	NOT NULL
변경일시	defective_product_update_date	DATETIME	NULL

생산공정 목록(production_process_list)			
생산공정코드(PK)	production_process_code	VARCHAR(50)	NOT NULL

생산공정대분류 항목	production_process_high_classification	VARCHAR(50)	NOT NULL
생산공정중분류 항목	production_process_low_classification	VARCHAR(50)	NOT NULL
생산공정순번	process_order_num	INT(11)	NOT NULL
등록일시	production_process_reg_date	DATETIME	NOT NULL
변경일시	production_process_update_date	DATETIME	NULL

품질검사 실시 시점 명시 목록(quality_inspection_conducting_point)			
대분류 검사코드(FK)	quality_inspection_code	VARCHAR(50)	NOT NULL
검사 실시 시점의 생산 공정 순번(FK)	process_order_conducting_point	INT(11)	NOT NULL
등록일시	inspection_conducting_point_reg_date	DATETIME	NOT NULL
변경일시	inspection_conducting_point_update_date	DATETIME	NULL

출하 지시 목록(shipment_order)			
출하지시번호(PK)	shipment_order_code	VARCHAR(50)	NOT NULL
품목별 의뢰 코드(FK)	requested_product_code	VARCHAR(50)	NOT NULL
담당 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
출하그룹코드	shipment_order_group_code	VARCHAR(50)	NOT NULL
납품일자	shipment_order_date	DATE	NOT NULL
계약상태	shipment_order_contract_state	VARCHAR(50)	NOT NULL
등록일시	inspection_conducting_point_reg_date	DATETIME	NOT NULL
변경일시	inspection_conducting_point_update_date	DATETIME	NULL

출하 송장 목록(shipment_invoice)			
출하 송장 번호(PK)	shipment_invoice_code	INT(11)	NOT NULL
출하그룹코드(FK)	shipment_order_group_code	VARCHAR(50)	NOT NULL
운송 거래처 코드(FK)	client_code	VARCHAR(50)	NOT NULL
담당사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
출하일	shipping_date	DATE	NULL
등록일시	shipment_invoice_reg_date	DATETIME	NOT NULL
변경일시	shipment_invoice_reg_update_date	DATETIME	NULL

출고 정보 목록(shipment_info)			
출고번호(PK)	shipment_code	VARCHAR(50)	NOT NULL
출하 송장번호(FK)	shipment_invoice_code	INT(11)	NOT NULL
담당자 사원 코드(FK)	charge_employee_code	VARCHAR(50)	NOT NULL
납품처 주소	requested_delivery_address	VARCHAR(50)	NOT NULL
출고일	shipped_date	DATE	NOT NULL
등록일시	shipment_info_reg_date	DATETIME	NOT NULL
변경일시	shipment_info_update_date	DATETIME	NULL

품목별 생산공정현황(product_production_process_status)			
생산공정현황코드(PK)	product_production_process_status_code	VARCHAR(50)	NOT NULL
생산공정코드	production_process_code	VARCHAR(50)	NOT NULL
품목별 의뢰코드	requested_product_code	VARCHAR(50)	NOT NULL

품목별 생산 시작일시	process_start_date	DATETIME	NOT NULL
품목별 생산 완료일시	completed_product_update_date	DATETIME	NOT NULL
공정상태	process_status	VARCHAR(50)	NOT NULL
등록일시	production_process_status_reg_date	DATETIME	NOT NULL
변경일시	production_process_status_update_date	DATETIME	NULL

완제품 관리(completed_product_list)			
품목별 의뢰코드(PK)	requested_product_code	VARCHAR(50)	NOT NULL
생산시작일시	production_start_date	DATETIME	NOT NULL
생산완료일시	production_finish_date	DATETIME	NOT NULL
등록일시	completed_product_reg_date	DATETIME	NOT NULL
변경일시	completed_product_update_date	DATETIME	NULL

4.3 네이밍 규칙

공통 규칙	<ul style="list-style-type: none"> 1 모든 이름은 a-z, A-Z, 0-9 의 글자로 조합한다. 1 프로그램 내에서 사용하는 모든 이름은 64 자를 넘지 않는 것으로 한다. 1 정확한 의미 전달을 위해 약어의 사용은 가능한 한 피한다. 1 특수기호는 사용하지 않는 것을 원칙으로 한다. 1 약어 사용이 의미 전달에 도움이 되는 경우에는 표준용어사전에 포함되어있는 약속된 약어를 사용할 수 있으나, 그렇지 않은 경우에는 단어의 spelling 을 모두 적는 것으로 한다. 	
	JAVA	대소문자를 사용하여 단어의 시작부분은 소문자로 표기하고 뒤에 오는 단어 앞자리 한 글자를 대문자를 사용한다. (카멜 표기법)
		단어는 표준용어사전을 따른다.
		명사 또는 동사 + 명사의 혼합된 형태로 이름을 구성한다.
		첫 번째 문자는 항상 대문자를 사용한다.
		클래스 유형에 따라 다음 표를 참고하여 구성한다.
파일명 규칙	mapper	Dao (java 파일)의 명명 규칙을 따른다. (ex: Dao ProductionrMapper -> mapper ProductionMapper)
		파일의 확장자는 .xml 로 한다.
	html	대소문자를 사용하여 단어의 시작부분은 소문자로 표기하고 뒤에 오는 단어 앞자리 한 글자를 대문자를 사용한다. (카멜 표기법)
		파일의 확장자는 .html 로 한다.
	css	대소문자를 사용하여 단어의 시작부분은 소문자로 표기하고 뒤에 오는 단어 앞자리 한 글자를 대문자를 사용한다. (카멜 표기법)
		파일의 확장자는 .css 로 한다.
	javaScript	대소문자를 사용하여 단어의 시작부분은 소문자로 표기하고 뒤에 오는 단어 앞자리 한 글자를 대문자를 사용한다. (카멜 표기법)

		파일의 확장자는 .js 로 한다.		
		오픈소스 자바스크립트의 경우, 최초 배포된 이름 그대로 유지한다.		
Method 규칙	l 동사 혹은 동사 + 명사로 작성하며, 혼합된 형태로 이름을 구성할 때에는 각 단어의 첫 글자는 대문자로 작성한다. (카멜표기법)			
	l 상황에 따라 다음 표를 참고하여 일반적인 Prefix 를 적용하며, 나머지는 필요에 따른 적절한 이름을 사용한다.			
	기능유형	대표용어	기능유형	대표용어
	get	조회	save	저장
	add	등록	call	호출
	modify	수정	send	송신
	delete	삭제	receive	수신
	confirm	확인	transfer	송수신
	cancel	취소	check	검토
	calc	계산	is	유무(여부)확인
변수명 규칙	l 대소문자를 사용하여 변수명의 시작부분을 소문자로 표기한다.			
	l 변수명에는 기호나 특수문자를 사용하지 않는다.			
	l 변수명은 의미 있는 단어를 사용하도록 한다.(표준 용어사전 참고)			
	l 변수에 해당하는 컬럼이 테이블에 존재하는 경우에는 컬럼명을 기준으로 camel 표기법을 준수하여 사용한다.			
	l 변수명은 중복되는 이름이 나오지 않도록 하여 일련번호를 붙여 사용하는 경우가 없도록 한다.			
	l 상수명은 공통코드를 제외한 변수명 전체를 대문자로 하고, 단어 연결에는 ‘_’를 사용한다.			
DB	l DB 의 테이블 및 칼럼의 모든 명칭은 영문자(소문자)로 시작한다.			
	l 테이블명칭은 해당 주요기능을 표현하며, 칼럼 또한 칼럼의 속성이 표현될 수 있는 명칭을 사용한다.			
	l 의미를 알 수 있는 약어는 사용하지 않으며 되도록 풀네임을 사용한다.			

4.4 디렉토리 구조

view	contract	contract.html	수/발주 관리 메인화면
		addBuyerOrder.html	주문서 등록 화면
		buyerOrderList.html	주문서 목록 및 조회 화면
		modifyBuyerOrder.html	주문서 수정 화면
		buyerOrderInfo.html	상세 주문서 화면
		buyerOrderApproval.html	승인된 주문 목록 화면
		addBuyerContract.html	수주 계약 등록 화면
		buyerContractList.html	수주 계약 목록 및 조회 화면
		modifyBuyerContract.html	수주 계약 수정 화면
		addSupplierRequest.html	발주요청화면
		supplierRequestList.html	발주요청리스트
		modifySupplierRequest.html	발주요청수정화면
		addSupplierContract.html	발주계약등록
		supplierContractList.html	발주계약리스트
		modifySupplierRequest.html	발주요청수정화면
		addSupplierContract.html	발주계약등록
		addSupplierContract.html	발주계약리스트
		modifySupplierContract.html	발주계약수정화면
	rawMaterials	rawMaterials.html	자재 관리 메인 화면
		addInWarehousing.html	자재 입고 등록
		inWarehousingList.html	자재 입고 조회
		modifyInWarehousing.html	자재 입고 수정
		addExWarehousing.html	자재 출고 등록 화면
		exWarehousingList.html	자재 출고 목록 및 조회 화면
		modifyExWarehousing.html	자재 출고 수정 화면

		inventoryStatus.html	원부자재 현재재고 현황 화면
		warehousingList.html	입/출고 종합 조회 화면
	production	production.html	생산관리 메인 화면
		addProductionPlan.html	생산 계획 등록
		productionPlanList.html	생산 계획 조회
		productionWeeklyPlanList.html	주간별 생산 계획 조회
		productionMonthlyPlanList.html	월별 생산 계획 조회
		addWorkOrder.html	작업지시서 등록
		workOrderList.html	작업지시 목록 및 조회
		productionOrderList.html	생산 대기중 의뢰품목 목록 화면
		orderProductionProcess.html	의뢰품목 생산 지시 화면
		stateByWorkOrder.html	작업지시별 생산현황
		stateByProductionPlan.html	생산계획별 생산현황 화면
		detailedStateByProduct.html	의뢰 품목별 상세 생산공정현황
		completedProductList.html	완제품 조회/목록 화면
	quality	quality.html	품질관리 메인화면
		addStandardTable.html	기준표 등록 화면
		standardTableList.html	기준표 조회/목록 화면
		modifyStandardTable.html	기준표 수정 화면
		qualityInspectionRequest.html	품질 검사 요청
		inspectionMeasurementValueList.html	품질 검사 요청 목록
		qualityInspectionReport.html	품질검사 성적서 조회및 목록
		qualityInspectionStatusNow.html	실시간 검사현황
		qualityInspectionStatusContract.html	계약별검사현황
		addDefectInspectionResultStatus.html	불량현황
		stateBuyerContractQualityInspection.html	수주계약별 검사현황

		searchByPeriodContractQualityInspection.html	수주계약 검사현황기간별조회
		inspectionPerformance.html	실적
		addDefectiveProduct.html	불량품 등록 화면
		defectiveProductList.html	불량품 목록 및 조회 화면
		modifyDefectiveProduct.html	불량품 수정 화면
	shipment	shipment.html	출하관리 메인 화면
		addShipmentOrder.html	출하지시등록
		shipmentOrderList.html	출하지시목록 및 조회
		modifyShipmentOrder.html	출하지시수정화면
		addShipmentInvoice.html	송장등록
		shipmentInvoiceList.html	송장목록 및 조회
		modifyShipmentInvoice.html	송장수정 화면
		courierlist.html	운송정보
		addShipment.html	출고 등록 화면
		shipmentList.html	출고 조회 화면
	system	system.html	기초정보관리화면
		addHumanResources.html	사원등록 화면
		HumanResourcesList.html	사원목록 조회 화면
		modifyHumanResources.html	사원 수정 화면
		addClient.html	거래처등록화면
		clientList.html	거래처목록 및 조회 화면
		modifyClient.html	거래처 수정 화면
		addAccountSubject.html	계정과목 등록
		accountSubjectList.html	계정과목 조회
		addRawMaterials.html	원부자재 등록
		modifyRawMaterials.html	원부자재 수정
		rawMaterialsList.html	원부자재 목록/조회

		addQualityInspection.html	품질검사등록화면
		qualityInspectionList.html	품질검사목록 및 조회
		modifyQualityInspection.html	품질검사 수정화면
		addProductionProcess.html	생산 공정 등록 화면
		productionProcessList.html	생산 공정 목록 화면

4.6 패키지 구조

패키지명	클래스명	메서드명	설명
ksmart39.springbo ot.controller	ContractController	getContractmangement	수/발주 관리 메인화면
		addBuyerOrder	수주 주문서 등록
		buyerOrderList	수주 주문서 목록 및 조회
		modifyBuyerOrder	수주 주문서 수정
		buyerOrderInfo	수주 상세 주문서 목록
		buyerOrderApproval	수주 주문서 승인완료 목록
		deleteBuyerOrder	수주 주문서 삭제
		addBuyerContract	수주 계약 등록
		buyerContractList	수주 계약 목록 및 조회

		modifyBuyerContract	수주 계약 수정
		deleteBuyerContract	수주 계약 삭제
		addSupplierRequest	발주요청등록
		modifySupplierRequest	발주요청 수정
		getSupplierRequestList	발주요청목록 및 조회
		deleteSupplierRequest	발주요청삭 wp
		addSupplierContract	발주계약등록
		getSupplierContractList	발주계약목록 및 조회
		modifySupplierContract	발주계약 수정
		deleteSupplierContract	발주계약 삭제
		addInvoicePayment	예정 수금/출금 등록
		getExpectedPaymentList	예정 수금/출금 목록 반환
		addInvoiceClaim	수금/출금 등록

		addInvoiceApproval	결제 등록
		getInvoiceApproval	결제 승인 현황 목록
		addPaymentInvoice	전표 등록
		paymentInvoiceList	전표 목록
	RawMaterialsController	meterials	자재정보 메인화면
		addInWarehousing	자재 입고 등록
		InWarehousingList	자재 입고 목록
		modifyInWarehousing	자재 입고 수정화면으로 파라미터 가지고 넘어가는 화면조회하는 메서드
		modifyInWarehousing	자재입고 수정 처리 후 자재입고 리스트 출력
		addExWarehousing	자재 출고 등록
		getExWarehousingList	자재 출고 목록 및 조회

		modifyExWarehousing	자재출고 수정
		deleteExWarehousing	자재 출고 삭제
		getInventorySearchKey	입출고 검색 처리
		getInventoryStatus	원부자재 현재 재고 현황 목록
		getWarehousingList	원부자재 입출고 종합 목록
		getContractmangement	자재입고현황에서 수정화면으로 넘어갈때 상황 추가
		modifyInWarehousing	자재입고 수정 후 자재입고 리스트 출력
	ProductionController.java	getProdcutioncontrol	생산관리 메인화면
		productioncontrolAdd	생산계획 등록
		productionPlanList	전체 생산계획 목록
		getProductionMonthlyPlanList	생산계획 월별 목록
		monthPlan	생산계획 데이터 가져오기

		getProductionMonthlyPlanList	생산계획 월별 목록
		productionWeeklyPlanList	생산계획 주간별 목록
		getProductionCode	성별별 양복명 값 가져오기
		getDetailCode	성별별 양복명에 맞는 소분류 가져오기
		modifyProductionPlan	생산계획 수정
		searchProductionPlan	생산계획 검색
		deleteProductionPlan	생산계획 삭제
		getStateByProduct	의뢰품목별 생산 현황 조회
		getStateByWorkOrder	작업지시별 생산 현황 조회
		getContractCode	작업지시등록화면에서 계약번호 검색 모달
		addWorkOrder	작업지시 등록
		getWorkerOrderList	작업지시 목록
		getCompletedProductList	완제품 목록
		modifyCompletedProduct	완제품 수정

		getStateByProductionPlan	생산계획별 생산 현황 조회
		startPCodeProduction	생산공정 시작
		getProductionOrderList	의뢰품목별 상세 생산 공정 현황
		getSearchKey	생산공정 지시검색모달창
		startProcess	생산공정시작버튼
		stopProcess	생산공정마침버튼
		getProductToStartResult	의뢰 품목별 생산공정 현황 조회
		orderProductionProcess	의뢰품목별 생산공정 지시
	QualityControlController.java	getQuality	품질관리 메인화면
		addStandardTable	품질검사 기준표 등록
		getStandardTableList	품질검사 기준표 목록
		getProductInfoForInspection	생산공정에따른검사 코드

		getDefectiveRate	품질검사별 불량률 현황
		getSearchQualityInspectionS tate	수주계약별 검색
		qualityInspectionStatusContr act	수주계약별 검색 품질검사 현황
		searchRequesetInspection	품질검사 최종등록 검색모달창
		getFinalResultReport	검사현황 조회 성적서 모달창값전달하기
		getQualityInspectionReport	검사현황 조회 성적서 결과값보여주기
		getProductName	검사현황 조회 품목명 가지고오기
		getRequestProductCode	검사현황 조회 의뢰코드가지고오기
		getClientName	검사현황 조회 거래처명가지고오기
		getQualityInspectionReport	검사현황 성적서조회 및리스트
		addQualityInspectionReport	검사현황최종결과등 록

		searchByPeriodContractQualityInspection	수주계약별 기간별조회
		qualityInspectionRequest	품질검사요청
		qualityControlRequestList	품질검사요청목록
		searchQualityInspectionRequest	품질검사 요청 목록 모달
		addInspectionMeasurementValue	품질검사 측정값 등록 화면
		getQualityInspectionCategory	품질검사 측정값 카테고리에 맞는 행 만들기
		addInspectionMeasurementValue	품질검사 측정값 등록
		inspectionMeasurementValueList	품질검사 측정값 목록
		inspectionPerformance	검사현황 실적
		qualityInspectionStatusNow	실시간 검사 현황
		searchByContractNum	실시간검사현황 계약번호 검색
		searchByClientName	실시간검사현황 거래처이름 검색

		searchInspectionCode	실시간검사현황 품질검사 코드 검색
		modifyDefectiveProduct	불량품 수정
		getDefectiveProductList	불량품 조회
		addDefectiveProduct	불량품등록
		getDefectiveRate	품질검사별 불량률 현황
	ShipmentController.java	getShipment()	출고관리메인
		addShipmentOrder	출하지시등록
		getShipmentOrderList	출하지시목록
		modifyShipmentOrder	출하지시수정화면
		deleteShipmentOrder	출하지시 삭제
		addShipmentInvoice	송장등록
		getShipmentInvoiceList	송장 목록
		modifyShipmentInvoice	수정 화면

		deleteShipmentInvoice	송장삭제
		getShipmentList	출고리스트
		modifyShipment	출고수정
		addShippedProductInfo	출고등록
	SystemController.java	system	기초정보관리
		addHumanResources	기초정보관리 사원등록
		memberIdCheck	기초정보관리 사원등록시 아이디중복체크
		getHumanResourcesList	사원목록
		modifyHumanResources	사원수정
		deleteHumanResources	사원삭제
		addClient	거래처등록
		getClientList	거래처리스트
		modifyClient	거래처수정
		deleteClient	거래처삭제
		addAccountSubject	계정과목 등록

		getAccountCategoryList	계정 과목 목록&검색
		modifyAccountSubject	계정 과목 수정
		addProductionProcess	생산공정 등록
		getProductionProcessList	생산공정 목록
		addRawMaterials	원부자재 등록
		getRawMaterialsList	원부자재 리스트 조회
		modifyRawMaterialsInfo	원부자재정보 수정
		sendMaterialsName	원부자재 재료 구분 검색후 처리
		getSearchValue	원부자재 재료 구분 검색
		addQualityInspection	품질검사종류 등록
		getQualityInspectionList	품질검사 리스트
		modifyQualityInspection	품질검사 수정
		getLowClassCate	품질검사 소분류 가지고오기
		getMediumClassCate	품질검사 중분류 가지고오기

		getHighClassCate	품질검사 대분류가지고오기
		deleteQualityInspection	품질검사 삭제
		searchInspectionList	품질검사 항목 검색
ksmart39.spring boot.service	OrderService.java	getOrderList	주문서 목록
	CompletedProductService.java	addCompletedProduct	완제품 품목 등록
		getCompletedProductList	완제품 품목 목록
	DefectiveProductService.java	addDefectiveProduct	불량품등록
		getDefectiveProduct	불량품목록
	LoginService.java	loginEmployee	로그인정보 확인
	ProductionPlanService.java	getProductCode	생산코드 조회
		getDetailCode	생산계획등록시소분 류 조회
		addProductionPlan	생산계획 등록
		getProductionAllPlanList	생산계획 전체 조회
		modifyProductionPlan	생산계획 수정 등록
		getProductionPlanListByCode	생산계획 코드검색

		searchProductionPlan	생산계획 검색
		deleteProductionPlan	생산계획 삭제
	ProductionService.java	startProcess	생산시작 버튼 누르면 시간 update
		insertNextProcess	생산공정 완료 버튼 누르면 다음 공정 자동등록
		completeProcess	생산공정 완료 버튼을 누르면 완료시간자동등록
		searchProductToStart	의뢰 품목별로의 생산현황 검색 결과
		searchClientName	거래처명 검색 모달 결과
		startProduction	생산 시작 생산현황에 첫공정 등록
		getProductReadyToStart	생산 대기중인 품목 목록
	ProductionStatusByProducti onPlanService.java	getFinishedProductionPlanInf o	종료된 생산계획 정보 조회
		getAchievePercentageByPla n	생산계획별 달성률 조회
	ProductionStatusService.jav a	getProductionStatus	생산현황

	ProductStateService.java	searchProductState	생산현황검색시 목록 보여주기
		getProductState	생산현황 목록
	QualityControlService.java	getInspectionStandard	품질검사 기준표 리스트
		getQualityInspectionList	품질검사 리스트 조회메서드
	QualityInsepctionFinalResult Service.java	getInsepectionFinalResult	품질검사 성적서 목록
		getClientName	검사현황 조회 거래처명가지고오기
		getRequestProductCode	검사현황 조회 의뢰코드명가지고오 기
		getProductName	검사현황 조회 품목명가지고오기
		getFinalResultReport	검사현황 조회 성적서 모달창값전달하기
		getQualityInspectionReport	검사현황 조회 성적서 결과값보여주기
		searchRequesetInspection	검사등록 검사결과 모달창 리스트
	QualityInsepctionStauteServ ice.java	getStateBuyerContractQualit yInspection	수주계약별 검사현황 리스트

		getSearchQualityInspectionState	수주계약별 검사 조회
	QualityInsMeasurementValueService.java	searchQualityInspectionRequest	품질검사 요청 검색 모달
		getQualityInspectionStandard	품질검사 카테고리 가져오기
		addQualityInspectionResult	품질검사 측정값 등록
		getInspectionMeasurementValueList	품질검사 측정값 목록
		getQualityInspectionStatusNow	품질검사 측정값 현황
		searchByContractNum	품질검사 측정값목록 계약번호검색
		searchByClientName	품질검사 측정값목록 거래처 검색
		searchInspectionCode	품질검사 측정값목록 품질검사 검색
	RawMaterialsInventoryStatusService.java	getSupplierRequest	발주요청 승인목록 가져오기
		getInventoryStatusByMCode	자재 현재고 리스트
		modifyMaterialIn	자재 입고 수정 처리

		getTransInfoByCode	자재 입고 수정 화면
		getRawMaterialsInventory	자재 입고고 현황
		getExwarehousing	자재출고 현황
		addExwarehousing	자재출고 등록
	RequestedProductService.java	getRequestedProduct	의뢰품목 dao
	getRequestedProduct	getSupplierContractList	계약 리스트
		getSupplierRequestList	발주요청 리스트
		addSupplierRequest	발주요청 등록
		getRawInfo	원부자재 정보
		getClientInfo	발주 요청 거래처 정보
		getRawInventoryState	발주요청에 자재 현황 리스트
	SystemService.java	getHumanResources	사원전체조회
		addHumanResources	사원정보등록
		getEmployeeInfoById	사원정정보조회

		getEmployeeInfoByCode	사원정보수정사원정보 가져오기
		modifyHumanResources	사원정보수정
		deleteHumanResources	사원정보삭제
		getClient	거래처 전체 조회
		addClient	거래처 등록
		modifyClient	거래처 수정
		getClientInfoByCode	거래처수정시거래처 정보 가져오기
		deleteClient	거래처목록 거래처 삭제
		modifyMember	계정과목 수정
		getAccountSubjectByCode	계정과목 수정 화면
		addAccountSubject	계정과목 추가
		getAccountingSubjectList	회계과목 조회
		searchInspectionList	품질검사 조회목록
		deleteQualityInspection	품질검사 목록 삭제
		addQualityInspection	품질검사 등록
		getHighClassCate	품질검사 등록 대분류 평가 가져오기

		getMediumClassCate	품질 검사 등록 중분류가져오기
		getLowClassCate	품질검사 소분류가져오기
		modifyQualityInspection	품질검사 수정
		getQualityInspectionCode	품질검사항목 가져오기
		getQualityInspectionList	품질검사 리스트
		getMaterialsList	원부자재 전체 리스트 조회
	WorkOrderService.java	getWorkOrderList	작업지시 목록
	ContractService.java	getSupplierContractList	발주계약목록
	ShipmentService.java	addShipmentOrder	출하지시등록
		getShipmentOrderList	출하지시목록
		modifyShipmentOrder	출하지시수정
		addShipmentInvoice	송장등록
		getShipmentInvoiceList	송장목록

		modifyShipmentInvoice	송장수정
ksmart39.spring boot.dao	CompletedProductMapper.java	getCompletedProductInfo	완제품 품목코드 + 최초시작일자 + 마지막공정완료일자 목록
		addCompletedProduct	완제품 품목 등록
		getCompletedProductList	완제품 품목 목록
	ContractMapper.java	getSupplierContractList	발주계약목록
	DefectiveProductMapper.java	getDefectiveProduct	불량품목록
		addDefectiveProduct	불량품등록
	LoginMapper.java	getEmployeeInfoById	로그인했을때 회원정보
	OrderMapper.java	getOrderList	주문서 목록 뿌려주기
	ProductionMapper.java	getProductReadyToStart	생산 대기중인 품목 목록
		startProduction	생산 시작하기 - 생산현황에 첫공정등록
		searchClientName	거래처명 검색 모달 결과
		searchProductToStart	의뢰 품목별로의 생산현황 검색 결과

		getProcessDetails	생산 프로세스의 정보 가져오기
		completeProcess	생산완료버튼 누르면 그에 해당하는 공정 완료일지 update
		insertNextProcess	생산완료버튼 누르면 다음 공정 insert
		startProcess	생산시작 버튼 누르면 시간 update
	ProductionPlanMapper.java	deleteProductionPlan	생산계획 삭제
		searchProductionPlan	생산계획 검색
		modifyProductionPlan	생산계획 수정 등록
		getProductionAllPlanList	생산계획 수정 폼
		getProductionAllPlanList	생산계획 전체 조회
		addProductionPlan	생산계획 등록
		getDetailCode	생산계획 등록 폼 - 소분류 조회
		getProductCode	생산계획 등록 폼 - 제품 코드 조회

	ProductionStatusByProductionPlanMapper.java	getFinishedProductionPlanInfo	생산계획별 달성률 조회
		getAchievePercentageByPlan	생산계획 끝난 계획의 정보
	ProductionStatusMapper.java	getProductionStatus	생산현황
	ProductStateMapper.java	searchProductState	생산현황목록 생산코드 검색시목록
		getProductState	생산현황 전체목록
	QualityControlMapper.java	getInspectionStandard	품질검사 기준표 리스트
		getQualityInspectionList	품질검사 리스트 조회메서드
	QualityInsepctionFinalResultMapper.java	getFinalInspectionResult	품질검사 최종결과값 가져오기
		getInsepectionFinalResult	품질검사 최종 리스트 화면
		getClientName	검사현황 조회 거래처명가지고오기
		getRequestProductCode	검사현황 조회 의뢰코드명가지고오기
		getProductName	검사현황 조회 품목명가지고오기

		getFinalResultReport	검사현황 조회 성적서 모달창값전달하기
		getQualityInspectionReport	검사현황 조회 성적서 결과값보여주기
		searchRequesetInspection	검사등록 검사결과 모달창 리스트
	QualityInsepctionStauteMapper.java	getSearchQualityInspectionState	수주계약별 검사 조회
		getStateBuyerContractQualityInspection	수주계약별 검사현황
	QualityInsMeasurementValueMapper.java	searchQualityInspectionRequest	품질검사 요청 검색 모달
		getBiochemFabricLevelStandard	원부자재 이화학검사 원단별 등급 기준치
		getQualityInspectionStandard	품질검사 기준표
		addQualityRawMaterialInspectionResult	품질검사 측정표 등록
		getInspectionMeasurementValueList	품질검사 측정 목록
		getQualityInspectionStatusNow	품질검사 현황

		searchByContractNum	계약번호 검색
		searchByClientName	거래처 검색
		searchInspectionCode	품질검사 측정표등록시품질검 사 검색
	RawMaterialsInventoryStatusMapper.java	getSupplierRequest	발주요청승인목록
		getInventoryStatusByMCode	자재 현재고 리스트
		modifyMaterialIn	자재 입고 수정 처리
		getTransInfoByCode	자재 입고 수정화면
		getMaterialsTransactionList	자재 입출고 현황 리스트
		getExwarehousing	출고현황 리스트
		addExwarehousing	출고 등록
	RequestedProductMapper.java	getRequestedProduct	의뢰품목 dao
	ShipmentMapper.java	addShipmentOrder	출하지시등록
		getShipmentOrderList	출하지시목록
			출하지시수정

		modifyShipmentOrder	
		addShipmentInvoice	송장등록
		getShipmentInvoiceList	송장목록
		modifyShipmentInvoice	송장수정
	SupplierMapper.java	getClientInfo	Modal 창 거래처 정보 리스트
		getRawInfo	원부자재 정보
		addSupplierRequest	발주요청 등록
		getSupplierRequestList	발주요청 리스트
		getSupplierContractList	발주계약 리스트
	SystemMapper.java	getHumanResources	사원 전체조회
		addHumanResources	사원등록
		getEmployeeInfoById	사원 정보조회
		getEmployeeInfoByCode	사원 정보조회
		modifyHumanResources	사원 정보수정
		deleteHumanResources	사원 정보삭제
		getClient	거래처 전체조회
		addClient	거래처 등록

		modifyClient	거래처 수정
		getClientInfoByCode	거래처 정보 가져오기
		deleteClient	거래처 삭제
		modifyMember	계정과목 수정
		getAccountSubjectByCode	계정과목 정보
		addAccountSubject	계정과목 추가
		searchInspectionList	품질검사 조회목록
		deleteQualityInspection	품질검사 목록 삭제
		addQualityInspection	품질검사등록
		modifyQualityInspection	품질검사 수정
		getQualityInspectionCode	품질검사 항목가져오기
		getQualityInspectionList	품질검사 리스트
		getHighClassCate	품질검사등록 대분류가져오기

		getMediumClassCate	품질 검사 등록 중분류 가져오기
		getLowClassCate	품질검사 소분류명 가져오기
		getMaterialsList	원부자재 전체 리스트 조회
	WorkOrderMapper.java	getWorkOrderList	작업지시 목록
		getPcodeFromProcessStatus	생산을 시작한 품목코드 리스트
ksmart39.springboot .domain	AccountingCategory.java		계정과목 dto
	Client.java		거래처 dto
	CompletedProduct.java		완성품 dto
	ContractGroupList.java		계약그룹 dto
	ContractInfo.java		계약정보 dto
	DefectiveProduct.java		불량품 dto
	HighClassInspection.java		품질검사 대분류 dto
	HumanResources.java		사원 dto
	HumanResourcesLevel.java		사원 erp 접근권한 dto
	LowClassInspection.java		품질검사소분류 dto

	MediumClassInspection.java		품질검사 중분류 dto
	ProductionPlan.java		생산계획 dto
	ProductionProcessList.java		생산공정목록 dto
	ProductProductionProcessStatus.java		생산공정현황 dto
	QualityBiochemFabricLevelStandard.java		
	QualityInspection.java		
	QualityInspectionFinalResult.java		품질검사최종결과 dto
	QualityInspectionResult.java		품질검사 측정값 dto
	QualityInspectionStandard.java		품질검사 기준표 dto
	RawMaterials.java		원부자재 dto
	RawMaterialsInventory.java		원부자재 입출고 dto
	RequestedProduct.java		의뢰품목 dto
	ShipmentInvoice.java		송장관리 dto
	ShipmentOrder.java		출하지시 dto
	SubClassInspection.java		품질검사 소분류상세 dto

	SupplierRequest.java		발주요청 dto
	WorkOrder.java		작업지시 dto
ksmart39.springboot.interceptor	LoginInterceptor.java		로그인 인터셉터

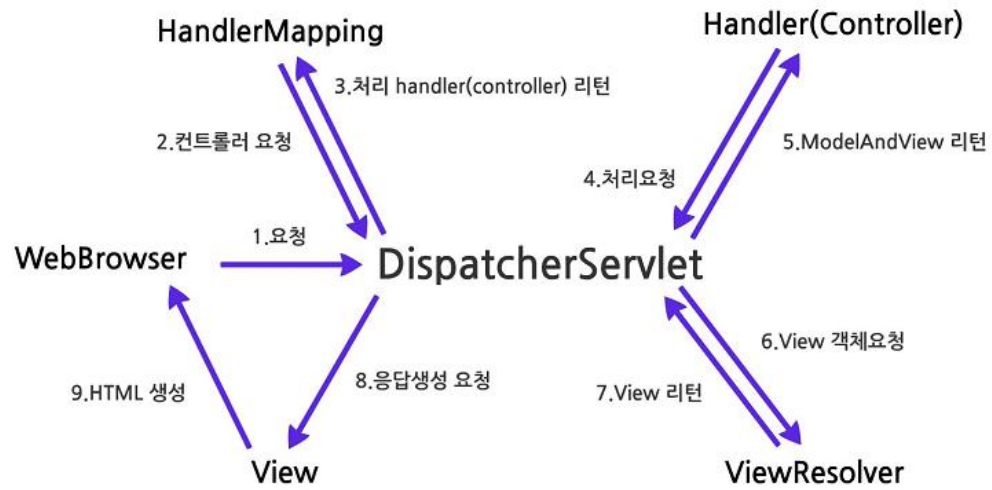
5. 개발 / 서비스 환경

*	구 분	개발 환경		서비스 환경
S / W	OS	Windows 10		Windows 10
	웹브라우저	Chrome		Chrome 최적화
	WAS	apache-tomcat-9.0.27		apache-tomcat-9.0.27
	개발언어	서버	Java	-
		클라이언트	JavaScript HTML5 CSS thymeleaf jQuery	
	Framework	bootstrap 4 / SpringFramework / Mybatis 3.0		클라이언트
	DBMS	MySQL5.5.32		MariaDB 10.1
	Tools	JDK Eclipse Spring Tool Suite		OpenJDK 8.x

6. 어플리케이션

6.1 어플리케이션 프로시저

기본 어플리케이션 프로시저



6.2 서버 및 기타 설정

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.4.5</version>
    <relativePath/> <!-- lookup parent from repository -->
  </parent>
  <groupId>ksmart39</groupId>
  <artifactId>teamBootstrap</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>
  <name>teamBootstrap</name>
  <description>ksmart39 project for Spring Boot</description>
  <properties>
    <java.version>1.8</java.version>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-thymeleaf</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.boot</groupId>
```

application.properties

#server port 설정

server.port=80

#thymeleaf 새 로고침 cache -> false

spring.thymeleaf.cache=false

#log4jdbc 설정 파일 위치

logging.config=classpath:logback-spring.xml

#session tracking 3 가지 ssl, cookie, url 다 허용으로 설정되어짐, 그래서 cookie 에 의해서만
트래킹할 수 있게 설정.

server.servlet.session.tracking-modes=cookie

#HTTP ONLY servlet 3.0 에서 추가되었음. 오직 http 에 의해서 해당 쿠키가 사용되는
것(스크립트 요청에 의해서는 쿠키를 사용하는 것을 방지한다.)

server.servlet.session.cookie.http-only=true

#DB 연동

spring.datasource.driver-class-name=net.sf.log4jdbc.sql.jdbcapi.DriverSpy

spring.datasource.url=jdbc:log4jdbc:mysql://sodak0127.cafe24.com:3306/sodak0127?serverT
imezone=UTC&characterEncoding=UTF8

spring.datasource.username=sodak0127

spring.datasource.password= password

#src/main/resources/ == classpath:

#mapper 파일 위치 설정

mybatis.mapper-locations=classpath:mapper/**/*.xml

#mapper dto 패키지 생략 설정

mybatis.type-aliases-package=ksmart39.springboot.domain

#spring banner 파일 위치 설정

spring.banner.location=classpath:templates/banner.txt

6.3 주요 기능별 정리 -각자

6.3.1 품질검사등록

품질검사등록

품질검사 등록시 품질검사 대분류 중분류 소분류 명을 선택하지않으면 alert 창으로 경고를 띄워주고 선택후 품질검사 소분류상세명에 한글로 6 글자이내로 등록하지않으면 경고창을띄워줍니다.

TEAMONEMES

HOME1.수/발주관리2.자재관리3.생산관리4.품질관리5.출하관리6.시스템관리

배포일시21-07-19 16:10:00관리자CEO 임원 이현빛님이 로그인 하셨습니다.

시스템관리

사용자관리 [한빛]

거래처관리 [한빛]

회계기초관리 [다미]

자재기초관리 [민아]

생산공정관리 [민아]

품질기초관리 [보람]

품질검사:검사등록

품질검사등록

품질검사 대분류명::대분류검사명::

품질검사 중분류명::중분류검사명::

품질검사 소분류명::소분류검사명::

품질검사 소분류상세명

품질검사등록품질검사등록취소

관리

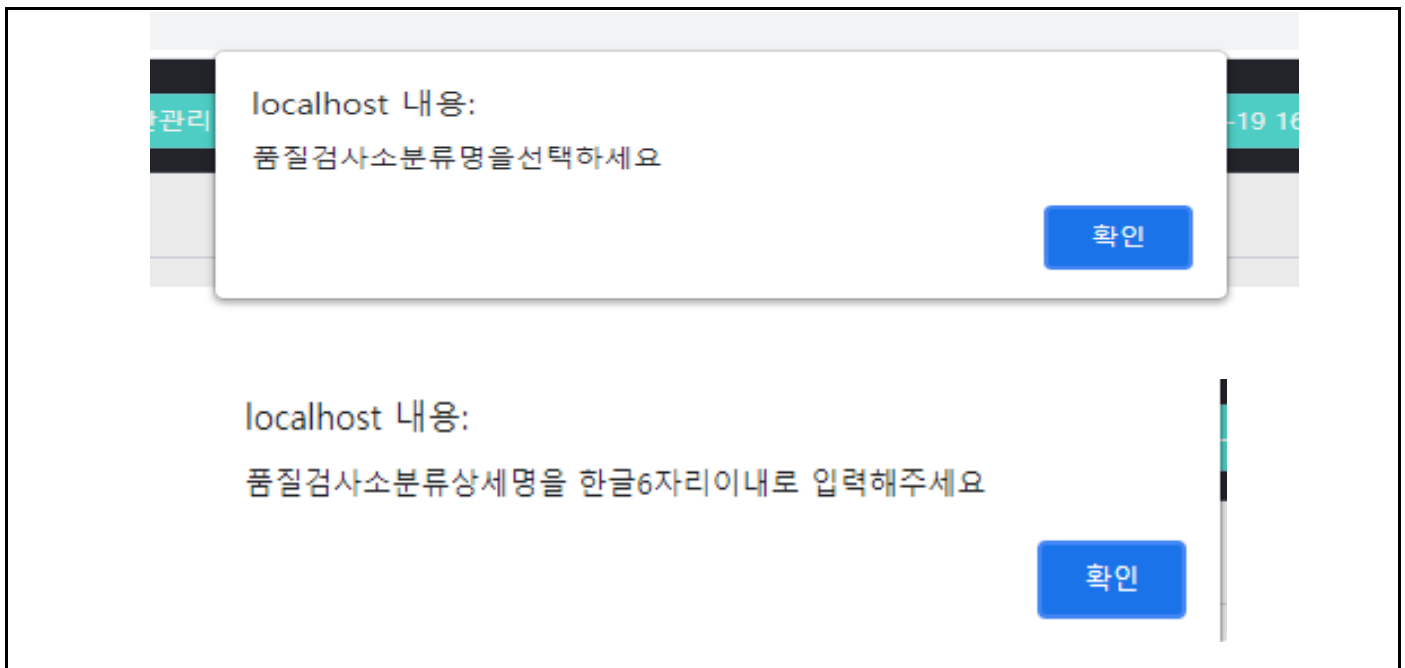
localhost 내용:
품질검사대분류명을선택하세요

확인

19

localhost 내용:
품질검사중분류명을선택하세요

확인



■ JavaScript

```
$(function(){
    var highClassCateName = $('#highClassCateName');
    var middleClassCateName = $('#middleClassCateName');
    var lowClassCateName = $('#lowClassCateName');
    var inspectionCode=[]
    //검사대분류명가지고오기
    var request = $.ajax({
        url: "/system/highClassCate",
        method: "get"

    });
    request.done(function( data ) {

        if(data != undefined || data != " " || data.length > 0){
            var html = "";
            for(var i = 0; i<data.length; i++){
                html += "<option value= ";
                html += data[i].highClassCode;
                html += ">";
                html += data[i].highClassName;
                html += "</option>";
            }
            highClassCateName.append(html);
        }

    });
    request.fail(function( jqXHR, textStatus ) {
        alert( "Request failed: " + textStatus );
    });
});
```

```
});
```

```
//중분류가지고오기
```

```
//대분류명 검사를 선택하게되면 data 화면에서 선택된 키와 값으로 data 를 전송  
$('#highClassCateName').change(function(){
```

```
    var highClassCateName = $('#highClassCateName option:selected').val();  
    inspectionCode.push(highClassCateName);  
    console.log(inspectionCode);
```

```
    var request = $.ajax({  
        url: "/system/mediumClassCate",  
        method: "get",  
        data: {highClassCateName : highClassCateName}  
    });
```

```
    request.done(function( data ) {  
        console.log(data);  
        if(data != undefined && data != "" && data.length > 0){  
            //선택박스 첫번째옵션빼고 초기화시키기  
            $('#middleClassCateName').find('option:not(:first)').remove();  
            var html = "";  
            for(var i = 0; i<data.length; i++){  
                html += "<option value=" + data[i].highMediumClassCode;  
                html += data[i].mediumClassName;  
                html += ">";  
            }  
            middleClassCateName.append(html);  
        }  
    });
```

```
    request.fail(function( jqXHR, textStatus ) {  
        alert( "Request failed: " + textStatus );  
    });
```

```
});
```

```
//중분류 검사를 선택하게되면 data 화면에서 선택된 키와 값으로 data 를 전송  
$('#middleClassCateName').change(function(){
```

```
    var middleClassCateName = $('#middleClassCateName option:selected').val();  
    inspectionCode.push(middleClassCateName);  
    console.log(inspectionCode);
```

```
    var request = $.ajax({  
        url: "/system/lowClassCate",  
        method: "get",  
        data: {middleClassCateName : middleClassCateName}  
    });
```

```
    request.done(function( data ) {  
        console.log(data);
```

```

        //선택박스 첫번째옵션빼고 초기화시키기
        $('#lowClassCateName').find('option:not(:first)').remove();
        if(data != undefined && data != "" && data.length > 0){
            var html = "";
            for(var i = 0; i<data.length; i++){
                html += "<option value= ";
                html += data[i].highMedLowClassCode;
                html += ">";
                html += data[i].lowClassName;
                html += "</option>";
            }
            lowClassCateName.append(html);
        }

    });

    request.fail(function( jqXHR, textStatus ) {
        alert( "Request failed: " + textStatus );
    });
})

$('#lowClassCateName').change(function(){
    var lowClassCateName = $('#lowClassCateName option:selected').val();
    $('#qualityinsertForm input').prop('disabled', false);
    inspectionCode.push(lowClassCateName);
    console.log(inspectionCode);
});
})

```

Controller

```

// ]품질검사 대분류 ajax 처리
@RequestMapping(value = "/highClassCate" ,method = RequestMethod.GET)
@ResponseBody
public List<Map<String,Object>> getHighClassCate() {
    List<Map<String,Object>> highCate =systemService.getHighClassCate();
    log.info("=====");
    log.info("highCate {}",highCate);
    log.info("=====");
    return highCate;
}

// 품질검사 중분류 가지고오기
@RequestMapping(value = "/mediumClassCate" ,method = RequestMethod.GET)
@ResponseBody
public List<Map<String,Object>> getMediumClassCate(@RequestParam(value =
"highClassCateName", required = false)String highClassCateName) {
    List<Map<String,Object>> mediumCate =
systemService.getMediumClassCate(highClassCateName);
    log.info("=====");
    log.info("mediumCate {}",mediumCate);
    log.info("=====");

    return mediumCate;
}

//[ 품질검사 소분류 가지고오기
@RequestMapping(value = "/lowClassCate" ,method = RequestMethod.GET)
@ResponseBody
public List<Map<String,Object>> getLowClassCate(@RequestParam(value =
"middleClassCateName", required = false)String middleClassCateName) {
    List<Map<String,Object>> lowCate =
systemService.getLowClassCate(middleClassCateName);
    log.info("=====");
    log.info("mediumCate {}",lowCate);
    log.info("=====");

    return lowCate;
}

//품질검사 등록 메서드
@PostMapping("/addQualityInspection")
public String addQualityInspection(@RequestParam(name = "highClassCode",required
= false)String highClassCode,

    @RequestParam(name = "highMedClassCode",required = false)String highMedClassCode,

    @RequestParam(name = "highMedLowClassCode",required = false)String
highMedLowClassCode,

```

```

@RequestParam(name = "subClassName",required = false)String subClassName) {
    log.info("=====");
    log.info("highClassCode {}:",highClassCode);
    log.info("highMedClassCode {}:",highMedClassCode);
    log.info("highMedLowClassCode {}:",highMedLowClassCode);
    log.info("subClassName {}:",subClassName);
    log.info("=====");
    systemService.addQualityInspection(highClassCode, highMedClassCode,
highMedLowClassCode, subClassName);
    return "redirect:/system/qualityInspectionList";
}

```

■ Service

//품질검사 등록

```

public int addQualityInspection(String highClassCode,String highMedClassCode,String
highMedLowClassCode,String subClassName) {
    Map<String, Object> qualityMap = new HashMap<String,Object>();
    qualityMap.put("highClassCode", highClassCode);
    qualityMap.put("highMedClassCode", highMedClassCode);
    qualityMap.put("highMedLowClassCode", highMedLowClassCode);
    qualityMap.put("subClassName", subClassName);
    log.info("qualityMap : {}", qualityMap);
    return systemMapper.addQualityInspection(qualityMap);
}

```

//품질검사 등록 대분류

```

public List<Map<String,Object>> getHighClassCate(){
    return systemMapper.getHighClassCate();
}

```

/]품질 검사 등록 중분류

```

public List<Map<String,Object>> getMediumClassCate(String highClassCateName){
    return systemMapper.getMediumClassCate(highClassCateName);
}

```

//품질검사 소분류

```

public List<Map<String,Object>> getLowClassCate(String middleClassCateName){
    return systemMapper.getLowClassCate(middleClassCateName);
}

```



```

<!-- 품질검사대분류카테고리 -->
<select id="getHighClassCate" resultType="map">
    SELECT
        h.high_class_code AS highClassCode,
        h.high_class_name AS highClassName
    FROM
        high_class_inspection_list AS h
    INNER JOIN
        medium_class_inspection_list AS m
    ON h.high_class_code=m.high_class_code
    INNER JOIN
        low_class_inspection_list AS l
    ON m.high_med_class_code= l.high_med_class_code
    INNER JOIN
        sub_class_inspection_list AS s
    ON l.high_med_low_class_code =s.high_med_low_class_code
    GROUP BY h.high_class_code;
</select>
<!-- 품질검사중분류카테고리 -->
<select id="getMediumClassCate" parameterType="String" resultType="map">
    SELECT
        h.high_class_code AS highClassCode,
        h.high_class_name AS highClassName,
        m.high_med_class_code AS highMediumClassCode,
        m.med_class_code AS mediumClassCode,
        m.med_class_name AS mediumClassName
    FROM
        high_class_inspection_list AS h
    INNER JOIN
        medium_class_inspection_list AS m
    ON h.high_class_code=m.high_class_code
    INNER JOIN
        low_class_inspection_list AS l
    ON m.high_med_class_code= l.high_med_class_code
    INNER JOIN
        sub_class_inspection_list AS s
    ON l.high_med_low_class_code =s.high_med_low_class_code
    WHERE h.high_class_code =#{highClassCateName}
    GROUP BY m.high_med_class_code;
</select>
<!-- 품질검사소분류카테고리 -->
<select id="getLowClassCate" parameterType="String" resultType="map">
    SELECT
        h.high_class_code AS highClassCode,
        h.high_class_name AS highClassName,

```

```

        m.high_med_class_code AS highMediumClassCode,
        m.med_class_code AS mediumClassCode,
        m.med_class_name AS mediumClassName,
        l.high_med_low_class_code AS highMedLowClassCode,
        l.low_class_code AS lowClassCode,
        l.low_class_name AS lowClassName
FROM
    high_class_inspection_list AS h
INNER JOIN
    medium_class_inspection_list AS m
ON h.high_class_code=m.high_class_code
INNER JOIN
    low_class_inspection_list AS l
ON m.high_med_class_code= l.high_med_class_code
INNER JOIN
    sub_class_inspection_list AS s
ON l.high_med_low_class_code =s.high_med_low_class_code
WHERE
    m.high_med_class_code=#{middleClassCateName}
GROUP BY l.high_med_low_class_code
</select>
<!--품질검사 등록 -->
<insert id="addQualityInspection" parameterType="map">
    INSERT INTO sub_class_inspection_list
        (quality_inspection_code,
        high_med_low_class_code,
        high_med_class_code,
        high_class_code,
        sub_class_code,
        sub_class_name,
        sub_class_reg_date,
        sub_class_update_date)
    VALUES (sf_QualityInspectionCode("#{highMedLowClassCode}"),
        #{highMedLowClassCode},
        #{highMedClassCode},
        #{highClassCode},
        sf_subInspectionCode(),
        #{subClassName},
        CURDATE(),
        CURDATE())
</insert>

```

품질검사수정

품질검사 목록에서 품질검사 수정 버튼을 누르면 품질검사 코드값을 가지고 품질검사 수정화면으로 이동합니다 품질검사 상세분류명을 수정하고 수정버튼을 누르면 품질검사 목록으로 화면 전환이됩니다.

TEAMONEMES

HOME1.수/발주관리2.차재관리3.생산관리4.품질관리5.출하관리6.가조정보관리

2021-07-21 16:10:00관리자CEO 임원 이현빛님이 로그인 하셨습니다

시스템관리

사용자관리 [한빛]기래서관리 [한빛]회계기조관리 [다미]자재기조관리 [민아]생산공정관리 [민아]품질기조관리 [보람]

품질검사수정

품질검사코드A02D050

품질검사대분류명공정중간검사

품질검사중분류명원단검사

품질검사 소분류명원단오염

품질검사 상세분류명오염이다

수정완료수정취소

품질검사수정

품질검사코드A02D050

품질검사대분류명공정중간검사

품질검사중분류명원단검사

품질검사 소분류명원단오염

품질검사 상세분류명오염이1

수정완료수정취소

50A02D050공정중간검사원단검사원단오염오염이12021-07-22 09:00:00기준표등록수정

✓검사등록/검사요청/검사삭제

■ Controller

//[보람]검사 수정 완료

```
@PostMapping("/modifyQualityInspection")
public String modifyQualityInspection(SubClassInspection subClassInspection) {
    log.info("=====");
    log.info("화면에서 입력받은 값(수정화면폼)subClassInspection:{},subClassInspection);
    log.info("=====");

    systemService.modifyQualityInspection(subClassInspection);
    return "redirect:/system/qualityInspectionList";
}
```

//[보람] 검사 수정 경로

```
@GetMapping("/modifyQualityInspection")
public String modifyQualityInspection(Model model
    ,@RequestParam(name ="qualityInspectionCode",required = false) String
qualityInspectionCode) {
    log.info("=====");
    log.info("화면에서입력받은검사수정폼 qualityInspectionCode:{},qualityInspectionCode);
    log.info("=====");
    //1.검사 코드로 검사객체조회한 SubClassInspection 객체
    SubClassInspection subClassInspection
=systemService.getQualityInspectionCode(qualityInspectionCode);

    //2.Model 화면에 객체 삽입
    model.addAttribute("subClassInspection", subClassInspection);
    log.info("subClassInspection:{},subClassInspection);

    return "system/modifyQualityInspection";
}
```

■ Service

//[보람]품질검사 상세 수정 메서드

```
public int modifyQualityInspection(SubClassInspection subClassInspection) {
    return systemMapper.modifyQualityInspection(subClassInspection);
}
```

//[보람] 품질검사 한행을 가지고오는메서드

```
public SubClassInspection getQualityInspectionCode(String qualityInspectionCode) {
    return systemMapper.getQualityInspectionCode(qualityInspectionCode);
}
```

■ SQL Mapper

//품질검사 dto

```
<resultMap type="SubClassInspection" id="subClassInspectionMap">
    <result property="qualityInspectionCode" column="quality_inspection_code"/>
    <result property="highClassCode" column="high_class_code"/>
    <result property="highMedClassCode" column="high_med_class_code"/>
    <result property="highMedLowClassCode" column="high_med_low_class_code"/>
    <result property="subClassCode" column="sub_class_code"/>
    <result property="subClassName" column="sub_class_name"/>
    <result property="subClassRegDate" column="sub_class_reg_date"/>
    <association property="highClassInspection" javaType="highClassInspection">
    <result column="high_class_name" property="highClassName"/>
    </association>
    <association property="mediumClassInspection" javaType="mediumClassInspection">
    <result column="med_class_name" property="medClassName"/>
    </association>
    <association property="lowClassInspection" javaType="lowClassInspection">
    <result column="low_class_name" property="lowClassName"/>
    </association>
</resultMap>
```

//품질검사 코드 가져오기

```
<select id="getQualityInspectionCode" parameterType="String"
resultMap="subClassInspectionMap">
    SELECT
        s.quality_inspection_code,
        h.high_class_name,
        m.med_class_code,
        m.med_class_name,
        l.low_class_name,
        s.sub_class_name,
        s.sub_class_reg_date
    FROM sub_class_inspection_list AS s
    INNER JOIN
        low_class_inspection_list AS l
        ON s.high_med_low_class_code = l.high_med_low_class_code
    INNER JOIN
        medium_class_inspection_list AS m
        ON s.high_med_class_code = m.high_med_class_code
    INNER JOIN
        high_class_inspection_list AS h
        ON s.high_class_code = h.high_class_code
    where  quality_inspection_code=#{qualityInspectionCode};
</select>
```

//품질검사 수정

```
<update id="modifyQualityInspection" parameterType="SubClassInspection">
    UPDATE sub_class_inspection_list AS s
        inner join
            low_class_inspection_list AS l
            ON s.high_med_low_class_code = l.high_med_low_class_code
    INNER JOIN
```

```

        medium_class_inspection_list AS m
        ON s.high_med_class_code =m.high_med_class_code
        INNER JOIN
        high_class_inspection_list AS h
        ON s.high_class_code = h.high_class_code
        <trim prefix="SET" prefixOverrides=",">
            <if test="highClassInspection.highClassName != null and
highClassInspection.highClassName != ".toString()">
                ,h.high_class_name=#{highClassInspection.highClassName}
            </if>
            <if test="mediumClassInspection.medClassName != null and
mediumClassInspection.medClassName != ".toString()">
                ,m.med_class_name=#{mediumClassInspection.medClassName}
            </if>
            <if test="lowClassInspection.lowClassName != null and
lowClassInspection.lowClassName != ".toString()">
                ,l.low_class_name=#{lowClassInspection.lowClassName}
            </if>
            <if test="subClassName != null and subClassName != ".toString()">
                ,s.sub_class_name =#{subClassName}
            </if>
        </trim>
        where
        s.quality_inspection_code=#{qualityInspectionCode};
    </update>

```

품질검사목록 조회

품질검사 목록에서 품질검사 조회를 하게 되면 품질검사 대분류 중분류 소분류 명을 선택하고 검색 버튼을 누르면 소분류검사에 따른 품질검사 소분류상세 검사 목록이 검색이됩니다.

시스템관리

사용자관리 [한빛]

거래처관리 [한빛]

외계기조관리 [다미]

자재기조관리 [민아]

생산공정관리 [민아]

품질기조관리 [보람]

검사대분류명

검사중분류명

검사소분류명

대분류 검사

중분류검사

소분류검사

검색

>품질검사목록

<input type="checkbox"/>	순번	검사코드	검사대분류명	검사중분류명	검사소분류명	검사 소분류상세명	등록날짜	기준표	수정버튼
<input type="checkbox"/>	1	A01B001C001D001	원자재검사	원단 이화학검사(파괴검사)	물세탁	변퇴색	2018-02-13 00:32:40.0	기준표등록	수정
<input type="checkbox"/>	2	A01B001C003D004	원자재검사	원단 이화학검사(파괴검사)	마찰변색도	견	2018-02-13 00:33:12.0	기준표등록	수정
<input type="checkbox"/>	3	A01B001C004D006	원자재검사	원단 이화학검사(파괴검사)	땀,일광복합견뢰도	일광복합견뢰도	2018-02-13 00:33:12.0	기준표등록	수정
<input type="checkbox"/>	4	A01B001C002D003	원자재검사	원단 이화학검사(파괴검사)	드라이클리닝_염색성	자체이염	2018-02-13 00:33:12.0	기준표등록	수정
<input type="checkbox"/>	5	A01B001C004D005	원자재검사	원단 이화학검사(파괴검사)	땀,일광복합견뢰도	땀견뢰도	2018-02-13 00:33:12.0	기준표등록	수정
<input type="checkbox"/>	6	A01B001C005D007	원자재검사	원단 이화학검사(파괴검사)	마찰견뢰도	석유계	2018-02-13 00:33:12.0	기준표등록	수정
<input type="checkbox"/>	7	A01B001C001D002	원자재검사	원단 이화학검사(파괴검사)	물세탁	직물 외관변화	2018-02-13 00:33:12.0	기준표등록	수정
<input type="checkbox"/>	8	A01B002C006D009	원자재검사	부자재외관검사	외관검사	스냅코팅불량	2018-02-15 00:34:17.0	기준표등록	수정
<input type="checkbox"/>	9	A01B002C006D011	원자재검사	부자재외관검사	외관검사	단추깨짐불량	2018-02-15 00:34:17.0	기준표등록	수정
<input type="checkbox"/>	10	A01B002C006D013	원자재검사	부자재외관검사	외관검사	지퍼도금불량	2018-02-15 00:34:17.0	기준표등록	수정

검사대분류명

검사중분류명

검사소분류명

원자재검사

중분류검사

소분류검사

검색

>원자재검사
원자재검사
공정중간검사
완제품검사

검사대분류명

검사중분류명

검사소분류명

원자재검사

중분류검사

소분류검사

검색

>품질검사목록

중분류검사명
원단 이화학검사(파괴검사)
부자재외관검사
부자재 기능검사

검사대분류명

검사중분류명

검사소분류명

원자재검사

원단 이화학

소분류검사

검색

>품질검사목록

<input type="checkbox"/>	순번	검사코드	검사대분류명	검사중분류명	검사소분류명	검사 소분류상세명	등록날짜	기준표	수정버튼
--------------------------	----	------	--------	--------	--------	-----------	------	-----	------

소분류검사명
물세탁
드라이클리닝_염색성
마찰변색도
땀,일광복합견뢰도
마찰견뢰도

>품질검사조회

검사대분류명

검사중분류명

검사소분류명

원자재검사

원단 이화학

물세탁

검색

>품질검사목록

<input type="checkbox"/>	순번	검사코드	검사대분류명	검사중분류명	검사소분류명	검사 소분류상세명	등록날짜	기준표	수정버튼
<input type="checkbox"/>	1	A01B001C001D001	원자재검사	원단 이화학검사(파괴검사)	물세탁	변퇴색	2018-02-12 15:32:40	기준표등록	수정
<input type="checkbox"/>	2	A01B001C001D002	원자재검사	원단 이화학검사(파괴검사)	물세탁	직물 외관변화	2018-02-12 15:33:12	기준표등록	수정

✓검사등록 수정 ✕검사요청 ✕검사삭제

■ JavaScript

//품질검사 항목조회

```
$('#inspectionSearchBtn').click(function(){
```

```
    var html =";
```

```
    //id 값의 value 값 Array 에 담기
```

```
    var inspectionArray ={
```

```
        highClassCateName : $('#highClassCateName').val(),
```

```
        middleClassCateName : $('#middleClassCateName').val(),
```

```
        lowClassCateName : $('#lowClassCateName').val(),
```

```
    };
```

```
    console.log(inspectionArray);
```

```
    //검색 ajax 전달
```

```
    var request = $.ajax({
```

```
        url: "/system/searchInspectionList",
```

```
        method: "GET",
```

```
        type: "post",
```

```
        data: inspectionArray,
```

```
        dataType: "json"
```

```
    });
```

```
    request.done(function( data ) {
```

```
        console.log(data);
```

```
        $( '#inspectionList > tbody').empty();
```

```
        //검색성공시 검사목록리시트를 tbody 아래는 비우고 반복을돌려서 검색된
```

```
        결과값을가지고오기
```

```
        if(data.length > 0){
```

```
            for(i=0; i<data.length; i++){
```

```
                html += "<tr>";
```

```
                html += "<td> <input type='checkbox' class='check' "
```

```
name='check' value="+ data[i].quality_inspection_code + "></td>";
```

```
                html += "<td>"+[i + 1]+"></td>";
```

```
                html += "<td>"+ data[i].quality_inspection_code+"></td>";
```

```
                html += "<td>"+ data[i].high_class_name+"></td>";
```

```
                html += "<td>"+ data[i].med_class_name+"></td>";
```

```
                html += "<td>"+ data[i].low_class_name+"></td>";
```

```
                html += "<td>"+ data[i].sub_class_name+"></td>";
```

```
                html += "<td>"+ data[i].inspection_reg_date+"></td>";
```

```
                html += "<td><a href='/quality/addStandardTable' class='btn btn-
```

```
warning btn-xs'><i class='fa fa-pencil'></i>기준표등록</a></td>";
```

```
                html += "<td><a
```

```
href='/system/modifyQualityInspection?qualityInspectionCode="+data[i].quality_inspection_code+" "
```

```
class='btn btn-primary btn-xs'><i class='fa fa-pencil'></i>수정</a></td>";
```

```
            }
```

```
        }else{
```

```
            html += "<tr><td colspan='11' style='text-align: center;'>
```

```
            검색된 결과가 없습니다. </td></tr>";
```

```
        }
```



```

        $('#inspectionTbody').append(html);
    });

    request.fail(function( jqXHR, textStatus ) {
        alert( "Request failed: " + textStatus );
    });

});

```

■ Controller

```

//[보람]품질검사 항목 검색
@RequestMapping(value = "/searchInspectionList",method = RequestMethod.GET)
@ResponseBody
public List<Map<String,Object>> searchInspectionList(@RequestParam(value
="highClassCateName",required = false )String highClassCateName,

                @RequestParam(value = "middleClassCateName",required = false)String
middleClassCateName,

                @RequestParam(value = "lowClassCateName",required = false)String
lowClassCateName

    ){
    //RequeusetParam 의 값을 map 을 담기
    HashMap<String, Object> map =new HashMap<String,Object>();
    map.put("highClassCateName", highClassCateName);
    map.put("middleClassCateName", middleClassCateName);
    map.put("lowClassCateName", lowClassCateName);
    List<Map<String,Object>> searchListMap= systemService.searchInspectionList(map);

    return searchListMap;
}

//[보람] 검사 리스트
@GetMapping("/qualityInspectionList")
public String getQualityInspectionList(Model model) {
    List<Map<String, Object>> resultMap = systemService.getQualityInspectionList();

    log.info("=====");
    log.info("qualityInspectionList - {}", resultMap.toString());
    log.info("=====");
    model.addAttribute("qualityInspectionList", resultMap);

    return"system/qualityInspectionList";
}

//[보람]품질검사 대분류 ajax 처리
@RequestMapping(value = "/highClassCate" ,method = RequestMethod.GET)

```

```

@ResponseBody
public List<Map<String,Object>> getHighClassCate() {
    List<Map<String,Object>> highCate =systemService.getHighClassCate();
    log.info("=====");
    log.info("highCate {}",highCate);
    log.info("=====");
    return highCate;
}
//[보람] 품질검사 중분류 가지고오기
@RequestMapping(value = "/mediumClassCate" ,method = RequestMethod.GET)
@ResponseBody
public List<Map<String,Object>> getMediumClassCate(@RequestParam(value =
"highClassCateName", required = false)String highClassCateName) {
    List<Map<String,Object>> mediumCate =
systemService.getMediumClassCate(highClassCateName);
    log.info("=====");
    log.info("mediumCate {}",mediumCate);
    log.info("=====");

    return mediumCate;
}
//[보람] 품질검사 소분류 가지고오기
@RequestMapping(value = "/lowClassCate" ,method = RequestMethod.GET)
@ResponseBody
public List<Map<String,Object>> getLowClassCate(@RequestParam(value =
"middleClassCateName", required = false)String middleClassCateName) {
    List<Map<String,Object>> lowCate =
systemService.getLowClassCate(middleClassCateName);
    log.info("=====");
    log.info("mediumCate {}",lowCate);
    log.info("=====");

    return lowCate;
}

```

■ Service

```

//[보람]품질검사 리스트 조회메서드
public List<Map<String, Object>> getQualityInspectionList() {
    return systemMapper.getQualityInspectionList();
}
//[보람]품질검사 등록 대분류
public List<Map<String, Object>> getHighClassCate(){
    return systemMapper.getHighClassCate();
}
//[보람]품질 검사 등록 중분류
public List<Map<String, Object>> getMediumClassCate(String highClassCateName){
    return systemMapper.getMediumClassCate(highClassCateName);
}
//[보람]품질검사 소분류
public List<Map<String, Object>> getLowClassCate(String middleClassCateName){
    return systemMapper.getLowClassCate(middleClassCateName);
}
//[보람]품질검사 조회목록
public List<Map<String, Object>>searchInspectionList(HashMap map){
    return systemMapper.searchInspectionList(map);
}

```

■ SQL Mapper

```

<select id="searchInspectionList" parameterType="hashmap" resultType="map">
    SELECT
        s.quality_inspection_code,
        h.high_class_code,
        h.high_class_name,
        m.high_med_class_code,
        m.med_class_code,
        m.med_class_name,
        l.high_med_low_class_code,
        l.low_class_code,
        l.low_class_name,
        s.sub_class_code,
        s.sub_class_name,
        DATE_FORMAT(s.sub_class_reg_date,'%Y-%m-%d %T') AS
inspection_reg_date
    FROM sub_class_inspection_list AS s
    INNER JOIN
        low_class_inspection_list AS l
        ON s.high_med_low_class_code = l.high_med_low_class_code
    INNER JOIN
        medium_class_inspection_list AS m
        ON s.high_med_class_code =m.high_med_class_code
    INNER JOIN
        high_class_inspection_list AS h
        ON s.high_class_code = h.high_class_code
    <trim prefix="WHERE" prefixOverrides="AND">

```

```

        <if test="highClassCateName != null and highClassCateName != ".toString()">

            AND h.high_class_code=#{highClassCateName}
        </if>
        <if test="middleClassCateName != null and middleClassCateName !=
".toString()">

            AND m.high_med_class_code=#{middleClassCateName}
        </if>
        <if test="lowClassCateName != null and lowClassCateName != ".toString()">
            AND l.high_med_low_class_code=#{lowClassCateName}
        </if>
        </trim>
    ORDER BY sub_class_reg_date;
</select>

<!-- 대분류카테고리 -->
<select id="getHighClassCate" resultType="map">
    SELECT
        h.high_class_code AS highClassCode,
        h.high_class_name AS highClassName
    FROM
        high_class_inspection_list AS h
    INNER JOIN
        medium_class_inspection_list AS m
    ON h.high_class_code=m.high_class_code
    INNER JOIN
        low_class_inspection_list AS l
    ON m.high_med_class_code= l.high_med_class_code
    INNER JOIN
        sub_class_inspection_list AS s
    ON l.high_med_low_class_code =s.high_med_low_class_code
    GROUP BY h.high_class_code;
</select>

<!-- 중분류카테고리 -->
<select id="getMediumClassCate" parameterType="String" resultType="map">
    SELECT
        h.high_class_code AS highClassCode,
        h.high_class_name AS highClassName,
        m.high_med_class_code AS highMediumClassCode,
        m.med_class_code AS mediumClassCode,
        m.med_class_name AS mediumClassName
    FROM
        high_class_inspection_list AS h
    INNER JOIN
        medium_class_inspection_list AS m
    ON h.high_class_code=m.high_class_code
    INNER JOIN
        low_class_inspection_list AS l
    ON m.high_med_class_code= l.high_med_class_code
    INNER JOIN
        sub_class_inspection_list AS s
    ON l.high_med_low_class_code =s.high_med_low_class_code

```

```

WHERE h.high_class_code =#{highClassCateName}
GROUP BY m.high_med_class_code;
</select>
<select id="getLowClassCate" parameterType="String" resultType="map">
    SELECT
        h.high_class_code AS highClassCode,
        h.high_class_name AS highClassName,
        m.high_med_class_code AS highMediumClassCode,
        m.med_class_code AS mediumClassCode,
        m.med_class_name AS mediumClassName,
        l.high_med_low_class_code AS highMedLowClassCode,
        l.low_class_code AS lowClassCode,
        l.low_class_name AS lowClassName
    FROM
        high_class_inspection_list AS h
    INNER JOIN
        medium_class_inspection_list AS m
    ON h.high_class_code=m.high_class_code
    INNER JOIN
        low_class_inspection_list AS l
    ON m.high_med_class_code= l.high_med_class_code
    INNER JOIN
        sub_class_inspection_list AS s
    ON l.high_med_low_class_code =s.high_med_low_class_code
    WHERE
        m.high_med_class_code=#{middleClassCateName}
    GROUP BY l.high_med_low_class_code
</select>

```

품질검사검사 삭제

품질검사 목록에서 잘못 등록된 품질검사 목록을 체크하여 삭제하기 버튼을 누르면 삭제하시겠습니까?라는 알림창이 뜨고 확인을 누르면 삭제 완료되었습니다 알림창이 뜨면서 목록에는 삭제된 검사항목은 삭제가 됩니다.

HOME	1.수/발주관리	2.자재관리	3.생산관리	4.품질관리	5.품역관리	6.기초정보관리	21-07-19 16:10:00	관리자CEO 임원 이현빛님이 로그인 하셨습니다.	Logout
<input type="checkbox"/>	38	A02B006C022D037	공정중간검사	봉제검사	하의치수	원단 좌우길이비교	2018-02-25 00:36:41.0	기준표등록	수정
<input type="checkbox"/>	39	A02B007C023D039	공정중간검사	완성검사	단추검사	단추달이 부착상태	2018-02-25 00:36:41.0	기준표등록	수정
<input type="checkbox"/>	40	A02B007C024D040	공정중간검사	완성검사	동작적합성	입고벗고편한정도	2018-02-25 00:39:49.0	기준표등록	수정
<input type="checkbox"/>	41	A03B008C025D041	완제품검사	착용검사	맞음새	전체길이적합성	2018-02-25 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	42	A03B009C026D042	완제품검사	기타검사	위싱불량	위싱면의뒤틀림	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	43	A03B010C027D044	완제품검사	봉제검사	상의 치수	총길이	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	44	A03B010C027D043	완제품검사	봉제검사	상의 치수	어깨넓이	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	45	A03B010C027D045	완제품검사	봉제검사	상의 치수	소매길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	46	A03B010C028D047	완제품검사	마무리	하의 치수	하의총길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	47	A03B010C028D046	완제품검사	봉제검사	하의 치수	바지의트임길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	48	A03B011C030D049	완제품검사	마무리	라벨	라벨표기사항확인	2018-03-04 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	49	A03B011C029D048	완제품검사	마무리	다림질	표면변색	2018-03-04 00:40:57.0	기준표등록	수정
<input checked="" type="checkbox"/>	50	A02D050	공정중간검사	원단검사	원단오염	오염이1	2021-07-22 09:00:00.0	기준표등록	수정

[✓검사등록](#) [🔍검사요청](#) [🔴검사사삭제](#)

HOME	1.수/발주관리	2.자재관리	3.생산관리	4.품질관리	5.품역관리	6.기초정보관리	21-07-19 16:10:00	관리자CEO 임원 이현빛님이 로그인 하셨습니다.	Logout
<input type="checkbox"/>	38	A02B006C022D037	공정중간검사	봉제검사	하의치수	원단 좌우길이비교	2018-02-25 00:36:41.0	기준표등록	수정
<input type="checkbox"/>	39	A02B007C023D039	공정중간검사	완성검사	단추검사	단추달이 부착상태	2018-02-25 00:36:41.0	기준표등록	수정
<input type="checkbox"/>	40	A02B007C024D040	공정중간검사	완성검사	동작적합성	입고벗고편한정도	2018-02-25 00:39:49.0	기준표등록	수정
<input type="checkbox"/>	41	A03B008C025D041	완제품검사	착용검사	맞음새	전체길이적합성	2018-02-25 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	42	A03B009C026D042	완제품검사	기타검사	위싱불량	위싱면의뒤틀림	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	43	A03B010C027D044	완제품검사	봉제검사	상의 치수	총길이	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	44	A03B010C027D043	완제품검사	봉제검사	상의 치수	어깨넓이	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	45	A03B010C027D045	완제품검사	봉제검사	상의 치수	소매길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	46	A03B010C028D047	완제품검사	마무리	하의 치수	하의총길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	47	A03B010C028D046	완제품검사	봉제검사	하의 치수	바지의트임길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	48	A03B011C030D049	완제품검사	마무리	라벨	라벨표기사항확인	2018-03-04 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	49	A03B011C029D048	완제품검사	마무리	다림질	표면변색	2018-03-04 00:40:57.0	기준표등록	수정
<input checked="" type="checkbox"/>	50	A02D050	공정중간검사	원단검사	원단오염	오염이1	2021-07-22 09:00:00.0	기준표등록	수정

[✓검사등록](#) [🔍검사요청](#) [🔴검사사삭제](#)

HOME	1.수/발주관리	2.자재관리	3.생산관리	4.품질관리	5.품역관리	6.기초정보관리	21-07-19 16:10:00	관리자CEO 임원 이현빛님이 로그인 하셨습니다.	Logout
<input type="checkbox"/>	38	A02B006C022D037	공정중간검사	봉제검사	하의치수	원단 좌우길이비교	2018-02-25 00:36:41.0	기준표등록	수정
<input type="checkbox"/>	39	A02B007C023D039	공정중간검사	완성검사	단추검사	단추달이 부착상태	2018-02-25 00:36:41.0	기준표등록	수정
<input type="checkbox"/>	40	A02B007C024D040	공정중간검사	완성검사	동작적합성	입고벗고편한정도	2018-02-25 00:39:49.0	기준표등록	수정
<input type="checkbox"/>	41	A03B008C025D041	완제품검사	착용검사	맞음새	전체길이적합성	2018-02-25 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	42	A03B009C026D042	완제품검사	기타검사	위싱불량	위싱면의뒤틀림	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	43	A03B010C027D044	완제품검사	봉제검사	상의 치수	총길이	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	44	A03B010C027D043	완제품검사	봉제검사	상의 치수	어깨넓이	2018-02-28 00:40:28.0	기준표등록	수정
<input type="checkbox"/>	45	A03B010C027D045	완제품검사	봉제검사	상의 치수	소매길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	46	A03B010C028D047	완제품검사	마무리	하의 치수	하의총길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	47	A03B010C028D046	완제품검사	봉제검사	하의 치수	바지의트임길이	2018-02-28 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	48	A03B011C030D049	완제품검사	마무리	라벨	라벨표기사항확인	2018-03-04 00:40:57.0	기준표등록	수정
<input type="checkbox"/>	49	A03B011C029D048	완제품검사	마무리	다림질	표면변색	2018-03-04 00:40:57.0	기준표등록	수정
<input checked="" type="checkbox"/>	50	A02D050	공정중간검사	원단검사	원단오염	오염이1	2021-07-22 09:00:00.0	기준표등록	수정

[✓검사등록](#) [🔍검사요청](#) [🔴검사사삭제](#)

■ JavaScript

```
$(function(){
    //checkAll 버튼 클릭 시 전체 checked
    $('#checkAll').click(function(){
        if($('#checkAll').prop('checked')){
            $('.check').prop('checked',true);
        }else{
            $('.check').prop('checked',false);
        }
    });
    //체크된 품질검사항목 삭제
    $('#delBtn').click(function(){
        var checkArray = new Array();
        var checkList = $('input[name= "check"]');
        //체크박스에 체크 된 체크박스의 값을 반복을 돌린다
        //체크된체크박스의값을 checkArray 에 담아준다.
        for(var i =0; i < checkList.length; i++){
            if(checkList[i].checked){
                checkArray.push(checkList[i].value);
            }
            console.log(checkArray);
        }
        //체크된 체크박스의 길이가 0 이면 품질검사항목이없다.
        if(checkArray.length == 0){
            alert("선택된 품질검사항목이 없습니다.");
        }else{
            var chk= confirm("정말로삭제하시겠습니까?");
            var request = $.ajax({
                typ:"POST",
                url:"/system/deleteInspection",
                data:{'checkArray':checkArray}
            });
            request.done(function(data) {
                if(data = 1){
                    alert("품질검사 항목이 삭제되었습니다.");
                    location.replace("qualityInspectionList")
                }else{
                    alert("품질검사항목 삭제 실패되었습니다.");
                }
            });
            request.fail(function( jqXHR, textStatus ) {
                alert( "Request failed: " + textStatus );
            });
        }
    });
});
```

■ Controller

```

//[보람]품질검사 삭제
@RequestMapping(value = "/deleteInspection")
@ResponseBody
public int deleteQualityInspection(@RequestParam(value = "checkArray[]")String[] checkArray) {
    int result =1;
    for(int i =0; i<checkArray.length; i++) {
        result = systemService.deleteQualityInspection(checkArray[i]);
    }
    log.info("=====");
    log.info("result {}",result);
    log.info("=====");
    //https://won-percent.tistory.com/48 참고한것
    return result;
}

```

■ Service

```

//[보람]품질검사 목록 삭제
public int deleteQualityInspection(String checkArray) {
    return systemMapper.deleteQualityInspection(checkArray);
}

```

■ SQL Mapper

```

<!--품질검사 삭제 -->
<delete id="deleteQualityInspection" parameterType="String">
DELETE
FROM
sub_class_inspection_list
WHERE quality_inspection_code =#{checkArray};

</delete>

```


6.3.2 자재관리 - 자재입고 등록

자재관리 자재입고 등록

원부자재 발주 코드를 검색해서 가져와서 실제로 입고되는 재고를 등록 한다.

TEAMONEMES

HOME 1.수령주요관리 2.자재관리 3.생산관리 4.물류관리 5.물류관리 6.기초정보관리 7.정보관리 21-07-19 16:18:00 관리자CEO 임원 이관철님의 로그인 하셨습니다. Logout

자재관리

자재입고 (관리)

자재입고 (조회)

자재입고 (관리)

자재 관리

자재 입고 등록

발주 코드 검색 검색하여 가져오기

원부자재 코드

재로명

색상

특징

발주 담당자 코드

발주 담당자명

수량

금액

입고일시

등록 취소

TEAMONEMES

자재관리

자재입고 (관리)

자재입고 (조회)

자재입고 (관리)

자재 관리

발주 목록 검색

발주 코드 검색어를 입력하세요.

거래처명 검색어를 입력하세요.

품목명 검색어를 입력하세요.

발주 날짜 연도-월-일 ~ 연도-월-일 검색

	no.	발주코드	원부자재 코드	거래처명	품목명	색상명	특징	수량	입고예정일	발주일자
<input checked="" type="radio"/>	1	123456	원부자재 코드	나는 단주	단주	발견색	특징1	100	2021-02-22	2021-02-10
<input type="radio"/>	1	123	원부자재 코드1	나는 단주1	단주1	발견색1	특징1	1001	2021-02-22	2021-02-10
<input type="radio"/>	1	1233	원부자재 코드2	나는 단주2	단주2	발견색2	특징2	1002	2021-02-22	2021-02-10
<input type="radio"/>	1	3333	원부자재 코드3	나는 단주3	단주3	발견색3	특징3	1003	2021-02-22	2021-02-10

Close Save changes

자재관리

자재입고 (관리)

자재출고 (관리)

자재정량 (관리)

자재 관리

자재 입고 등록

발주 코드 검색

123

검색하여 가져오기

원부자재 코드

원부자재 코드1

재료명

단조1

식상

불건적1

특징

특징1

발주 담당자 코드

발주 담당자명

갯수

1001

금액

1200000

입고일시

날짜

시간

등록

취소

6.3.3 자재관리 - 입고자재조회

자재관리 - 자재입고 조회

원부자재 입고 조회를 할 수 있다.

원부자재 코드를 누르면 해당 원부자재의 상세 정보가 뜬다.

TEAMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.물류관리

5.출하관리

6.기초정보관리

메뉴일시21-07-19 16:10:00

관리자CEO 임원 이원빛님이 로그인 하셨습니다.

Logout

자재관리

자재입고 (다미)

입고등록 (다미)

입고현황 (다미)

자재출고 (한빛)

자재현황 (한빛)

>자재 입고 조회

발주처별

재료별

입고일별

원부자재구분

직접 입력 또는 검색

직접 입력 또는 검색

연도-월-일

연도-월-일

All

검색

>자재 입고 목록

번호	원부자재 코드	원부자재 발주 코드	구분	재료명	수량	단위	입고일시	수정
1	button01_FF87D2	materialOrder000001	부자재	폴리에스터	90	개	2016-12-07 12:37:47	수정
2	button01_FF87D2	materialOrder000001	부자재	폴리에스터	20	개	2016-12-08 20:18:30	수정
3	button01_FF87D2	materialOrder000001	부자재	폴리에스터	30	개	2016-12-10 12:40:15	수정
4	button01_FF87D2	materialOrder000002	부자재	폴리에스터	100	개	2016-12-07 12:37:47	수정
5	fabric01_646EFF	materialOrder000003	원자재	제니스	18	마	2016-07-07 18:20:20	수정
6	fabric01_646EFF	materialOrder000003	원자재	제니스	2	마	2016-07-07 18:20:20	수정
7	fabric04_2478CD	materialOrder000004	원자재	VIP	20	마	2019-05-09 11:40:57	수정
8	thread01_E6A35A	materialOrder000005	부자재	명주실	15	kg	2020-03-08 14:18:18	수정
9	innerfabric02_E1F6FA	materialOrder000010	부자재	큐프라	15	마	2021-03-28 12:40:48	수정
10	button03_7A6F95	materialOrder000009	부자재	자개	15	개	2021-03-08 12:40:48	수정
11	innerfabric01_E1F6FA	materialOrder000008	부자재	폴리에스테르	15	마	2021-03-26 12:40:48	수정
12	fabric01_7878E1	materialOrder000007	원자재	제니스	30	마	2021-02-17 12:40:48	수정
13	interfacing02_DCE8FF	materialOrder000006	부자재	비접착식	15	m	2021-02-08 12:40:48	수정

입고 등록

출하

TEAMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.물류관리

5.출하관리

6.기초정보관리

메뉴일시21-07-19 16:10:00

관리자CEO 임원 이원빛님이 로그인 하셨습니다.

Logout

자재관리

자재입고 (다미)

자재출고 (한빛)

자재현황 (한빛)

>자재 입고 조회

발주처별

재료별

입고일별

원부자재구분

직접 입력 또는 검색

직접 입력

연도-월-일

All

검색

>자재 입고 목록

번호	원부자재 코드	원부자재 발주 코드	구분	재료명	수량	단위	입고일시	수정
1	button01_FF87D2	materialOrder000001	부자재	폴리에스터	90	개	2016-12-07 12:37:47	수정
2	button01_FF87D2	materialOrder000001	부자재	폴리에스터	20	개	2016-12-08 20:18:30	수정
3	button01_FF87D2	materialOrder000001	부자재	폴리에스터	30	개	2016-12-10 12:40:15	수정
4	button01_FF87D2	materialOrder000002	부자재	폴리에스터	100	개	2016-12-07 12:37:47	수정
5	fabric01_646EFF	materialOrder000003	원자재	제니스	18	마	2016-07-07 18:20:20	수정
6	fabric01_646EFF	materialOrder000003	원자재	제니스	2	마	2016-07-07 18:20:20	수정
7	fabric04_2478CD	materialOrder000004	원자재	VIP	20	마	2019-05-09 11:40:57	수정
8	thread01_E6A35A	materialOrder000005	부자재	명주실	15	kg	2020-03-08 14:18:18	수정
9	innerfabric02_E1F6FA	materialOrder000010	부자재	큐프라	15	마	2021-03-28 12:40:48	수정
10	button03_7A6F95	materialOrder000009	부자재	자개	15	개	2021-03-08 12:40:48	수정
11	innerfabric01_E1F6FA	materialOrder000008	부자재	폴리에스테르	15	마	2021-03-26 12:40:48	수정
12	fabric01_7878E1	materialOrder000007	원자재	제니스	30	마	2021-02-17 12:40:48	수정
13	interfacing02_DCE8FF	materialOrder000006	부자재	비접착식	15	m	2021-02-08 12:40:48	수정

입고 등록

출하

품목 상세 정보

자재코드

123456

자재명

폴리에스터

규격명

box

대분류

원자재

발주처

나는원단

발주단위

개

중분류

원단

배송요일

물 수 목

최소수량

1

소분류

실크

재고문명

TC

최대수량

99

입고예정(D+1)

0

입고예정(D+2)

0

6.3.4 생산관리 - 생산계획 등록, 수정

생산관리 - 생산계획 등록,수정

생산계획을 누르면 성별별 양복명과 품목 소분류가 ajax 호출을 통해 불러와진다.

품목 소분류를 선택하고 수량과 작업예상시작일과 예상종료일을 등록,수정할 수 있다.

TEAMONEMES

HOME1수/발주관리2차재관리3생산관리4품질관리5출하관리6기초정보관리메포일시21-07-19 16:10:00관리자CEO 임원 이원빛님이 로그인 하셨습니다Logout

생산관리

생산계획 등록 [다미]

생산계획 등록 [다미]

생산계획 조회/목록 [다미]

주간별 생산계획 조회 [다미]

월별 생산계획 조회 [다미]

작업 지시 [보람]

생산지시 [민아]

생산현황 [민아+관보]

원재료관리 [민아]

생산계획 등록

생산계획 등록

성별별 양복명--선택--

품목소분류--선택--

수량

단위--선택--

작업시작 예정일연도-월-일

작업마감 예정일연도-월-일

담당 사원명김다미

담당자 사원 코드E0011

비고

등록취소

TEAMONEMES

HOME1수/발주관리2차재관리3생산관리4품질관리5출하관리6기초정보관리메포일시21-07-19 16:10:00관리자CEO 임원 이원빛님이 로그인 하셨습니다Logout

생산관리

생산계획 등록 [다미]

생산계획 등록 [다미]

생산계획 조회/목록 [다미]

주간별 생산계획 조회 [다미]

월별 생산계획 조회 [다미]

작업 지시 [보람]

생산지시 [민아]

생산현황 [민아+관보]

원재료관리 [민아]

생산계획 등록

생산계획 등록

성별별 양복명신사양복

품목소분류--선택--

수량

단위--선택--

작업시작 예정일연도-월-일

작업마감 예정일연도-월-일

담당 사원명김다미

담당자 사원 코드E0011

비고

등록취소

TEAMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.물질관리

5.출하관리

6.기초정보관리

배달일시21-07-19 16:10:00

관리자CEO 임원 이원빛님이 로그인 하셨습니다.

Logout

생산관리

생산 계획 [다미]

작업 지시 [보람]

생산지시 [민아]

생산현황 [민아+현빛]

안재물관리 [민아]

>생산 계획 조회

양품명

소분류

계획일자

계획시작일 정렬

검색

일별 조회

주간별 조회

>생산 계획 목록

<input type="checkbox"/>	번호	생산계획 코드	생산품명	생산소분류	생산일자 사원명	생산량	생산단위	생산시작 예정일	생산종료 예정일	생산정
<input type="checkbox"/>	1	production_plan_01	신사양복	바지	이원빛	111	벌	2021-07-08	2021-07-19	<div>수정</div>
<input type="checkbox"/>	2	production_plan_02	신사양복	바지	김도연	200	벌	2017-12-06	2018-01-06	<div>수정</div>
<input type="checkbox"/>	3	production_plan_03	숙녀양복	블라우스	김도연	200	벌	2017-12-06	2018-01-06	<div>수정</div>
<input type="checkbox"/>	4	production_plan_04	숙녀양복	자켓	김도연	200	벌	2017-12-06	2018-01-06	<div>수정</div>
<input type="checkbox"/>	5	production_plan_05	신사양복	바지	박수현	128	벌	2019-04-12	2019-04-20	<div>수정</div>
<input type="checkbox"/>	6	production_plan_06	숙녀양복	블라우스	박수현	128	벌	2019-04-12	2019-04-23	<div>수정</div>
<input type="checkbox"/>	7	production_plan_07	숙녀양복	자켓	박수현	128	벌	2019-04-12	2019-04-25	<div>수정</div>
<input type="checkbox"/>	8	production_plan_08	숙녀양복	치마	한보람	50	벌	2019-06-20	2019-06-25	<div>수정</div>
<input type="checkbox"/>	9	production_plan_09	신사양복	자켓	박수현	120	벌	2021-07-01	2021-07-31	<div>수정</div>
<input type="checkbox"/>	10	production_plan_11	숙녀양복	치마	류준혁	200	벌	2021-06-01	2021-08-31	<div>수정</div>
<input type="checkbox"/>	11	production_plan_12	신사양복	바지	류준혁	50	벌	2021-07-18	2021-07-24	<div>수정</div>
<input type="checkbox"/>	12	production_plan_13	신사양복	자켓	이원빛	4	벌	2021-07-11	2021-07-31	<div>수정</div>

등록

휴식재

TEAMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.물질관리

5.출하관리

6.기초정보관리

배달일시21-07-19 16:10:00

관리자CEO 임원 이원빛님이 로그인 하셨습니다.

Logout

생산관리

생산 계획 [다미]

작업 지시 [보람]

생산지시 [민아]

생산현황 [민아+현빛]

안재물관리 [민아]

생산계획 등록

생산계획 수정

성분품 양품명

신사양복

품목소분류

바지

수량

111

단위

벌

작업시작 예정일

2021-07-08

작업마감 예정일

2021-07-19

수정자 사원명

김다미

수정자 사원 코드

E0001

비고

등록

취소

6.3.5 생산관리 - 생산계획 조회

생산계획

월간, 주간별로 생산계획을 확인 할 수 있다.

TEAMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.품질관리

5.출하관리

6.기초정보관리

🕒 배포일시:21-07-19 16:10:00

관리자CEO 임원 이한빛

생산관리

📄 생산 계획 [[다미]]

📄 작업 지시 [[보람]]

🏠 생산지시 [[민아]]

🏠 생산현황 [[민아+한빛]]

📄 완제품관리 [[민아]]

>주간별 생산 계획

< > today

month week day

Sun 7/25	Mon 7/26	Tue 7/27	Wed 7/28	Thu 7/29	Fri 7/30	Sat 7/31
숙녀양복/지마						
신사양복/자켓						
신사양복/자켓						

TEAMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.품질관리

5.출하관리

6.기초정보관리

🕒 배포일시:21-07-19 16:10:00

관리자CEO 임원 이한빛

생산관리

📄 생산 계획 [[다미]]

📄 작업 지시 [[보람]]

🏠 생산지시 [[민아]]

🏠 생산현황 [[민아+한빛]]

📄 완제품관리 [[민아]]

>월별 생산 계획

< > today

month week day

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	1	2	3
숙녀양복/지마						
				신사양복/자켓		
4	5	6	7	8	9	10
숙녀양복/지마						
신사양복/자켓				신사양복/바지		
11	12	13	14	15	16	17
숙녀양복/지마						
신사양복/자켓						
신사양복/바지						
신사양복/자켓						
18	19	20	21	22	23	24
숙녀양복/지마						
신사양복/자켓						
신사양복/바지						
신사양복/자켓						
신사양복/바지						
25	26	27	28	29	30	31
숙녀양복/지마						
신사양복/자켓						
신사양복/자켓						
1	2	3	4	5	6	7

```

var genderCate = $('#genderCate');
var detailCate = $('#detailCate');
    //성별별 이름 가져오기
var request = $.ajax({
    url: "/production/productCode",
    method: "get"
});
request.done(function( data ) {
    console.log(data);
    if(data != undefined || data != "" || data.length > 0){
        var html = "";
        for(var i = 0; i<data.length; i++){
            html += "<option value= ";
            html += data[i].gender_categorized_name;
            html += ">";
            html += data[i].gender_categorized_name+"</option>";

        }
        genderCate.append(html);
    }
});
request.fail(function( jqXHR, textStatus ) {
    alert( "Request failed: " + textStatus );
});

//소분류 가져오기
$('#genderCate').change(function(){

    var gender = $("#genderCate option:selected").val();
    //옵션 값 초기화
    detailCate.html("<option value='>::선택::</option>");

    var request = $.ajax({
        url: "/production/detailCode",
        method: "get",
        data: {gender : gender}
    });
    request.done(function( data ) {
        console.log(data);
        if(data != undefined && data != "" && data.length > 0){
            var html = "";
            for(var i = 0; i<data.length; i++){
                html += "<option value= ";
                html += data[i].product_code;
                html += ">";
                html += data[i].detailed_categorized_name;
                html += "</option>";

            }

```

```

        detailCate.append(html);
    }
});
request.fail(function( jqXHR, textStatus ) {
    alert( "Request failed: " + textStatus );
});
});

```

■Controller

```

//[다미]생산계획 삭제
@PostMapping("/deleteProductionPlan")
@ResponseBody
public int deleteProductionPlan(@RequestParam(value = "delArr[]")String[] delArr) {
    int result = 1;

    for(int i = 0; i<delArr.length; i++) {
        result = productionPlanService.deleteProductionPlan(delArr[i]);
    }
    return result;
}

//[다미]생산계획 검색
@RequestMapping(value = "/searchProductionPlan", method = RequestMethod.GET)
@ResponseBody
public List<Map<String, Object>> searchProductionPlan(@RequestParam(value =
"genderCate", required = false) String genderCate
, @RequestParam(value =
"detailCate", required = false) String detailCate
, @RequestParam(value =
"startDate", required = false) String startDate
, @RequestParam(value =
"endDate", required = false) String endDate
, @RequestParam(value =
"range", required = false) String range) {
    HashMap<String, Object> map = new HashMap<>();
    map.put("genderCate", genderCate);
    map.put("detailCate", detailCate);
    map.put("startDate", startDate);
    map.put("endDate", endDate);
    map.put("range", range);
    List<Map<String, Object>> listMap =
productionPlanService.searchProductionPlan(map);
    log.info("{} ", listMap);
    return listMap;
}

```



```

//[다미]생산계획 수정
@PostMapping("/modifyProductionPlan")
public String modifyProductionPlan(ProductionPlan productionPlan) {
    productionPlanService.modifyProductionPlan(productionPlan);
    return "redirect:productionPlanList";
}

//[다미]생산계획 수정 폼
@GetMapping("/modifyProductionPlan")
public String modifyProductionPlan(@RequestParam(value = "planCode", required = false)String
planCode
                                                                    ,Model model) {
    Map<String, Object> resultMap =
productionPlanService.getProductionPlanListByCode(planCode);
    model.addAttribute("resultMap", resultMap);
    log.info("=====");
    log.info("계획코드로 조회한 값:          {}", resultMap);
    log.info("=====");
    return "production/modifyProductionPlan";
}

//[다미]생산계획 등록
@PostMapping("/addProductionPlan")
public String productioncontrolAdd(ProductionPlan productionPlan) {
    log.info("=====");
    log.info("화면에서 받아온 값:          {}", productionPlan);
    log.info("=====");

    productionPlanService.addProductionPlan(productionPlan);

    return "redirect:productionPlanList";
}

//[다미]성별별 양복명에 맞는 소분류 가져오기
@RequestMapping(value = "/detailCode", method = RequestMethod.GET)
@ResponseBody
public List<Map<String, Object>> getDetailCode(@RequestParam(value = "gender", required =
false)String gender){
    List<Map<String, Object>> name = productionPlanService.getDetailCode(gender);
    log.info("=====");
    log.info("화면에서 받아온 값:          {}", gender);
    log.info("생산코드 조회 :          {}", name);
    log.info("=====");

    return name;
}

//[다미]성별별 양복명 값 가져오기
@RequestMapping(value="/productCode", method = RequestMethod.GET)
@ResponseBody
public List<Map<String, Object>> getProductionCode() {

```

```

        List<Map<String, Object>> list = productionPlanService.getProductCode();
        log.info("=====");
        log.info("생산코드 조회 :      {}", list);
        log.info("=====");
        return list;
    }

//[다미]생산계획 주간별 목록
@GetMapping("/productionWeeklyPlanList")
public String productionWeeklyPlanList() {
    return "production/productionWeeklyPlanList";
}

//[다미]생산계획 월별 목록
@GetMapping("/productionMonthlyPlanList")
public String getProductionMonthlyPlanList(Model model) {

    return "production/productionMonthlyPlanList";
}

//[다미]생산계획 데이터 가져오기
@RequestMapping(value="/monthPlan", method = RequestMethod.GET)
@ResponseBody
public List<Map<String, Object>> monthPlan() {
    List<Map<String, Object>> list = productionPlanService.getProductionAllPlanList();

    JSONObject jsonObj = new JSONObject();
    JSONArray jsonArr = new JSONArray();
    HashMap<String, Object> hash = new HashMap<String, Object>();
    log.info("{} ",list);

    for(int i=0; i < list.size(); i++) {
        String genderCate = (String) list.get(i).get("gender_categorized_name");
        String cateName = (String) list.get(i).get("detailed_categorized_name");

        hash.put("title", genderCate + "/" + cateName);
        hash.put("start", list.get(i).get("expected_production_start_date"));
        hash.put("end", list.get(i).get("expected_production_end_date"));

        if(genderCate.equals("신사양복")) {
            hash.put("color", "#F08080");
        }else if(genderCate.equals("숙녀양복")){
            hash.put("color", "#006400");
        }
        jsonObj = new JSONObject(hash);
        jsonArr.add(jsonObj);
    }
    log.info("jsonArr{}", jsonArr);
    return jsonArr;
}

```

```

//[다미]생산계획 월별 목록 test
@ResponseBody
@RequestMapping(value="/production/productionMonthlyPlanList", method =
RequestMethod.POST)
public String getProductionMonthlyPlanList(){
    return "production/productionMonthlyPlanList";
}

//[다미]전체 생산계획 목록
@GetMapping("/productionPlanList")
public String productionPlanList(Model model) {
    List<Map<String, Object>> resultMap =
productionPlanService.getProductionAllPlanList();
    log.info("=====");
    log.info("전체 생산계획 목록:      {}", resultMap);
    log.info("=====");
    model.addAttribute("list", resultMap);
    return "production/productionPlanList";
}

//[다미]생산계획 등록
@GetMapping("/addProductionPlan")
public String productioncontrolAdd() {
    return "production/addProductionPlan";
}
}

```

■Service

```

//[다미]생산계획 삭제
public int deleteProductionPlan(String delArr) {
    return productionPlanMapper.deleteProductionPlan(delArr);
};

//[다미]생산계획 검색
public List<Map<String, Object>> searchProductionPlan(HashMap map){
    return productionPlanMapper.searchProductionPlan(map);
};

//[다미]코드로 검색
public Map<String, Object> getProductionPlanListByCode(String planCode){
    return productionPlanMapper.getProductionAllPlanList(planCode);
}

//[다미]생산계획 수정 등록
public int modifyProductionPlan(ProductionPlan productionPlan) {
    return productionPlanMapper.modifyProductionPlan(productionPlan);
};

//[다미]생산계획 전체 조회

```

```

public List<Map<String, Object>>getProductionAllPlanList(){
    return productionPlanMapper.getProductionAllPlanList();
};

//[다미]생산계획 등록
public int addProductionPlan(ProductionPlan productionPlan) {
    return productionPlanMapper.addProductionPlan(productionPlan);
};

//[다미]소분류 조회
public List<Map<String, Object>> getDetailCode(String gender){
    return productionPlanMapper.getDetailCode(gender);
}

//[다미]생산코드 조회
public List<Map<String, Object>> getProductCode(){
    return productionPlanMapper.getProductCode();
}

}

```

■SQL Mapper

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-
mapper.dtd">
<mapper namespace="ksmart39.springboot.dao.ProductionPlanMapper">

<resultMap type="ProductionPlan" id="getProductionPlan">
    <result property="productionPlanCode" column="production_plan_code"/>
    <result property="productCode" column="product_code"/>
    <result property="chargeEmployeeCode" column="charge_employee_code"/>
    <result property="productionPlanQuantity" column="production_plan_quantity"/>
    <result property="productionPlanQantityUnit" column="production_plan_qantity_unit"/>
    <result property="expectedProductionStartDate" column="expected_production_start_date"/>
    <result property="expectedProductionEndDate" column="expected_production_end_date"/>
    <result property="productionPlanPeriod" column="production_plan_period"/>
    <result property="productionPlanNote" column="production_plan_note"/>
    <result property="productionPlanRegDate" column="production_plan_reg_date"/>
    <result property="productionPlanUpdateDate" column="production_plan_update_date"/>
</resultMap>

    <delete id="deleteProductionPlan" parameterType="String">
        DELETE
        FROM
            production_plan
        WHERE
            production_plan_code=#{delArr};
    </delete>

```

```
<select id="searchProductionPlan" parameterType="hashmap" resultType="Map">
```

```
  SELECT
```

```
    p.production_plan_code
    ,d.gender_categorized_name
    ,d.detailed_categorized_name
    ,e.employee_name
    ,p.production_plan_period
    ,p.production_plan_quantity_unit
    ,p.expected_production_start_date
    ,p.expected_production_end_date
```

```
  FROM
```

```
    production_plan AS p
  JOIN
    product_code_detail AS d
  ON p.product_code = d.product_code
  JOIN
    employee_info AS e
  ON p.charge_employee_code = e.employee_code
```

```
  <trim prefix="WHERE" prefixOverrides="AND">
```

```
    <if test="genderCate != null and genderCate != ''">
```

```
      AND    d.gender_categorized_name = #{genderCate}
```

```
    </if>
```

```
    <if test="detailCate != null and detailCate != ''">
```

```
      AND    d.product_code = #{detailCate}
```

```
    </if>
```

```
    <if test="startDate != null and startDate != ''">
```

```
      AND    p.expected_production_start_date >= DATE(#{startDate})
```

```
    </if>
```

```
    <if test="endDate != null and endDate != ''">
```

```
      AND    p.expected_production_end_date <= DATE(#{endDate})
```

```
    </if>
```

```
  </trim>
```

```
  <trim prefix="ORDER BY" suffixOverrides="WHERE">
```

```
    p.expected_production_start_date ${range};
```

```
  </trim>
```

```
</select>
```

```
<update id="modifyProductionPlan" parameterType="ProductionPlan">
```

```
  UPDATE
```

```
    production_plan
```

```
  SET
```

```
    product_code=#{productCode}
    ,charge_employee_code=#{chargeEmployeeCode}
    ,production_plan_quantity=#{productionPlanQuantity}
    ,production_plan_quantity_unit=#{productionPlanQuantityUnit}
    ,expected_production_start_date=#{expectedProductionStartDate}
    ,expected_production_end_date=#{expectedProductionEndDate}
```

```
    ,production_plan_period=sf_productionPlanDate(#{expectedProductionStartDate},#{expectedProductionEndDate})
```

```
    ,production_plan_note=#{productionPlanNote}
```

```

        ,production_plan_update_date=NOW()
    WHERE
        production_plan_code=#{productionPlanCode};
</update>

<select id="getProductionAllPlanList" resultType="Map" parameterType="String">
    SELECT
        p.production_plan_code
        ,c.gender_categorized_name
        ,c.detailed_categorized_name
        ,e.employee_name
        ,p.production_plan_quantity
        ,p.production_plan_qantity_unit
        ,p.expected_production_start_date
        ,p.expected_production_end_date
    FROM
        production_plan AS p
    INNER JOIN
        product_code_detail AS c
    ON
        p.product_code = c.product_code
    INNER JOIN
        employee_info AS e
    ON
        p.charge_employee_code = e.employee_code
    <trim prefix="WHERE" prefixOverrides="AND |OR ">
        <if test="planCode != null and planCode != ''">
            p.production_plan_code = #{planCode};
        </if>
    </trim>
</select>

<insert id="addProductionPlan" parameterType="ProductionPlan">
    INSERT INTO production_plan
        (production_plan_code
        ,product_code
        ,charge_employee_code
        ,production_plan_quantity
        ,production_plan_qantity_unit
        ,expected_production_start_date
        ,expected_production_end_date
        ,production_plan_period
        ,production_plan_note
        ,production_plan_reg_date
        ,production_plan_update_date)
    VALUES
        (sf_productionPlanCode()
        ,#{productCode}
        ,#{chargeEmployeeCode}
        ,#{productionPlanQuantity}
        ,#{productionPlanQantityUnit}
        ,#{expectedProductionStartDate}

```

```
        ,#{expectedProductionEndDate}

    ,sf_productionPlanDate(#{expectedProductionStartDate},#{expectedProductionEndDate})
        ,#{productionPlanNote}
        ,NOW()
        ,NULL);

</insert>

<select id="getDetailCode" parameterType="String" resultType="Map">
    SELECT
        detailed_categorized_name
        ,product_code
    FROM
        product_code_detail
    WHERE
        gender_categorized_name = #{gender};
</select>

<select id="getProductCode" resultType="Map">
    SELECT
        DISTINCT(gender_categorized_name)
    FROM
        product_code_detail;
</select>
</mapper>
```

6.3.6 품질관리 - 품질검사 요청

품질관리 - 품질검사 요청

품질검사 요청을 할 때 계약번호를 검색해서 품목별 의뢰 코드를 가져온다.

품질검사 종류 대,중,소 분류는 ajax 호출을 통해 DB에 연결된 값을 가져온다.

TEAMONEMES

HOME1.수/발주관리2.자재관리3.생산관리4.품질관리5.출하관리6.기초정보관리

●테포일시21-07-19 16:10:00관리자

품질관리

기준표 [민아]

검사요청 [다미]

품질 검사 요청 [다미]

요청목록 [다미]

검사측정값 [다미&보람]

검사현황 [다미&보람]

불량품 [한빛]

품질 관리

품질 검사 요청

계약번호

검색하여 가져오기

품목별 의뢰 코드

대분류 검사명::선택::

중분류 검사명::선택::

소분류 검사명::선택::

소분류 검사 상세명::선택::

샘플링 검사 유무::선택::

특이사항

검사 완료 요청 시각

✕

📅

담당 사원명김다미

담당자 사원 코드W01

TEAMONEMES

HOME1.수/발주관리2.자재관리3.생산관리4.품질관리5.출하관리6.기초정보관리

●테포일시21-07-19 16:10:00관리자

품질관리

기준표 [민아]

검사요청 [다미]

품질 검사 요청 [다미]

요청목록 [다미]

검사측정값 [다미&보람]

검사현황 [다미&보람]

불량품 [한빛]

품질 관리

품질 검사 요청

계약번호

품목별 의뢰 코드

대분류 검사명::선택::

중분류 검사명::선택::

발주처 검색

거래처명 품목별 코드 품목명All 품목소분류명All

검색

✓no.계약번호거래처명품목별 코드품목명품목소분류명

○1123456나는 단추req01_MS_pants남성양복바지

Close

Save changes

<p>품질관리 - 품질검사 요청 목록</p>	
<p>품질검사 요청 목록을 조회할 수 있다.</p>	

TEMONEMES

HOME

1.수/발주관리

2.자재관리

3.생산관리

4.용질관리

5.출하관리

6.기초정보관리

0배도일시21-07-19 16:10:00

관리자CEO 임원 이완빛님이 로그인 하셨습니다.

Logout

품질관리

기준표 [민미]

검사요청 [다미]

품질 검사 요청 [다미]

요청목록 [다미]

검사측정값 [다미&보림]

검사현황 [다미&보림]

불량품 [한빛]

>품질 검사 요청 조회

품질검사코드

품목별 의외 코드

요청일

샘플링구분

>품질검사 요청 목록

<input type="checkbox"/>	번호	품질검사 요청 코드	제계번호	제품품명의의외코드	품질검사 코드	검사항명	품질등급 여부	품질인여부	품질일자시	품질판	품질정
<input type="checkbox"/>	1	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인완료	2021-06-18 18:15:11	NA전함	수정
<input type="checkbox"/>	2	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인완료	2021-06-18 18:15:11	NA전함	수정
<input type="checkbox"/>	3	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인완료	2021-06-18 18:15:11	NA전함	수정
<input type="checkbox"/>	4	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인완료	2021-06-18 18:15:11	NA전함	수정
<input type="checkbox"/>	5	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인완료	2021-06-18 18:15:11	NA전함	수정
<input type="checkbox"/>	6	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인완료	2021-06-18 18:15:11	NA전함	수정
<input type="checkbox"/>	7	QC_REQUEST01	1234	req01_MS_pants	A Ae1	원단 이화학검사(과괴검사)	샘플링	승인요청중	2021-06-18 18:15:11	NA전함	수정

[등록](#)
[삭제](#)

6.3.8 품질관리 - 품질검사 측정값 등록

품질관리 - 품질검사 측정값 등록, 요청코드 검색

품질검사 요청 코드 검색버튼을 누르면 **ajax** 호출을 통해 품질검사가 끝나지 않은 품질검사 요청 목록들이 뜬다. 품질검사 최종 **3** 회차까지 완료된 요청코드는 뜨지 않는다. 계약번호, 품목별 의뢰 코드, 요청날짜, 품질검사 명, 원부자재명, 거래처 명으로 검색 할 수 있다.

TEAMONEMES [HOME](#) [1.수/발주관리](#) [2.자재관리](#) [3.생산관리](#) [4.품질관리](#) [5.물류관리](#) [6.기초정보관리](#) [배달일시: 21-07-19 16:10:00](#) 관리자CEO 임원 이원철님이 로그인 하셨습니다. [Logout](#)

품질관리

- 기초표 (만여)
- 검사요청 (다미)
- 검사측정값 (다미&보합)**
- 품질검사 측정값 등록 (다미)
- 검사 측정값 목록 (다미)
- 품질검사결과등록 (보합)
- 생략서 조회/목록 (보합)
- 검사현황 (다미&보합)
- 물량통 (만여)

품질검사 측정값 등록(1.검색 2.품질검사요청 목록에서 가져오기 3.측정값 입력 4.등록 5.합/불 자동판정)

품질검사 요청코드 검색

<input type="checkbox"/> *품질검사 요청코드	검사명	품질검사코드	원부자재명	원부자재코드	측정값	시작일시	완료일시	등록	삭제
<input type="checkbox"/> 품질검사요청코드	검사명	품질검사코드	원부자재명	원부자재코드	측정값	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>

TEAMONEMES [HOME](#) [1.수/발주관리](#) [2.자재관리](#) [3.생산관리](#) [4.품질관리](#) [5.물류관리](#) [6.기초정보관리](#) [배달일시: 21-07-19 16:10:00](#) 관리자CEO 임원 이원철님이 로그인 하셨습니다. [Logout](#)

품질관리

- 기초표 (만여)
- 검사요청 (다미)
- 검사측정값 (다미&보합)**
- 품질검사 측정값 등록 (다미)
- 검사 측정값 목록 (다미)
- 품질검사결과등록 (보합)
- 생략서 조회/목록 (보합)
- 검사현황 (다미&보합)
- 물량통 (만여)

품질검사요청코드 목록

계약번호 품목별 의뢰 코드 요청날짜 연도-월-일 ~ 연도-월-일

품질검사 명 원부자재명 거래처명

<input type="checkbox"/>	no.	계약번호	품목별 의뢰 코드	품질검사대분류	품질검사중분류	품질검사소분류	거래처명	원부자재명	요청날짜
<input type="checkbox"/>	1	buyerContract002	rep00003_WS_blosure	공정중간검사	재단물검사	너지위치	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	2	buyerContract002	rep00003_WS_blosure	공정중간검사	재단물검사	너지위치	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	3	buyerContract002	rep00003_WS_blosure	공정중간검사	재단물검사	너지위치	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	4	buyerContract002	rep00003_WS_blosure	공정중간검사	재단물검사	너지위치	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	5	buyerContract002	rep00003_WS_blosure	공정중간검사	원단검사	오염	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	6	buyerContract002	rep00003_WS_blosure	공정중간검사	원단검사	눈에알피는부분들이있는경우	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	7	buyerContract002	rep00003_WS_blosure	공정중간검사	봉제검사	바늘질상태	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	8	buyerContract002	rep00003_WS_blosure	공정중간검사	봉제검사	시합이 맞물리경우	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	9	buyerContract002	rep00003_WS_blosure	공정중간검사	봉제검사	고지선발음확인	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	10	buyerContract002	rep00003_WS_blosure	공정중간검사	봉제검사	합착심지치수오차	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	11	buyerContract002	rep00003_WS_blosure	공정중간검사	봉제검사	어깨끝핀-소매 길이	살토로정장	해당없음	2018-02-28 14:13:14
<input type="checkbox"/>	12	buyerContract002	rep00003_WS_blosure	공정중간검사	봉제검사	밑단 좌우길이비교	살토로정장	해당없음	2018-02-28 14:13:14

품질검사 측정값 등록의 종류에는 합/불, 수치별, 등급별이 있는데 그에 맞춰서 측정값 품이 생성 된다. 측정값을 입력하면 처리과정에서 기준값과 비교해 등급을 측정하고 그 등급에 따라 합/불 판정이 된다.

TEAMONEMES

HOME

1.수/발주관리

2.사재관리

3.생산관리

4.품질관리

5.출력관리

6.기초정보관리

메포일시(21-07-19 16:10:00)

관리자CEO 방문 이원빛님이 로그인 하셨습니다.

Logout

품질관리

기공표 [한미]

검사요청 [다미]

검사속성값 [다미&보통]

품질검사 측정값 등록 [다미]

검사 측정값 목록 [다미]

최종검사결과등록 [보통]

상역서 조회/목록 [보통]

검사현황 [다미&보통]

불량품 [한빛]

품질검사 측정값 등록(1.검사 2.품질검사요청 목록에서 가져오기 3.측정값 입력 4.등록 5.합/불 자동판정)

품질검사 요청코드 검색

검색

1

직접입력 행 추가

*품질검사 요청코드	검사명	품질검사코드	원부자재명	원부자재코드	측정값	시작일시	완료일시	등록	삭제
<input type="checkbox"/> QC_REQUEST_0023	넘버링중복및누락	A028004C013D027	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0020	너치위치	A028004C012D024	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0025	오염	A028005C015D029	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0021	너치깊이	A028004C012D025	해당없음	해당없음	측정값	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0026	눈에알피는부분흠이있?	A028005C016D030	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0024	넘버링위치시접예물린?	A028004C014D028	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0027	바늘질상태	A028006C017D031	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0030	접착심지치수오차	A028006C020D035	해당없음	해당없음	측정값	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>
<input type="checkbox"/> QC_REQUEST_0029	고지선반출확인	A028006C019D034	해당없음	해당없음	<input type="checkbox"/> 합 <input type="checkbox"/> 불	yyyy-MM-dd hh:mm:ss	yyyy-MM-dd hh:mm:ss	<input type="button" value="등록"/>	<input type="button" value="삭제"/>

■JavaScript

```

$(function){
    //모달 실행
    $(document).on('click', '#requestCode, #requestCode2', function(){
        $('#requestCodeModal').modal("show");

        //검색조건, 체크박스 초기화
        $('#requestCodeModal').find('input[type=text]').val("");
        $("input[type=checkbox]:checked").attr('checked',false);
        //검색버튼 클릭
        $('#serchBtnModal').click(function(){
            $('.removeTr').remove();
            //검색값 넣기
            var searchValue = {
                contractCode : $('#contractCode').val()
                , qualityInspectionRequestCode:
                $('#qualityInspectionRequestCode').val()
                , inspectionRequestStartDate:
                $('#inspectionRequestStartDate').val()
                , inspectionRequestEndDate:
                $('#inspectionRequestEndDate').val()
                , subClassName :$('#subClassName').val()
                , rawMaterialName :$('#rawMaterialName').val()
                , clientName :$('#clientName').val()
            }
        })
    })
}

```

```

    };

    //param 데이터 직렬화
    var jsonData = JSON.stringify(searchValue);

    var html = "";
    var request = $.ajax({
        url: "/quality/searchQualityInspectionRequest",
        method: "post",
        type: "post",
        data: {"jsonData" : jsonData},
        dataType: "json"
    });

    request.done(function( data ) {
        console.log(data);
        if(data.length > 0){
            for(i=0; i<data.length; i++){
                var date = data[i].inspection_request_date;

                //date 포맷 변경 yyyy-MM-dd hh:mm:ss
                function formatDate(date) {
                    var d = new Date(date),
                        month = " " + (d.getMonth() + 1),
                        day = " " + d.getDate(),
                        year = d.getFullYear(),
                        hour = d.getHours(),
                        min = d.getMinutes(),
                        sec = d.getSeconds();

                    if (month.length < 2) month = '0' +
month;

                    if (day.length < 2) day = '0' + day;

                    return [year, month, day].join('-') + ' ' +
[hour, min, sec].join(':');

                }

                var dateText = "/Date(1519794794410)"/;
                var myDate = new Date(dateText.match(/<\/> *
1);

                date = formatDate(myDate);

                if(data[i].raw_material_name == undefined){
                    data[i].raw_material_name =
'해당없음';

                }

                html += '<tr class="removeTr">';
                html += '<td><input type="checkbox"
class="checkedModal" value="';

```

```
data[i].quality_inspection_request_code;
```

```
value="";
```

```
'</td>';
```

```
class="quality_inspection_code" value="";
```

```
'</td>';
```

```
class="raw_material_code" value="";
```

```
'</td>';
```

```
html +=
```

```
html += "></td>";
```

```
html += '<td>' + [i + 1] + '</td>';
```

```
//계약번호
```

```
html += '<td class="contract_code" value="';
```

```
html += data[i].contract_code
```

```
html += ">" + data[i].contract_code + '</td>';
```

```
//품목별의뢰번호
```

```
html += '<td class="requested_product_code"
```

```
html += data[i].requested_product_code;
```

```
html += ">" + data[i].requested_product_code +
```

```
//품질검사대분류명
```

```
html += '<td class="high_class_name" value="';
```

```
html += data[i].high_class_name;
```

```
html += ">" + data[i].high_class_name + '</td>';
```

```
//품질검사중분류명
```

```
html += '<td class="med_class_name" value="';
```

```
html += data[i].med_class_name;
```

```
html += ">" + data[i].med_class_name + '</td>';
```

```
//품질검사명
```

```
html += '<td class="sub_class_name" value="';
```

```
html += data[i].sub_class_name;
```

```
html += ">" + data[i].sub_class_name + '</td>';
```

```
//품질검사 코드
```

```
html += '<td style="display:none"
```

```
html += data[i].quality_inspection_code;
```

```
html += ">" + data[i].quality_inspection_code +
```

```
//원부자재 코드
```

```
html += '<td style="display:none"
```

```
html += data[i].raw_material_code;
```

```
html += ">" + data[i].raw_material_code +
```

```
//거래처명
```

```
html += '<td class="client_name" value="';
```

```
html += data[i].client_name;
```

```
html += ">" + data[i].client_name + '</td>';
```

```

value="";

//원부자재명
html += '<td class="raw_material_name"

html += data[i].raw_material_name;
html += ">' + data[i].raw_material_name +

'</td>';

html += '<td>' + date + '</td>';
html += '</tr>';

    }
    }else{
        $('#removeTr').remove();
        html += '<tr class="removeTr"><td colspan="15"
style="text-align: center;"> 검색된 결과가 없습니다. </td></tr>';
    }

    $('#mainTbody').append(html);

});

request.fail(function( jqXHR, textStatus ) {
    alert( "Request failed: " + textStatus );
});

});

});

//모달 save 눌렀을때
$('#saveBtn , #saveBtn2').click(function(){
    $('#tableTr').remove();

    //체크된 것 값 가져오기
    $('.checkedModal:checked').each(function(index,item){
        var requestCode = $(this).val();
        var qualityInspectionCode =
$(this).parent().parent().find('td.quality_inspection_code').text();
        var contractCode =
$(this).parent().parent().find('td.contract_code').text();
        var subClassName =
$(this).parent().parent().find('td.sub_class_name').text();
        var rawMaterialName =
$(this).parent().parent().find('td.raw_material_name').text();
        var rawMaterialCode =
$(this).parent().parent().find('td.raw_material_code').text();
        var insReqCode = $('.insReqCode');
        var inspectionName = $('.inspectionName');
        var materialsName = $('.materialsName');
        var cate = "";
        var cateRadio = "";
        var valueT = "";

```

```

var valueF = "";

//품질검사 카테고리 가져오기
var request = $.ajax({
    url: "/quality/getQualityInspectionCategory",
    method: "POST",
    data: { qualityInspectionCode :
qualityInspectionCode },

    dataType: "text"
});

request.done(function( data ) {
    console.log(data);

    if(data != " " && data != undefined && data !=
null){

        //카테고리에 따른 측정값 품 변경
        if(data == '합격/불합격'){
            cate = '<input type="checkbox"'
name="inspectionPassCheck" value="합격">합<input type="checkbox" name="inspectionPassCheck"
value="불합격">불';

            rawMaterialName = '해당없음';
            rawMaterialCode = '해당없음';

        }else if(data == '수치'){
            cate = '<input type="number"'
name="inspectionMeasurementValue" class="form-control" placeholder="측정값">';
            rawMaterialName = '해당없음';
            rawMaterialCode = '해당없음';

        }else if(data == '등급')
            cate = '<input type="number"'
name="inspectionMeasurementValue" class="form-control" placeholder="측정값">';

        //체크된 수만큼 측정값 등록 행 추가
        if(index >= 0 && index !=null &&
index != undefined){

            var tbody = $('#tbody');

            for(var i = 0; i < index+2; i++){
                var innerHtml = '<tr>';
                innerHtml +=
'<td><input type="checkbox" class="checked"></td>';

                //품질검사요청코드
                innerHtml += '<td><div
class="input-group"><input type="text" name="qualityInspectionRequestCode" class="form-control"
value="';

```

```

requestCode;
placeholder="품질검사요청코드 검색"></div></td>';

'<td><input type="text" class="form-control" value="" + subClassName + " readonly="readonly"></td>';
//품질검사코드
innerHtml +=
'<td><input type="text" name="qualityInspectionCode" class="form-control" value="" +
qualityInspectionCode + "></td>';

//원부자재명
innerHtml +=
'<td><input type="text" class="form-control" value="" + rawMaterialName + "
readonly="readonly"></td>';

//원부자재코드
innerHtml +=
'<td><input type="text" name="rawMaterialCode" class="form-control" value="" + rawMaterialCode + ">';
//측정값
innerHtml += '<td>';
innerHtml += cate;
innerHtml += '</td>';
//측정시간
innerHtml +=
'<td><input type="text" name="inspectionStartDate" class="form-control" placeholder="yyyy-MM-dd
hh:mm:ss"></td>';

innerHtml +=
'<td><input type="text" name="inspectionEndDate" class="form-control" placeholder="yyyy-MM-dd
hh:mm:ss"></td>';

innerHtml +=
'<td><button type="button" class="addBtn">등록</button></td>';

innerHtml +=
'<td><button type="button" class="removeButton">삭제</button></td>';

innerHtml += '</tr>';

tbody.append(innerHtml);

break;
}
}
});

request.fail(function( jqXHR, textStatus ) {
    alert( "Request failed: " + textStatus );
});

});
//모달 닫기
$('#requestCodeModal').modal("hide");
});

```


//모달 전체 체크

```
$(document).on('click', '#checkAllModal', function(){
    if($('#checkAllModal').prop('checked')){
        $('#checkedModal').prop('checked',true);
    }else{
        $('#checkedModal').prop('checked',false);
    }
});
```

//메인 전체 체크

```
$(document).on('click', '#checkAll', function(){
    if($('#checkAll').prop('checked')){
        $('#checked').prop('checked',true);
    }else{
        $('#checked').prop('checked',false);
    }
});
```

//다중 등록 ajax

```
var addInsValueAjax = function(array){
    $.ajax({
        url      : '/quality/addInspectionMeasurementValue',
        type     : 'POST',
        data     : JSON.stringify(array),
        contentType : 'application/json; charset=UTF-8',
        dataType: 'text',
        success : function(data) {
            console.log(data);

            //리로드
            if(data) location.reload(true);
            else{
                alert('오류가 발생했습니다. 관리자에게 문의해주세요.');
```

}

},

error : function(xhr,status,error) {

console.log('오류가 발생했습니다. 관리자에게 문의해주세요.');

}

});

}

//다중 등록

```
$(document).on('click', '#addInsValue', function(){

    var validationCheck = false;

    var addIns = $('#tbody tr');
    var insArray = [];

    for(var i = 0; i < addIns.length; i++){
        var insInfo = {};
```

```

        var bindElement = addIns.eq(i).find('input[type=text],
input[type=number], input[type=checkbox]:checked');

        $.each(bindElement, function(){
            var insKey = $(this).attr('name');
            var insValue = $(this).val();

            if(insValue == undefined || insValue == "" || insValue ==
null){

                alert('값을 입력하여주세요. ');
                $(this).focus();
                validationCheck = false;
                return false;
            }

            if(insKey != null && insKey != undefined && insKey != ""
&& insValue != '해당없음'){

                insInfo[insKey] = insValue;
                validationCheck = true;
            }
            insArray.push(insInfo);
        });
        console.log(JSON.stringify(insArray));
        if(validationCheck) addInsValueAjax(insArray);
    });

    // 체크된 행 삭제
    $(document).on('click', '#deleteBtn', function(){
        if (confirm("정말 삭제하시겠습니까?")) {
            // 확인 버튼 클릭 시 동작
            var checkObj = $('input[class="checked"]:checked');
            // 체크된 항목 수
            var checkCount = checkObj.length;
            // 전체 행의 수
            var bodyCount = $('#tbody tr').length;

            if(checkCount < 1){
                alert('체크된 항목이 없습니다. ');
                return false;
            }

            if(checkCount >= bodyCount) {
                alert('모든 행을 지울 수 없습니다. ');
                return false;
            }

            $.each(checkObj, function(){
                $(this).parent().parent().remove();
                $('#checkAll').prop('checked', false);
            });
        }
    });

```

```

        var lastTrRemoveBtnObj = $('#tbody tr:last td').find('.removeButton');

    } else {
        // 취소 버튼 클릭 시 동작
        alert("동작을 취소했습니다.");
    }

    });
    //행 추가 기능
    $('#addRows').click(function(){
        var count = $('#trCount').val();
        var tbody = $('#tbody');

        if(count != null && count != undefined && count > 0){

            for(var i = 0; i < count; i++){
                var lastIndex = $(tbody).find('tr:last').length;
                var innerHtml = '<tr>';
                innerHtml += '<td><input type="checkbox"
class="checked"></td>';

                //품질검사 요청 코드
                innerHtml += '<td><input type="text"
name="qualityInspectionRequestCode" class="form-control" placeholder="품질검사요청코드
직접입력"><div class="input-group-btn"></td>';

                //검사명
                innerHtml += '<td><input type="text" class="form-control"
placeholder="검사명 직접입력"></td>';

                //품질검사 코드
                innerHtml += '<td><input type="text"
name="qualityInspectionCode" class="form-control" </td>';

                //원부자재명
                innerHtml += '<td><input type="text" class="form-control"
placeholder="원부자재명 직접입력"></td>';

                //원부자재코드
                innerHtml += '<td><input type="text" name="rawMaterialCode"
class="form-control" placeholder="원부자재명 직접입력"></td>';

                //측정값
                innerHtml += '<td><input type="number"
name="inspectionMeasurementValue" class="form-control" placeholder="측정값"></td>';

                //측정시작시간
                innerHtml += '<td><input type="text"
name="inspectionStartDate" class="form-control" placeholder="yyyy-MM-dd hh:mm:ss"></td>';

                //측정종료시간
                innerHtml += '<td><input type="text"
name="inspectionEndDate" class="form-control" placeholder="yyyy-MM-dd hh:mm:ss"></td>';

                //등록버튼
                innerHtml += '<td><button type="button"
id="addInsValueSBtn">등록</button></td>';
            }
        }
    });

```

```

        if(i == (count - 1)){
            innerHtml += '<td><button type="button"
class="removeButton">삭제 </button></td>';
        }else{
            innerHtml += '<td><button type="button"
class="removeButton">삭제 </button></td>';
        }

        innerHtml += '</tr>';
        tbody.append(innerHtml);
    }
}
}else{
    alert('확인');
    $('#trCount').focus();
    return false;
}
$('#trCount').val(1);
});

//행 삭제
$(document).on('click', '.removeButton', function(){
    var bodyCount = $('#tbody tr').length;
    var removeBtn = $('.removeButton').parent().parent();

    if(bodyCount > removeBtn.length ) {
        alert('모든 행을 지울 수 없습니다. ');
        return false;
    }

    else if (confirm("정말 삭제하시겠습니까?")) {
        // 확인 버튼 클릭 시 동작
        var tr = $(this).parent().parent();
        tr.remove();
    } else {
        // 취소 버튼 클릭 시 동작
        alert("동작을 취소했습니다.");
    }

    $('#addMemberBody tr:last td:last button').attr('class', 'addButton');
    $('#addMemberBody tr:last td:last button').text('삭제');

});
});

```

■Controller

@Controller

@RequestMapping("/quality")

public class QualityControlController{

private static final Logger log = LoggerFactory.getLogger(QualityControlController_KDM.class);

@Autowired

private QualityInsMeasurementValueService qualityInsMeasurementValueService;

@Autowired

private QualityInsepctionStauteService qualityInsepctionStauteService;

@Autowired

public QualityControlController_KDM(QualityInsMeasurementValueService
qualityInsMeasurementValueService,QualityInsepctionStauteService qualityInsepctionStauteService) {
this.qualityInsepctionStauteService = qualityInsepctionStauteService;
this.qualityInsMeasurementValueService =qualityInsMeasurementValueService;
}

//[다미] 품질검사 측정값 등록

@PostMapping("/addInspectionMeasurementValue")

@ResponseBody

public boolean addInspectionMeasurementValue(@RequestBody List<QualityInspectionResult>
qualityInspectionResult) {

int value =

qualityInsMeasurementValueService.addQualityInspectionResult(qualityInspectionResult);

boolean result = false;

//insert 완료

if(value > 0) {

result = true;

//insert 실패

}else {

result = false;

}

return result;

}

//[다미] 품질검사 측정값 카테고리에 맞는 행 만들기

@PostMapping("/getQualityInspectionCategory")

@ResponseBody

public String getQualityInspectionCategory(@RequestParam(value =
"qualityInspectionCode")String qualityInspectionCode){

List<QualityInspectionStandard> cate =

qualityInsMeasurementValueService.getQualityInspectionStandard(qualityInspectionCode);

String result = cate.get(0).getCategory();

```

        return result;
    }

    //[다미] 품질검사 측정값 등록 화면
    @GetMapping("/addInspectionMeasurementValue")
    public String addInspectionMeasurementValue() {
        return "quality/addInspectionMeasurementValue";
    }

    //[다미]품질검사 요청 목록 모달
    @RequestMapping(value = "searchQualityInspectionRequest", method =
RequestMethod.POST)
    @ResponseBody
    public List<Map<String, Object>> searchQualityInspectionRequest(@RequestParam(value =
"jsonData", required = false)String jsonData) {

        log.info("1. 처음 들어온 데이터(json 문자열): jsonData: {} ", jsonData);

        JSONParser parser = new JSONParser();
        Object obj = null;
        try {
            // parsing 후 Object 객체에 담음
            obj = parser.parse(jsonData);
            log.info("2.파싱한 데이터: obj: {} ", obj);
        } catch (ParseException e) {
            e.printStackTrace();
        }

        //3. JsonObject 로 변환
        JsonObject jsonObj = (JsonObject) obj;

        Map<String, Object> map = new HashMap<String, Object>();
        List<Map<String, Object>> map2 =
qualityInsMeasurementValueService.searchQualityInspectionRequest(jsonObj);
        log.info("searchQualityInspectionRequest: {} ", map2);

        return map2;
    }

    //[다미]품질검사이청목록
    @GetMapping("/qualityInspectionRequestList")
    public String qualityControlRequestList() {
        return "quality/qualityInspectionRequestList";
    }

    //[다미]품질검사이청 메서드
    @PostMapping("/qualityInspectionRequest")
    public String qualityInspectionRequest(@RequestParam(name = "qualityInspection", required =
false)String qualityInspection) {
        return "redirect:/qualityInspectionRequestList";
    }

```

```

//[다미]품질검사요청
@GetMapping("/qualityInspectionRequest")
public String qualityControlRequest() {
    return "quality/qualityInspectionRequest";
}
}

```

■Service

```

//품질검사 측정값 등록
public int addQualityInspectionResult(List<QualityInspectionResult> qualityInspectionResult) {
    int result = 0;

    log.info("##처음 들어오는 배열## : {}", qualityInspectionResult);

    for(int i = 0; i < qualityInspectionResult.size(); i++) {
        String category = null; //품질검사 카테고리
        String passValue = null; //최종 합/불
        int measuredLevel = 0; //등급값
        int passCheckValue = 0; //판정 기준 최소값
        int resultValue = 0; //측정값
        int min = 0; //최소값
        int max = 0; //최대값

        //품질검사 코드
        String qualityInspectionCode =
qualityInspectionResult.get(i).getQualityInspectionCode();
        String qualityInspectionRequestCode =
qualityInspectionResult.get(i).getQualityInspectionRequestCode();

        //01. 등급/수치/pass 인지
        List<QualityInspectionStandard> qi =
qualityInsMeasurementValueMapper.getQualityInspectionStandard(qualityInspectionCode);

        log.info("1. 품질검사코드로 조회하는 기준치 : {}", qi);

        //01-1. 품질검사 카테고리 가져오기
        category = qi.get(0).getCategory();
        log.info("1-1. 카테고리 : {}", category);

        //=====
        //02. 합.불 카테고리일때
        //=====
        if(category.equals("합격/불합격")) {

```

```

//02-1. 바로 최종 합/불
passValue = qualityInspectionResult.get(i).getInspectionPassCheck();

//=====
//03.수치별 카테고리일때
//=====
}else if(category.equals("수치")) {
    log.info("[수치]: 01. 처음 들어오는 배열 : {}",
qualityInspectionResult.get(i));

    //03-1. 측정값 변수 resultValue 에 담기
    resultValue =
qualityInspectionResult.get(i).getInspectionMeasurementValue();

    //03-2. 기준값 만들기(사이즈 측정값 +- 기준값(오차범위))
    List<Map<String, Object>> measureList
        =
qualityInsMeasurementValueMapper.getQualityInspectionStandardByNumber(qualityInspectionCode,
qualityInspectionRequestCode);
    log.info("[수치]: 02. 의류 부위별 측정값 가져오기 : {}", measureList);

    //03-3. 의류 품목별 측정값이 아닐시에(ex. 너치깊이) 바로 기준값과
비교하기
    if(!(measureList.size() > 0)) {
        //03-3-1. 품질검사 측정값 기준치 최소값, 최대값
        min = qi.get(0).getMinValue();
        max = qi.get(0).getMaxValue();
        log.info("[수치]: 03. 품질검사 측정값 기준치 최소값: {}", min);

        log.info("[수치]: 03-1. 품질검사 측정값 기준치 최대값: {}",
max);
    }else {
        //03-4. 의류 부위별 품질검사의 기준이 되는 측정값
        int measuredValue = (int)
measureList.get(0).get("measured_value");
        //03-4-1. 기준이 되는 측정값의 허용오차를 반영한 최소값,
최대값

        min = qi.get(0).getMinValue() + measuredValue;
        max = qi.get(0).getMaxValue() + measuredValue;
        log.info("[수치]: 03. 최소 허용오차 : {}", min);

        log.info("[수치]: 03-1. 최대 허용 오차 : {}", max);
    }

    //03-5 최소값, 최대값과 비교
    if(max >= resultValue && resultValue >= min) {

```



```

        passValue = "합격";
    }else {
        passValue = "불합격";
    }

//=====
//04.등급별 카테고리일 때
//=====
}else if(category.equals("등급")) {

    //04-1.원부자재 기준치와 값 비교
    //04-2. 측정값 변수 resultValue 에 담기
    log.info("[등급]: 01. 처음 들어오는 배열 : {}",
qualityInspectionResult.get(i));

    resultValue =
qualityInspectionResult.get(i).getInspectionMeasurementValue();
    log.info("[등급]: 01-2. 측정값 : {}", resultValue);

    //04-3.해당 원부자재의 기준치 가져오기
    List<QualityBiochemFabricLevelStandard> bioStandard
        =
qualityInsMeasurementValueMapper.getBiochemFabricLevelStandard(qualityInspectionResult.get(i));
    log.info("[등급]: 02. 원부자재 기준치 가져오기 : {}",
bioStandard);

    //04-4.측정값과 기준치 비교해서 등급 판정
    for(int j = 0; j < bioStandard.size(); j++) {

        if(bioStandard.get(j).getMinValue() <= resultValue &&
bioStandard.get(j).getMaxValue() >= resultValue) {
            measuredLevel =
bioStandard.get(j).getMeasuredLevel();
            log.info("[등급]: 03. 원부자재 기준치와 비교해
등급 측정 : {}", measuredLevel);
        }

    }

    //04-5.합/불 판정
    //04-6 판정 최소 값 가져오기
    List<QualityInspectionStandard> insStandard
        =
qualityInsMeasurementValueMapper.getQualityInspectionStandard(qualityInspectionResult.get(i).getQualityInspectionCode());

    passCheckValue = insStandard.get(0).getMinValue();
    log.info("[등급]: 04. 최소 등급 가져오기 : {}",
passCheckValue);

```

```

//04-7 최소 등급과 현재 등급 비교해서 합/불 판정하기
if(measuredLevel >= passCheckValue) {
    passValue = "합격";
}else if(measuredLevel <= passCheckValue) {
    passValue = "불합격";
}
log.info("[등급]: 05. 최종 합/불 : {}", passValue);
}else {
    result = 0;
}

//05. 해당 DTO 에 담기

qualityInspectionResult.get(i).setInspectionMeasurementValue(resultValue);//측정 값
qualityInspectionResult.get(i).setInspectionPassCheck(passValue);//최종 합/불

qualityInspectionResult.get(i).setInspectionMeasurementLevelResult(measuredLevel);//등급 값
qualityInspectionResult.get(i).setMinTolerance(passCheckValue);//판정 기준
최소 등급 값

log.info("06. 최종 결과값 set : {}", qualityInspectionResult.get(i));

//06. Insert 실행

//qualityInsMeasurementValueMapper.addQualityRawMaterialInspectionResult(qualityInspection
Result.get(i));

    result = 1;
}
for(int i=0; i < qualityInspectionResult.size(); i++) {
    String inspectionCode =
qualityInspectionResult.get(i).getQualityInspectionCode();
    Map<String,Object> maxNumMap =
finalResultMapper.getMaxMeasurementNum(inspectionCode);
    String maxNum = (String) maxNumMap.get("maxNum");
    if(maxNum.equals("3 회차")) {

    }

}

/*
* 배열.i.inspectionCode 찾아 mapper 에서 max measurementNum =='3 회차'
최종결과값 넣는 mapper 에서
* 가져오는 쿼리 실행
*/

return result;
}

```

```

//품질검사 카테고리 가져오기
public List<QualityInspectionStandard> getQualityInspectionStandard(String
qualityInspectionCode) {
    return
qualityInsMeasurementValueMapper.getQualityInspectionStandard(qualityInspectionCode);
}

//품질검사 요청 검색 모달
public List<Map<String, Object>> searchQualityInspectionRequest(Map<String,Object> map){

    return qualityInsMeasurementValueMapper.searchQualityInspectionRequest(map);
};
}

```

■SQL Mapper

```

<mapper namespace="ksmart39.springboot.dao.QualityInsMeasurementValueMapper">

    <resultMap type="QualityBiochemFabricLevelStandard"
id="QualityBiochemFabricLevelStandardResultMap">
        <result property="rawMaterialCode"
column="raw_material_code" />
        <result property="qualityInspectionCode"
column="quality_inspection_code" />
        <result property="minValue"
column="min_value" />
        <result property="maxValue"
column="max_value" />
        <result property="measuredLevel"
column="measured_level" />
        <result property="biochemInspectionStandardRegDate"
column="biochem_inspection_standard_reg_date" />
        <result property="biochemInspectionStandardUpdateDate"
column="biochem_inspection_standard_update_date" />
    </resultMap>

    <resultMap type="QualityInspectionStandard"
id="QualityInspectionStandardResultMap">
        <result property="qualityInspectionCode"
column="quality_inspection_code"/>
        <result property="category"
column="category"/>
        <result property="minValue"
column="min_value"/>
        <result property="maxValue"
column="max_value"/>

```

```
        <result property="method"
column="method"/>
        <result property="standardMeasurementUnit"
column="standard_measurement_unit"/>
        <result property="standardTolerance"
column="standard_tolerance"/>
        <result property="requiredInspectionCheck"
column="required_inspection_check"/>
        <result property="inspectionStandardRegDate"
column="inspection_standard_reg_date"/>
        <result property="inspectionStandardUpdateDate"
column="inspection_standard_update_date"/>
</resultMap>
```

```
<resultMap type="QualityInspectionResult"
id="QualityInspectionResultMap">
    <result property="inspectionResultCode"
column="inspection_result_code"/>
    <result property="qualityInspectionRequestCode"
column="quality_inspection_request_code"/>
    <result property="chargeEmployeeCode"
column="charge_employee_code"/>
    <result property="qualityInspectionCode"
column="quality_inspection_code"/>
    <result property="inspectionMeasurementNum"
column="inspection_measurement_num"/>
    <result property="inspectionMeasurementValue"
column="inspection_measurement_value"/>
    <result property="inspectionMeasurementLevelResult"
column="inspection_measurement_level_result"/>
    <result property="minTolerance"
column="min_tolerance"/>
    <result property="maxTolerance"
column="max_tolerance"/>
    <result property="errorRange"
column="error_range"/>
    <result property="inspectionPassCheck"
column="inspection_pass_check"/>
    <result property="inspectionStartDate"
column="inspection_start_date"/>
    <result property="inspectionDuration"
column="inspection_duration"/>
    <result property="inspectionEndDate"
column="inspection_end_date"/>
    <result property="inspectionResultRegDate"
column="inspection_result_reg_date"/>
    <result property="inspectionResultUpdateDate"
column="inspection_result_update_date"/>
    <result property="rawMaterialCode"
column="raw_material_code"/>
</resultMap>
```

<!-- 품질검사 측정값 등록 -->

<insert id="addQualityRawMaterialInspectionResult" parameterType="QualityInspectionResult">

INSERT INTO inspection_result

(

inspection_result_code

,quality_inspection_request_code

,charge_employee_code

,quality_inspection_code

,raw_material_code

,inspection_measurement_num

,inspection_measurement_value

,inspection_measurement_level_result

,min_tolerance

,max_tolerance

,error_range

,inspection_pass_check

,inspection_start_date

,inspection_duration

,inspection_end_date

,inspection_result_reg_date

,inspection_result_update_date

)VALUES(

sf_qualityInspectionResultCode(#{qualityInspectionRequestCode})

,#{qualityInspectionRequestCode}

,#{chargeEmployeeCode}

,#{qualityInspectionCode}

,#{rawMaterialCode}

,sf_inspectionMeasurementNum(#{qualityInspectionRequestCode})

,#{inspectionMeasurementValue}

,#{inspectionMeasurementLevelResult}

,#{minTolerance}

,#{maxTolerance}

,#{errorRange}

,#{inspectionPassCheck}

,#{inspectionStartDate}

,sf_minuteDuration(#{inspectionStartDate},#{inspectionEndDate})

,#{inspectionEndDate}

,NOW()

,NULL

);

</insert>

<!-- 품질검사 수치별 등록 기준값 가져오기 -->

<select id="getQualityInspectionStandardByNumber" parameterType="String"

resultType="Map">

SELECT

s.quality_inspection_code

,s.measurement_part

,ps.measured_value

FROM

required_size_list AS s

```

        INNER JOIN
            requested_product_size AS ps ON s.required_size_code =
ps.required_size_code
        INNER JOIN
            requested_product_detail AS p ON ps.requested_product_code =
p.requested_product_code
        WHERE
            s.quality_inspection_code = #{qualityInspectionCode}
        AND
            ps.requested_product_code = (SELECT
                r.requested_product_code
            FROM
                requested_product_detail AS r
            INNER JOIN
                quality_inspection_request AS q ON r.requested_product_code =
q.requested_product_code
            WHERE
                q.quality_inspection_request_code = #{qualityInspectionRequestCode})
        GROUP BY s.quality_inspection_code;
</select>

```

<!-- 품질검사 측정값 기준치 -->

```

<select id="getQualityInspectionStandard" parameterType="String"
resultMap="QualityInspectionStandardResultMap" >

```

```

    SELECT
        quality_inspection_code
        ,category
        ,min_value
        ,max_value
        ,method
        ,standard_measurement_unit
        ,standard_tolerance
        ,required_inspection_check
        ,inspection_standard_reg_date
        ,inspection_standard_update_date

```

```

    FROM
        inspection_standard AS i

```

```

    WHERE
        i.quality_inspection_code = #{qualityInspectionCode};

```

</select>

<!-- 품질검사 원부자재 기준치 목록 -->

```

<select id="getBiochemFabricLevelStandard" parameterType="QualityInspectionResult"
resultMap="QualityBiochemFabricLevelStandardResultMap">

```

```

    SELECT

```

```

        raw_material_code
        ,quality_inspection_code
        ,min_value
        ,max_value
        ,measured_level
        ,biochem_inspection_standard_reg_date
        ,biochem_inspection_standard_update_date
FROM
        biochem_fabric_level_standard AS b
WHERE
        b.quality_inspection_code = #{qualityInspectionCode}
        AND
        b.raw_material_code = #{rawMaterialCode};
</select>

<!-- 품질검사 요청목록 목록 -->
<select id="searchQualityInspectionRequest" resultType="Map" parameterType="Map">
    SELECT
        q.quality_inspection_request_code
        ,q.requested_product_code
        ,r.contract_code
        ,h.high_class_name
        ,md.med_class_name
        ,s.quality_inspection_code
        ,s.sub_class_name
        ,q.raw_material_code
        ,m.raw_material_name
        ,q.inspection_request_date
        ,c.client_name
    FROM
        quality_inspection_request AS q INNER JOIN requested_product_detail AS r
ON q.requested_product_code = r.requested_product_code
        INNER JOIN sub_class_inspection_list AS s ON q.quality_inspection_code =
s.quality_inspection_code
        INNER JOIN medium_class_inspection_list AS md ON s.high_med_class_code =
= md.high_med_class_code
        INNER JOIN high_class_inspection_list AS h ON md.high_class_code =
h.high_class_code
        LEFT JOIN raw_material_info AS m ON q.raw_material_code =
m.raw_material_code
        LEFT JOIN client_info AS c ON r.client_code = c.client_code
        LEFT JOIN inspection_result AS re ON q.quality_inspection_request_code =
re.quality_inspection_request_code
    <trim prefix="WHERE" prefixOverrides="AND">
        AND
        q.quality_inspection_request_code NOT IN
            ( select
                ins.quality_inspection_request_code
            from
                inspection_result AS ins
            WHERE
                ins.inspection_measurement_num='3 회차')
    </trim>
</select>

```

```

                <if test="contractCode != null and contractCode != ".toString()">
                    AND    r.contract_code LIKE CONCAT('%', #{contractCode} ,'%')
                </if>
                <if test="requestedProductCode != null and requestedProductCode !=
".toString()">
                    AND q.requested_product_code LIKE CONCAT('%',
#{requestedProductCode} ,'%')
                </if>
                <if test="inspectionRequestStartDate != null and inspectionRequestStartDate !=
".toString()">
                    AND q.inspection_request_date >=
DATE(#{inspectionRequestStartDate})
                </if>
                <if test="inspectionRequestEndDate != null and inspectionRequestEndDate !=
".toString()">
                    AND    q.inspection_request_date <=
DATE(#{inspectionRequestEndDate})
                </if>
                <if test="subClassName != null and subClassName != ".toString()">
                    AND s.sub_class_name LIKE CONCAT('%', #{subClassName} ,'%')
                </if>
                <if test="rawMaterialName != null and rawMaterialName != ".toString()">
                    AND m.raw_material_name LIKE CONCAT('%',
#{rawMaterialName} ,'%')
                </if>
                <if test="clientName != null and clientName != ".toString()">
                    AND c.client_name LIKE CONCAT('%', #{clientName} ,'%')
                </if>
                GROUP BY q.quality_inspection_request_code;
            </trim>
        </select>
    </mapper>

```


6.3.9 품질관리 - 품질검사 측정값 목록

품질관리 - 품질검사 측정 목록

품질검사 측정 목록을 조회할 수 있다.

TEAMONEMES

HOME1.수/발주관리2.자재관리3.생산관리4.품질관리5.물류관리6.기초정보관리

테스트입력시21-07-19 16:10:00관리자CEO 임원 이명빛님이 로그인 하셨습니다.

Logout

품질관리

기존표 (관리)

검사요청 (관리)

검사측정값 (관리&보통)

품질검사 측정값 등록 (관리)

품질검사결과등록 (보통)

상적서 조회/목록 (보통)

검사현황 (관리&보통)

물류통 (관리)

>품질 검사 요청 조회

품질검사코드품질별 의뢰 코드요청일샘플링구분

직접 입력 또는 검색직접 입력 또는 검색연도-월-일연도-월-일All검색

>품질검사 요청 목록

선택	번호	품질검사 요청 코드	품질검사 코드	품질격여부	품질현황	품질정
<input type="checkbox"/>	1	QC_REQUEST_0001	A01B001C001D001	합격	내완료	수정
<input type="checkbox"/>	2	QC_REQUEST_0002	A01B001C001D002	합격	내완료	수정
<input type="checkbox"/>	3	QC_REQUEST_0003	A01B001C002D003	불합격	내완료	수정
<input type="checkbox"/>	4	QC_REQUEST_0004	A01B001C003D004	합격	내완료	수정
<input type="checkbox"/>	5	QC_REQUEST_0005	A01B001C004D005	불합격	내완료	수정
<input type="checkbox"/>	6	QC_REQUEST_0006	A01B001C004D006	불합격	내완료	수정
<input type="checkbox"/>	7	QC_REQUEST_0007	A01B001C005D007	합격	내완료	수정
<input type="checkbox"/>	8	QC_REQUEST_0008	A01B003C008D016	합격	내완료	수정
<input type="checkbox"/>	9	QC_REQUEST_0009	A01B003C008D017	합격	내완료	수정
<input type="checkbox"/>	10	QC_REQUEST_0010	A01B003C008D018	합격	내완료	수정
<input type="checkbox"/>	11	QC_REQUEST_0011	A01B003C008D016	합격	내완료	수정
<input type="checkbox"/>	12	QC_REQUEST_0012	A01B003C008D017	합격	내완료	수정
<input type="checkbox"/>	13	QC_REQUEST_0013	A01B003C008D018	합격	내완료	수정
<input type="checkbox"/>	14	QC_REQUEST_0014	A01B003C008D016	합격	내완료	수정
<input type="checkbox"/>	15	QC_REQUEST_0015	A01B003C008D017	합격	내완료	수정

6.3.10 품질관리 - 품질검사 실시간 현황

품질관리 - 품질검사 실시간 현황

품질검사 실시간 현황을 조회할 수 있는 화면이다.

계약번호를 필수로 기입을 해야하는데 검색을 통해 불러 올 수 있다.

또한 거래처명, 검사 요청일, 검사 실시일, 제품품목 별로 검색이 가능하다.

품질관리 - 현황 불러오기

조회 버튼을 누르면 현황을 볼 수 있다.

계약번호 전체의 품질검사 요청건수, 진행건수, 완료건수, 완료율을 볼 수 있고 한눈에 보기 편하게 차트로 구성되어 있다. 어떤 품질검사에서 합격이 되었는지 파악이 가능하다.

TEAMONEMES

HOME1수/발주관리2차재관리3생산관리4품질관리5공정관리6기초정보관리●테포일시21-07-19 16:10:00관리자CEO 방문 이원빛님이 로그인 하셨습니다Logout

품질관리

기본표 (민이)

검사요청 (다미)

검사숙경감 (다미&보람)

검사현황 (다미&보람)

실시간 검사현황 (다미)

계약별 검사현황 (다미)

품질검사별 합격률 조회 (민이)

이

불량품 (민이)

* 거래처명 or 수주계약 물수 기업 사항

계약번호 buyerContract002Q

거래처명 검색어를 입력하세요.Q

검사요청일 연도-월-일 ~ 연도-월-일

제작품목 All::선택::

품질검사코드 검색어를 입력하세요.Q

검사실시일 연도-월-일 ~ 연도-월-일

조회

> 요청 대비 검사 현황

수주계약번호 buyerContract002로 검색한 결과입니다.

요청42건

진행24건

완료15건

완료율36%

현황

최종 합격/불합격

검사별 불합격 비율

no.	공정	공정소분	공정세분	품목코드	품목명	요청시간	검사단위	숙정값			처리정보		최종합격여부	기타
								1회차	2회차	3회차	시작시간	완료시간		
1	원자재검사	물세탁	변회색	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-07 11:26:00	2018-06-08 12:29:00	합격	없음
2	원자재검사	물세탁	작물 외관변화	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급/불합격	합격	합격	합격	2018-06-11 16:40:00	2018-06-11 16:46:00	합격	없음
3	원자재검사	드라이클리닝_염색성	자체이염	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	불합격	불합격	불합격	2018-06-07 14:08:00	2018-06-07 14:22:00	불합격	없음
4	원자재검사	마찰변색도	견	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-07 15:10:00	2018-06-07 15:28:00	합격	없음
5	원자재검사	땀,알광복합견뢰도	방견뢰도	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	불합격	불합격	합격	2018-06-08 09:50:00	2018-06-08 10:25:00	불합격	없음
6	원자재검사	땀,알광복합견뢰도	알광복합견뢰도	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	불합격	불합격	불합격	2018-06-08 11:45:00	2018-06-08 14:15:00	불합격	없음
7	원자재검사	마찰견뢰도	석유계	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-08 13:44:00	2018-06-08 14:44:00	합격	없음
8	원자재검사	식지기능검사	상출현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-11 10:50:00	2018-06-11 14:47:00	합격	없음

TEAMONEMES

HOME1수/발주관리2차재관리3생산관리4품질관리5공정관리6기초정보관리●테포일시21-07-19 16:10:00관리자CEO 방문 이원빛님이 로그인 하셨습니다Logout

품질관리

기본표 (민이)

검사요청 (다미)

검사숙경감 (다미&보람)

검사현황 (다미&보람)

실시간 검사현황 (다미)

계약별 검사현황 (다미)

품질검사별 합격률 조회 (민이)

이

불량품 (민이)

no.

공정

공정소분

공정세분

품목코드

품목명

요청시간

검사단위

숙정값

처리정보

최종합격여부

기타

1	원자재검사	물세탁	변회색	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-07 11:26:00	2018-06-08 12:29:00	합격	없음
2	원자재검사	물세탁	작물 외관변화	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급/불합격	합격	합격	합격	2018-06-11 16:40:00	2018-06-11 16:46:00	합격	없음
3	원자재검사	드라이클리닝_염색성	자체이염	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	불합격	불합격	불합격	2018-06-07 14:08:00	2018-06-07 14:22:00	불합격	없음
4	원자재검사	마찰변색도	견	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-07 15:10:00	2018-06-07 15:28:00	합격	없음
5	원자재검사	땀,알광복합견뢰도	방견뢰도	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	불합격	불합격	합격	2018-06-08 09:50:00	2018-06-08 10:25:00	불합격	없음
6	원자재검사	땀,알광복합견뢰도	알광복합견뢰도	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	불합격	불합격	불합격	2018-06-08 11:45:00	2018-06-08 14:15:00	불합격	없음
7	원자재검사	마찰견뢰도	석유계	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-08 13:44:00	2018-06-08 14:44:00	합격	없음
8	원자재검사	심지기능검사	상출현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-11 10:50:00	2018-06-11 14:47:00	합격	없음
9	원자재검사	심지기능검사	물결모양현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-11 11:08:00	2018-06-11 16:10:00	합격	없음
10	원자재검사	심지기능검사	비불현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-11 13:03:00	2018-06-11 16:40:00	합격	없음
11	원자재검사	심지기능검사	상출현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-12 09:50:00	2018-06-12 10:30:00	합격	없음
12	원자재검사	심지기능검사	물결모양현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-12 10:40:00	2018-06-12 11:35:00	합격	없음
13	원자재검사	심지기능검사	비불현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-12 11:00:00	2018-06-12 12:10:00	합격	없음
14	원자재검사	심지기능검사	상출현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	합격	2018-06-12 13:07:00	2018-06-12 14:50:00	합격	없음
15	원자재검사	심지기능검사	물결모양현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	불합격	불합격	2018-06-12 13:17:00	2021-07-27 13:30:13	종료전	없음
16	원자재검사	심지기능검사	비불현상	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	등급	합격	합격	불합격	2018-06-12 13:29:00	2018-06-08 10:25:00	종료전	없음
17	원자재검사	외관검사	단추로고불량	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	합격	합격	불합격	2018-06-08 10:20:00	2018-06-08 11:16:00	종료전	없음
18	원자재검사	외관검사	단추재질불량	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	합격	합격	합격	2018-06-08 10:20:00	2018-06-08 10:35:00	종료전	없음
19	원자재검사	단추기능검사	케미진단추단검	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	합격	불합격	불합격	2018-06-08 10:20:00	2018-06-08 10:48:00	종료전	없음
20	공정중간검사	너치	너치위치	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	심사중	심사중	심사중	시작전	종료전	종료전	없음
21	공정중간검사	너치	너치길이	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	수치	심사중	심사중	심사중	시작전	종료전	종료전	없음
22	공정중간검사	너치	너치빠짐	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	합격	불합격	불합격	2021-07-27 12:13:13	2021-07-28 08:53:13	불합격	없음
23	공정중간검사	넙비형요류	넙비형종류불일	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	심사중	심사중	심사중	시작전	종료전	종료전	없음
24	공정중간검사	넙비형위치	넙비형위치시합예물린경우	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	심사중	심사중	심사중	시작전	종료전	종료전	없음
25	공정중간검사	원단오염	오염	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	불합격	심사중	심사중	2021-07-27 12:13:13	종료전	종료전	없음
26	공정중간검사	원단구멍	논해침의부분홀이있는경우	rep00003_WS_blosure	플라우스	2018-02-28 14:13:14	합격/불합격	심사중	심사중	심사중	시작전	종료전	종료전	없음

■JavaScript

```
$(function(){
    var value = "";
    var html = "";
    var searchKey = "";
    var searchValue = "";
    var searchObj = {};

    //계약번호 모달
    $('#bycontractNum').click(function(){
        var bindElement = $('#searchContractForm').find('input, select');
        var mainTbodyContract = $('#mainTbodyContract');

        $('#searchContractModal').click(function(){
            $('#removeTrCN').remove();
            html = "";
            searchKey = "";
            searchValue = "";
            searchObj = {};

            $.each(bindElement, function(){
                searchKey = $(this).attr('name');
                searchValue = $(this).val();

                if(searchValue != null && searchValue != undefined &&
searchValue != ""){
                    searchObj[searchKey] = searchValue;
                }
            });

            var request = $.ajax({
                url: "/quality/searchByContractNum",
                method: "POST",
                data: JSON.stringify(searchObj),
                contentType: "application/json",
                dataType: "json"
            });

            request.done(function( data ) {
                //console.log(data);
                if(data.length > 0){
                    for(var i = 0; i<data.length; i++){

                        html += "<tr class='removeTrCN'>";
                        html += "<td><input type='radio' name='check'
value='' + data[i].contractCode + "</td>";

                        html += "<td>" + [i+1] + "</td>";
                        html += "<td>" + data[i].contractCode + "</td>";
                        html += "<td>" + data[i].clientName + "</td>";
```

```

+ "</td>";

html += "<td>" + data[i].contractedAmount

html += "<td>" + data[i].contractedDate + "</td>";
html += "</tr>";

    }
    }else{
        html += "<tr class='removeTrCN'><td style='text-align:center;' colspan='6'>검색 결과가 없습니다.</td></tr>"
    }
    mainTbodyContract.append(html);

    //메인 화면 input 박스에 넣기
    $('#input[type=radio]').click(function(){
        value = $(this).val();
        $('#contractNum').val(value);
    });

});

request.fail(function( jqXHR, textStatus ) {
    alert( "Request failed: " + textStatus );
});

});

});

//거래처 검색 모달
$('#byClient').click(function(){
    var mainTbodyClient = $('#mainTbodyClient');

    $('#searchClientBtn').click(function(){
        $('#removeTrCL').remove();
        html = "";
        searchKey = "";
        searchValue = "";
        searchObj = {};

        var clientName = $('#searchClientName').val();

        var request = $.ajax({
            url: "/quality/searchByClientName",
            method: "POST",
            data: {"clientName":clientName},
            dataType: "json"
        });

        request.done(function( data ) {
            //console.log(data);
            if(data.length > 0){
                for(var i = 0; i<data.length; i++){
                    html += '<tr class="removeTrCL">';

```

```

+   data[i].clientName + "</td>";

                                html += "<td><input type='radio' name='check' value='"
                                html += '<td>' + [i+1] + '</td>';
                                html += '<td>' + data[i].clientName + '</td>';
                                html += '</tr>';

                                }
                            }else{
                                html += '<tr class="removeTrCL"><td colspan="3" style="text-align:center;">검색 결과가 없습니다</td></tr>';
                            }
                            mainTbodyClient.append(html);

                            //메인화면 input 박스에 넣기
                            $('#input[type=radio]').click(function(){
                                value = $(this).val();
                                $('#clientName').val(value);
                            });
                        });

                        request.fail(function( jqXHR, textStatus ) {
                            alert( "Request failed: " + textStatus );
                        });
                    });

                });

//품질검사 코드 검색 모달
$('#byInspectionCode').click(function(){
    var mainTbodyInsCode = $('#mainTbodyInsCode');
    var bindElement = $('#searchInsCodeForm').find('select');

    $('#searchInsCodeModal').click(function(){
        $('.removeTrIns').remove();
        html = "";
        searchKey = "";
        searchValue = "";
        searchObj = {};

        $.each(bindElement, function(){
            searchKey = $(this).attr('name');
            searchValue = $(this).val();

            if(searchKey != null && searchKey != undefined && searchKey != ""){
                searchObj[searchKey] = searchValue;
            }
        });

        var request = $.ajax({
            url: "/quality/searchInspectionCode",
            method: "POST",
            data: JSON.stringify(searchObj),
            contentType: "application/json",

```

```

        dataType: "json"
    });

    request.done(function( data ) {
        //console.log(data);
        if(data.length > 0){
            for(var i = 0; i<data.length; i++){
                html += '<tr class="removeTrIns">';
                html += "<td><input type='radio' name='check' value='"
+   data[i].qualityInspectionCode + "'></td>";

                html += '<td>'+ [i+1] + '</td>';
                html += '<td>'+ data[i].highClassificationName + '</td>';
                html += '<td>'+ data[i].mediumClassificationName
+ '</td>';

                html += '<td>'+ data[i].lowClassificationName + '</td>';
                html += '<td>'+ data[i].subClassificationName + '</td>';
                html += '</tr>';

            }
        }else{
            html += '<tr class="removeTrIns"><td colspan="3" style="text-align:center;">검색 결과가 없습니다</td></tr>';
        }
        mainTbodyInsCode.append(html);

        //메인 화면 input 박스에 넣기
        $('input[type=radio]').click(function(){
            value = $(this).val();
            $('#inspectionCode').val(value);
        });
    });

    request.fail(function( jqXHR, textStatus ) {
        alert( "Request failed: " + textStatus );
    });
});

//수주계약번호
var contractCode = "";
//품질검사 요청 총갯수
var requestCount = 0;
//최종 불합격 총갯수
var finalFailCount = 0;
//최종 합격 총갯수
var finalPassCount = 0;
//최종 완료 갯수
var finalCompleteCount = 0;
//완료율
var completeRate = 0;
//진행건수
var progressRate = 0;

```

//그래프

```
var ctx1 = document.getElementById('satateChart').getContext('2d');
```

```
var chart1 = new Chart(ctx1, {  
    type: 'doughnut',  
    data: {  
        labels: ['완료','진행율'],  
        datasets: [{  
            label: 'My First Dataset',  
            data: [50, 50],  
            backgroundColor: [  
                'rgb(54, 162, 235)',  
                'rgb(255, 205, 86)'  
            ],  
            hoverOffset: 4  
        }]  
    },  
});
```

```
var ctx2 = document.getElementById('finalPassChart').getContext('2d');
```

```
var chart2 = new Chart(ctx2, {  
    type: 'doughnut',  
    data: {  
        labels: ['합격','불합격'],  
        datasets: [{  
            label: 'My First Dataset',  
            data: [50, 50],  
            backgroundColor: [  
                'rgb(54, 162, 235)',  
                'rgb(255, 205, 86)'  
            ],  
            hoverOffset: 4  
        }]  
    },  
});
```

```
var ctx3 = document.getElementById('inspectionPass').getContext('2d');
```

```
var chart3 = new Chart(ctx3, {  
    type: 'doughnut',  
    data: {  
        labels: ['수입검사','공정검사','완제품검사'],  
        datasets: [{  
            label: 'My First Dataset',  
            data: [30, 30, 40],  
            backgroundColor: [  
                'rgb(54, 162, 235)',  
                'rgb(255, 205, 86)',  
                'rgb(120, 205, 86)'  
            ],  
            hoverOffset: 4  
        }]  
    }  
});
```



```

    },
    });

//품질검사 현황 조회
$(document).on('click','#searchBtn',function(){

    //완료율 바
    var progressBar = "";
    var statusElement = $('#statusElement').find('input, select');
    html = "";

    //계약번호 필수 입력사항
    if($('#contractNum').val()==""){
        alert('계약번호를 입력해주세요.');
        $('#contractNum').focus();
        $('#contractNum').val('buyerContract002');
    }else{

        $.each(statusElement,function(){
            searchkey = $(this).attr('name');
            searchValue = $(this).val();

            if(searchValue != null && searchValue != undefined && searchValue !=

        "){

            searchObj[searchkey] = searchValue;
        }
        //console.log(searchObj);
    });
    var request = $.ajax({
        url: "/quality/qualityInspectionStatusNow",
        data: JSON.stringify(searchObj),
        contentType: "application/json",
        method: "POST",
        dataType: "json"
    });

    request.done(function( data ) {
        //console.log(data);
        $('tr.removeTr').remove();
        //원래 있던 progressBar 지우기
        $('#originalProgressBar').remove();
        //새로 생성된 progressBar 리셋
        $('tr.progress-bar').remove();

        if(data.length > 0){

            //수주계약번호
            contractCode = data[0].contractCode;
            //품질검사 요청 총갯수
            requestCount = data[0].requestCount;

```

```

//최종 불합격 총갯수
finalFailCount = data[0].finalFailCount;
//최종 합격 총갯수
finalPassCount = data[0].finalPassCount;
//최종 완료 갯수
finalCompleteCount = data[0].finalCompleteCount;
//완료율(반올림)
completeRate =
Math.round((finalCompleteCount*100)/requestCount);
//진행건수
progressRate = data[0].progressRate;
//완료율바
progressBar += '<div class="progress-bar" role="progressbar"
aria-valuenow="' + completeRate +'" aria-valuemin="0"' +
progressBar += 'aria-valuemax="100" style="width: ' +
completeRate + '%"><span class="sr-only">' + completeRate + '% Complete</span>';

chart1.data.datasets[0].data=[progressRate, completeRate];
chart1.update();

chart2.data.datasets[0].data=[passNum, failedNum];
chart2.update();

chart3.data.datasets[0].data=[first, second, third];
chart3.update();

$('#stateMain').val(contractCode);
$('#requestCount').val(requestCount);
$('#finalCompleteCount').val(finalCompleteCount);
$('#completeRate').val(completeRate);
$('#progressRate').val(progressRate);
$('#progressBar').append(progressBar);

var html = "";
for(var i = 0; i<data.length; i++){
    var requestDate = data[i].inspectionRequestDate;

    //date 포맷 변경 yyyy-MM-dd hh:mm:ss
    function formatDate(requestDate) {
        var d = new Date(requestDate),
            month = " + (d.getMonth() + 1),
            day = " + d.getDate(),
            year = d.getFullYear(),
            hour = d.getHours(),
            min = d.getMinutes(),
            sec = d.getSeconds();

        if (month.length < 2) month = '0' + month;

```

```

        if (day.length < 2) day = '0' + day;

        return [year, month, day].join('-') + ' ' + [hour,
min, sec].join(':');

    }

    var dateText = "/Date(1519794794410)/";
    var myDate = new Date(dateText.match(/d+/) * 1);
    date = formatDate(myDate);

    //List
    html += '<tr class="removeTr">';
    html += '<th scope="row">' + [i+1] + '</th>';
    html += '<td>' + data[i].highClassName + '</td>';
    html += '<td>' + data[i].lowClassName + '</td>';
    html += '<td>' + data[i].subClassName + '</td>';
    html += '<td>' + data[i].requestProductCode + '</td>';
    html += '<td>' + data[i].detailedCateName + '</td>';
    html += '<td>' + date + '</td>';
    html += '<td>' + data[i].inspectionCategory + '</td>';
    html += '<td>' + data[i].insFirst + '</td>';
    html += '<td>' + data[i].insSecond + '</td>';
    html += '<td>' + data[i].insThird + '</td>';
    html += '<td>' + data[i].insStart + '</td>';
    html += '<td>' + data[i].insEnd + '</td>';
    html += '<td>' + data[i].finalPassCheck + '</td>';
    html += '<td>없음</td>';
    html += '</tr>';

    }

    }else{
        html += '<tr class="removeTr"><td colspan="15" style="text-align:center;">조회 결과가 없습니다.</td></tr>';
    }
    $('#mainTbody').append(html);
});
request.fail(function( jqXHR, textStatus ) {
    alert( "Request failed: " + textStatus );
});
}

});
});

```

■Controller

```

//[다미] 품질검사 코드
@PostMapping("/searchInspectionCode")
@ResponseBody
public List<QualityInspection> searchInspectionCode(@RequestBody QualityInspection
qualityInspection){
    log.info("화면에서 받은 값: {}", qualityInspection);
    List<QualityInspection> inspectionCode =
qualityInsMeasurementValueService.searchInspectionCode(qualityInspection);
    return inspectionCode;
}

//[다미] 거래처이름 검색
@PostMapping("/searchByClientName")
@ResponseBody
public List<Map<String, Object>> searchByClientName(@RequestParam(value = "clientName",
required = false)String clientName){
    log.info("화면에서 받은 값: {}", clientName);
    List<Map<String, Object>> clientNameMap =
qualityInsMeasurementValueService.searchByClientName(clientName);
    return clientNameMap;
}

//[다미] 계약번호 검색
@PostMapping("/searchByContractNum")
@ResponseBody
public List<Map<String, Object>> searchByContractNum(@RequestBody Map<String, Object>
searchByContractNum){
    log.info("화면에서 받은 값: {}", searchByContractNum);
    List<Map<String, Object>> ContractNumMap =
qualityInsMeasurementValueService.searchByContractNum(searchByContractNum);
    return ContractNumMap;
}

//[다미] 실시간 검사 현황
@PostMapping("/qualityInspectionStatusNow")
@ResponseBody
public List<Map<String, Object>> qualityInspectionStatusNow(@RequestBody Map<String,
Object> searchMap) {
    log.info("화면에서 받은 값: {}", searchMap);
    List<Map<String, Object>> map =
qualityInsMeasurementValueService.getQualityInspectionStatusNow(searchMap);

    return map;
}

//[다미] 실시간 검사 현황
@GetMapping("/qualityInspectionStatusNow")
public String qualityInspectionStatusNow(Model model) {

```

```

        return "quality/qualityInspectionStatusNow";
    }

    //검사현황 실적
    @GetMapping("/inspectionPerformance")
    public String inspectionPerformance(Model model) {

        model.addAttribute("title", "검사현황관리:실적");
        return "quality/inspectionPerformance";
    }

    //[다미] 품질검사 측정값 목록
    @GetMapping("/inspectionMeasurementValueList")
    public String inspectionMeasurementValueList(Model model) {

        List<QualityInspectionResult> list =
qualityInsMeasurementValueService.getInspectionMeasurementValueList();

        model.addAttribute("list", list);

        log.info("{} ", list);

        return "quality/inspectionMeasurementValueList";
    }
}

```

■ Service

```

//품질검사 검색
public List<QualityInspection> searchInspectionCode(QualityInspection qualityInspection){
    return qualityInsMeasurementValueMapper.searchInspectionCode(qualityInspection);
};

//거래처 검색
public List<Map<String, Object>> searchByClientName(String clientName){
    return qualityInsMeasurementValueMapper.searchByClientName(clientName);
}

//계약번호 검색
public List<Map<String, Object>> searchByContractNum(Map<String, Object>
searchByContractNum){
    return
qualityInsMeasurementValueMapper.searchByContractNum(searchByContractNum);
};

//품질검사 측정값 현황

```

```

        public List<Map<String, Object>> getQualityInspectionStatusNow(Map<String, Object>
searchMap){
            return qualityInsMeasurementValueMapper.getQualityInspectionStatusNow(searchMap);
        }

```

■SQL Mapper

```

<!-- 품질검사 코드 검색 -->
<select id="searchInspectionCode" resultType="QualityInspection"
parameterType="QualityInspection">
    SELECT
        quality_inspection_code as qualityInspectionCode
        ,high.high_class_name as highClassificationName
        ,med.med_class_name as mediumClassificationName
        ,low.low_class_name as lowClassificationName
        ,sub.sub_class_name as subClassificationName
    from
        sub_class_inspection_list AS sub
    INNER JOIN
        low_class_inspection_list AS low ON sub.high_med_low_class_code =
low.high_med_low_class_code
    INNER JOIN
        medium_class_inspection_list AS med ON low.high_med_class_code =
med.high_med_class_code
    INNER JOIN
        high_class_inspection_list AS high ON med.high_class_code =
high.high_class_code
    <trim prefix="WHERE" prefixOverrides="AND">
        <if test="highClassificationName != null and highClassificationName !=
".toString()">
            high.high_class_code=#{highClassificationName}
        </if>
        <if test="mediumClassificationName != null and mediumClassificationName !=
".toString()">
            AND med.high_med_class_code=#{mediumClassificationName}
        </if>
        <if test="lowClassificationName != null and lowClassificationName !=
".toString()">
            AND low.high_med_low_class_code=#{lowClassificationName};
        </if>
    </trim>
</select>

<!-- 품질검사 현황 거래처 검색 -->
<select id="searchByClientName" resultType="Map" parameterType="String">
    SELECT
        ci.client_name AS clientName
    FROM
        client_info AS ci
    INNER JOIN

```

```

        requested_product_detail AS r ON ci.client_code = r.client_code
    INNER JOIN
        contract_info AS co ON co.contract_code = r.contract_code
    <trim prefix="WHERE">
        <if test="clientName != null and clientName != '.toString()'">
            ci.client_name LIKE CONCAT ('%', #{clientName}, '%')
        </if>
    </trim>
    GROUP BY
        ci.client_name;
</select>

<!-- 품질검사 현황 계약번호 검색 -->
<select id="searchByContractNum" resultType="Map" parameterType="Map">
    SELECT
        co.contract_code AS contractCode
        ,ci.client_name AS clientName
        ,co.contracted_amount AS contractedAmount
        ,co.contracted_date AS contractedDate
    FROM
        client_info AS ci
    INNER JOIN
        requested_product_detail AS r ON ci.client_code = r.client_code
    INNER JOIN
        contract_info AS co ON co.contract_code = r.contract_code
    <trim prefix="WHERE" prefixOverrides="AND">
        <if test="contractCode != null and contractCode != '.toString()'">
            co.contract_code LIKE CONCAT ('%', #{contractCode}, '%')
        </if>
        <if test="clientName != null and clientName != '.toString()'">
            AND ci.client_name LIKE CONCAT ('%', #{clientName}, '%')
        </if>
        <if test="contractedStrartDate != null and contractedStrartDate != '.toString()'">
            AND co.contracted_date >= DATE#{contractedStrartDate}
        </if>
        <if test="contractedEndDate != null and contractedEndDate != '.toString()'">
            AND co.contracted_date <= DATE#{contractedEndDate}
        </if>
    </trim>
    GROUP BY
        co.contract_code
    <trim prefix="ORDER BY" suffixOverrides="WHERE">
        co.contracted_date ${range};
    </trim>
</select>

<!-- 품질검사 현황 -->
<select id="getQualityInspectionStatusNow" resultType="Map" parameterType="Map">
    SELECT
        (select
            COUNT(*)
        from

```

```

            inspection_result AS i
WHERE
            i.inspection_measurement_num = '1 회차') AS 'progressRate'
,
(select
            COUNT(*)
        from
            inspection_final_result AS i
        WHERE
            i.low_cate_final_pass_check='합격') AS finalPassCount
,(select
            COUNT(*)
        from
            inspection_final_result AS i
        WHERE
            i.low_cate_final_pass_check='불합격') AS finalFailCount
,(SELECT
            COUNT(*)
        FROM
            quality_inspection_request AS q) AS requestCount
,(SELECT
            COUNT(*)
        FROM
            inspection_final_result AS f) AS finalCompleteCount
,re.quality_inspection_request_code AS inspectionRequestCode
,high.high_class_name AS highClassName
,med.med_class_name AS medClassName
,low.low_class_name AS lowClassName
,sub.sub_class_name AS subClassName
,p.requested_product_code AS requestProductCode
,pc.detailed_categorized_name AS detailedCateName
,q.inspection_request_date AS inspectionRequestDate
,res.category AS inspectionCategory
,ifnull(group_concat(if(re.inspection_measurement_num='1 회차',
re.inspection_pass_check, NULL)),'심사중') AS insFirst
,ifnull(group_concat(if(re.inspection_measurement_num='2 회차',
re.inspection_pass_check, NULL)),'심사중') AS insSecond
,ifnull(group_concat(if(re.inspection_measurement_num='3 회차',
re.inspection_pass_check, NULL)),'심사중') AS insThird
,ifnull(group_concat(if(re.inspection_measurement_num='1 회차',
re.inspection_start_date, NULL)),'시작전') AS insStart
,ifnull(group_concat(if(re.inspection_measurement_num='3 회차',
re.inspection_end_date, NULL)),'종료전') AS insEnd
,IFNULL(ref.low_cate_final_pass_check,'종료전') AS finalPassCheck
,p.contract_code AS contractCode
FROM
            requested_product_detail AS p
INNER JOIN
            quality_inspection_request AS q ON p.requested_product_code =
q.requested_product_code

```



```

LEFT JOIN
    inspection_result AS re ON q.quality_inspection_request_code =
re.quality_inspection_request_code
LEFT JOIN
    inspection_final_result AS ref ON re.quality_inspection_request_code =
ref.quality_inspection_request_code
INNER JOIN
    inspection_standard AS res ON q.quality_inspection_code =
res.quality_inspection_code
INNER JOIN
    sub_class_inspection_list AS sub ON q.quality_inspection_code =
sub.quality_inspection_code
INNER JOIN
    low_class_inspection_list AS low ON sub.high_med_low_class_code =
low.high_med_low_class_code
INNER JOIN
    medium_class_inspection_list AS med ON low.high_med_class_code =
med.high_med_class_code
INNER JOIN
    high_class_inspection_list AS high ON med.high_class_code =
high.high_class_code
INNER JOIN
    product_code_detail AS pc ON p.product_code = pc.product_code
INNER JOIN
    client_info AS c ON p.client_code = c.client_code
<trim prefix="WHERE" prefixOverrides="AND">
    <if test="contractNum != null and contractNum != ".toString()">
        p.contract_code = #{contractNum}
    </if>
    <if test="clientName != null and clientName != ".toString()">
        AND c.client_name LIKE CONCAT ('%', #{clientName} ,'%')
    </if>
    <if test="requestStartDate != null and requestStartDate != ".toString()">
        AND q.inspection_request_date >= DATE(#{requestStartDate})
    </if>
    <if test="requestEndDate != null and requestEndDate != ".toString()">
        AND q.inspection_request_date <= DATE(#{requestEndDate})
    </if>
    <if test="genderCate != null and genderCate != ".toString()">
        AND pc.gender_categorized_name LIKE CONCAT ('%',
#{genderCate} ,'%')
    </if>
    <if test="productCode != null and productCode != ".toString()">
        AND pc.detailed_categorized_code LIKE CONCAT ('%',
#{productCode} ,'%')
    </if>
    <if test="inspectionCode != null and inspectionCode != ".toString()">
        AND sub.quality_inspection_code LIKE CONCAT ('%',
#{inspectionCode} ,'%')
    </if>
    <if test="startDate != null and startDate != ".toString()">
        AND re.inspection_start_date >= DATE(#{startDate})

```

```
        </if>
        <if test="endDate != null and endDate != ''">
            AND re.inspection_start_date <= DATE("#{endDate}")
        </if>
    </trim>
    GROUP BY q.quality_inspection_request_code;
</select>
```