

| 중단원 | 소단원 | 번호 | 문제 | 키워드 | 답 |
|-----|---------------|----|-------------------------------------------------------------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | 1. 연계 요구사항 분석 | 1 | 연계 요구사항 분석 기법 종류 | 연계 요구사항 분석 | * 인터뷰 * 체크리스트 * 설문지 * 델파이 기법 * 브레인스토밍 |
| | | 2 | 체크 리스트 개념 | 연계 요구사항 분석 | 시스템 운영 환경, 성능, 보안, 데이터 발생 주기 등의 기준에 대한 점검을 통한 분석 기법 |
| | | 3 | 설문지 개념 | 연계 요구사항 분석 | 서비스 활용 목적에 따라 연계 필요 데이터 식별 및 연계 주기 등을 설문지를 통한 분석 기법 |
| | | 4 | 델파이 기법 개념 | 연계 요구사항 분석 | * 전문가의 경험적 지식을 통한 분석 기법 * 업무 전문가 등 각 분야 전문가로 구성 |
| | | 5 | 브레인스토밍 기법 개념 | 연계 요구사항 분석 | * 소속된 인원들이 자발적으로 자연스럽게 제시된 아이디어 목록을 통해서 연계 요구사항을 도출하는 요구사항 분석 기법이다 * 창의적인 아이디어를 생산하기 위한 학습 도구이자 회의 기법 * 주제에 제한이 없고 실행하기 쉽고 질보다 양에 초점을 맞추기에 많은 아이디어를 얻을 수 있는 것이 장점 |
| | | 6 | 연계 요구사항 분석 참고문서 종류 | 연계 요구사항 분석 | 1. 코드 정의서 2. 테이블 정의서 3. 응용 프로그램 구성도 4. 시스템 구성도 |
| | | 7 | 연계 요구사항 분석 참고문서 종류 및 개념 | 연계 요구사항 분석 | 1. 코드 정의서 - 공통 코드 정의서 (코드 ID, 코드명, 코드설명 등) 2. 테이블 정의서 - 데이터 모델링 정의서, 테이블/프로세스 연관도, 테이블별 컬럼 속성 정의서 3. 응용 프로그램 구성도 - 애플리케이션의 메뉴구성, 화면설계, 데이터 발생 시점, 주기, 발생 패턴 등 확인 4. 시스템 구성도 - 하드웨어, 소프트웨어, 네트워크 등 연계 대상 시스템 구성도 |
| | | 8 | 연계 요구사항 분석 참고문서 종류 | 연계 요구사항 분석 | 1. 개체 정의서 2. 테이블 정의서 3. 인터페이스 명세서 / 연계 명세서 |
| | | 9 | 개체 정의서 개념 | 연계 요구사항 분석 | 데이터베이스 개념 모델링 단계에서 도출한 개체의 타입과 관련 속성, 식별자 등의 정보를 개괄적으로 명세화한 정의서 |
| | | 10 | 다음 설명하는 연계 요구사항 분석 참고문서는? 논리 DB명 엔티티명 엔티티 설명 주 식별자 슈퍼타입 엔티티명 | 연계 요구사항 분석 | 개체 정의서 |
| | | 11 | 테이블 정의서 개념 | 연계 요구사항 분석 | 논리 및 물리 모델링 과정에서 작성하는 설계 산출물로 테이블을 구성하는 컬럼들의 특성, 인덱스 업무 규칙을 문서화한 것이다 |
| | | 12 | 인터페이스 명세서 개념 | 연계 요구사항 분석 | 연계(인터페이스) 정의서에 작성한 인터페이스 ID별로 송수신하는 데이터 타입, 길이 등 인터페이스 항목을 상세하게 작성한 것 |

| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|-----------|-----------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|-------|--------|--------|--------|--|--------|--------|-----------------------------------------------|------|--------|--|--------|------|--|----------|------|----------------------|------|----------|-------|
| 01. 연계 데이터 구성 | | 13 | <p>다음 설명하는 연계 요구사항 분석 참고문서는?</p> <p>인터페이스 ID 송수신 구분 시스템명 업무 서비스명</p> <p>[데이터 항목] Seq Field Key Type Size null 허용 Description condition</p> | 연계 요구사항 분석 | 인터페이스 명세서 / 연계 명세서 | | | | | | | | | | | | | | | | | | | | | |
| | | 14 | 연계 시스템 구성 | 연계 시스템 | <ul style="list-style-type: none"> * 송신 시스템 * 수신 시스템 * 중계 서버 | | | | | | | | | | | | | | | | | | | | | |
| | | 15 | <p>연계 시스템 구성 = 인터페이스 시스템 구성</p> | 연계 시스템 | <ul style="list-style-type: none"> * 송신 시스템 <ul style="list-style-type: none"> - 연계할 데이터를 데이터베이스와 애플리케이션으로부터 연계 테이블 또는 파일 형태로 생성하여 송신하는 시스템 * 수신 시스템 <ul style="list-style-type: none"> - 수신한 연계 테이블 또는 파일의 데이터를 수신 시스템에서 관리하는 데이터 형식에 맞게 변환하여 데이터베이스에 저장하거나 애플리케이션에서 활용할 수 있도록 제공하는 시스템 * 중계 서버 <ul style="list-style-type: none"> - 송신 시스템과 수신 시스템 사이에서 데이터를 송수신하고 연계 데이터의 송수신 현황을 모니터링 하는 시스템 - 연계 데이터의 보안 강화 및 다중 플랫폼 지원 등이 가능 | | | | | | | | | | | | | | | | | | | | | |
| 2. 연계 데이터 식별 및 표준화 | | 1 | 연계 데이터 범위 검토사항 | 연계 데이터 식별 | <table border="1"> <tr> <td>* 송신부</td> <td>* 수신부</td> <td>* 검토사항</td> </tr> <tr> <td>String</td> <td>Number</td> <td></td> </tr> <tr> <td>Number</td> <td>String</td> <td>한 가지의 데이터 타입으로 정의 및 적용 (일반적으로 수신 시스템 기준으로 적용)</td> </tr> <tr> <td>Date</td> <td>String</td> <td></td> </tr> <tr> <td>String</td> <td>Date</td> <td></td> </tr> <tr> <td>Not Null</td> <td>Null</td> <td>송신 시스템에 입력된 값 그대로 연계</td> </tr> <tr> <td>Null</td> <td>Not Null</td> <td>연계 불가</td> </tr> </table> | * 송신부 | * 수신부 | * 검토사항 | String | Number | | Number | String | 한 가지의 데이터 타입으로 정의 및 적용 (일반적으로 수신 시스템 기준으로 적용) | Date | String | | String | Date | | Not Null | Null | 송신 시스템에 입력된 값 그대로 연계 | Null | Not Null | 연계 불가 |
| | | * 송신부 | * 수신부 | * 검토사항 | | | | | | | | | | | | | | | | | | | | | | |
| String | Number | | | | | | | | | | | | | | | | | | | | | | | | | |
| Number | String | 한 가지의 데이터 타입으로 정의 및 적용 (일반적으로 수신 시스템 기준으로 적용) | | | | | | | | | | | | | | | | | | | | | | | | |
| Date | String | | | | | | | | | | | | | | | | | | | | | | | | | |
| String | Date | | | | | | | | | | | | | | | | | | | | | | | | | |
| Not Null | Null | 송신 시스템에 입력된 값 그대로 연계 | | | | | | | | | | | | | | | | | | | | | | | | |
| Null | Not Null | 연계 불가 | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | 연계 시스템 구성 | 연계 데이터 식별 및 표준화 절차 | <ol style="list-style-type: none"> 연계 범위 및 항목 정의 <ul style="list-style-type: none"> - 시스템 간의 연계하려는 정보를 구체화하여 범위와 항목을 정의 - 송신 시스템과 수신 시스템에서 연계하고자 하는 각 항목의 데이터 타입 및 길이, 코드화 여부 등을 확인 연계 코드 변환 및 매핑 <ul style="list-style-type: none"> - 연계대상 범위 및 항목 식별 이후, 연계 정보 중 코드로 관리되는 항목을 변환해야 함 - 코드로 관리되는 정보는 검색 조건으로 활용 인터페이스(연계) 정의서 작성 <ul style="list-style-type: none"> - 정의된 연계 항목의 데이터 타입, 길이, 공통 코드 여부 등의 속성을 확인하고 연계(인터페이스) 정의서 및 명세서 작성 | | | | | | | | | | | | | | | | | | | | | | | |

| | | | | | |
|--|--|---|------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | 3 | 연계 데이터 표준화 개념 | 연계 데이터 표준화 | <ul style="list-style-type: none"> * 인터페이스 시스템 사이에서 교환되는 데이터는 규격화된 표준 형식을 정의하여 사용하는 규칙이다 * 인터페이스 설계 단계에서 송/수신 시스템 사이에 전송되는 표준항목과 업무 처리 데이터, 공통 코드 정보 등을 누락 없이 식별하고 인터페이스 명세서를 작성해야 한다 * 인터페이스 데이터 공통부, 인터페이스 데이터 개별부, 인터페이스 종료부로 구성되어 있다 |
| | | 4 | 연계 데이터 표준화 구성 | 연계 데이터 표준화 | <ul style="list-style-type: none"> * 인터페이스 데이터 공통부 * 인터페이스 데이터 개별부 * 인터페이스 데이터 종료부 |
| | | 5 | 연계 데이터 표준화 구성 | 연계 데이터 표준화 | <ul style="list-style-type: none"> * 인터페이스 데이터 공통부 - 인터페이스 표준 항목 포함 * 인터페이스 데이터 개별부 - 송/수신 시스템에서 업무 처리에 필요한 데이터 포함 * 인터페이스 데이터 종료부 - 전송 데이터의 끝을 표시하는 문자를 포함하여 종료 표시 |
| | | 1 | 연계 메커니즘 개념 | 연계 메커니즘 | 응용 소프트웨어와 연계 대상 모듈 간의 데이터 연계 시 요구사항을 고려한 연계방법과 주기를 설계하기 위한 흐름이다 |
| | | 2 | 연계 메커니즘 구성 | 연계 메커니즘 | <ul style="list-style-type: none"> * 데이터를 생성하여 전송하는 송신 시스템 * 데이터를 수신하여 DB에 반영하는 수신 시스템 <p>으로 구성된다</p> |
| | | 3 | 연계 메커니즘 연계 방식 종류 | 연계 메커니즘 | <ul style="list-style-type: none"> * 직접 연계 방식 + 연계 및 통합 구현이 단순하며 용이함 + 개발 소요 비용 및 기간이 짧음 + 중간 매개체가 없으므로 데이터 연계 처리 성능이 대체로 좋음 - 시스템 간의 결합도가 높아 시스템 변경에 민감 - 암/복호화 처리 불가 - 연계 및 통합 가능한 시스템 환경이 제한적 * 간접 연계 방식 + 서로 상이한 네트워크, 프로토콜 연계 및 통합 가능 + 인터페이스 변경 시에도 장애나 오류 없이 서비스 가능 + 보안이나 업무 처리 로직을 자유롭게 반영 가능 - 연계 아키텍처 및 메커니즘이 복잡해 성능 저하 요소 존재 - 개발 및 적용을 위한 테스트 기간이 상대적 장기간 소요 |

| | | | | | |
|----------------|---------------|----|----------------------------------------------------------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 02. 연계 메커니즘 구성 | 1. 연계 메커니즘 정의 | 4 | 연계 메커니즘 연계 방식 종류 | 연계 메커니즘 | <ul style="list-style-type: none"> * 직접 연계 방식 <ul style="list-style-type: none"> - DB 링크 (DB Link) - DB 연결(DB Connection) - API/Open API - JDBC - 하이퍼링크 * 간접 연계 방식 <ul style="list-style-type: none"> - 연계 솔루션 (EAI) - Web Service/ESB - 소켓 (Socket) |
| | | 5 | 연계 기술 중 DB 링크 개념 | 직접 연계 방식 | <ul style="list-style-type: none"> * 데이터베이스에서 제공하는 DB 링크 객체를 이용 * 수신 시스템에서 DB 링크를 생성하고 송신 시스템에서 해당 DB 링크를 직접 참조하는 방식이다 * 표현 식은 테이블명@DBLINK명으로 사용한다 |
| | | 6 | 연계 기술 중 DB 연결(DB Connection) 개념 | 직접 연계 방식 | 수신 시스템의 WAS에서 송신 시스템 DB로 연결하는 DB 커넥션 풀(DB Connection Pool)을 생성하고 연계 프로그램에서 해당 DB 커넥션 풀 명을 이용 |
| | | 7 | 연계 기술 중 JDBC 개념 | 직접 연계 방식 | <ul style="list-style-type: none"> * 수신 시스템의 프로그램에서 JDBC 드라이버를 이용하여 송신 시스템 DB와 연결 * DBMS 유형, DBMS 서버 IP와 Port, DB 인스턴스(Instance) 정보가 필요하다 |
| | | 8 | 연계 기술 중 하이퍼링크 개념 | 직접 연계 방식 | <ul style="list-style-type: none"> * 현재 페이지에서 다른 부분으로 가거나 전혀 다른 페이지로 이동하게 해주는 속성 * ex) Link 대상 |
| | | 9 | 간접 연계 방식 분류 | 간접 연계 방식 | <ul style="list-style-type: none"> * 연계 솔루션 (EAI) * Web Service / ESB * 소켓 (Socket) |
| | | 10 | 웹 서비스 기술 중 SOAP(Simple Object Access Protocol) 개념 간략히 서술 | 연계 메커니즘 | HTTP, HTTPS, SMTP 등을 사용하여 XML 기반의 메시지를 네트워크 상태에서 교환하는 프로토콜이다 |
| | | 11 | 간접 연계 방식 분류 | 간접 연계 방식 | <ul style="list-style-type: none"> * 연계 솔루션 (EAI) <ul style="list-style-type: none"> - 기업에서 운영되는 서로 다른 플랫폼 및 애플리케이션들 간의 정보 전달, 연계, 통합을 가능하게 해주는 솔루션 - 송수신 시스템에 설치되는 어댑터를 이용 * Web Service / ESB <ul style="list-style-type: none"> - 웹 서비스가 설명된 WSDL과 SOAP 프로토콜을 이용한 시스템 간 연계 * 소켓 (Socket) <ul style="list-style-type: none"> - 소켓을 생성하여 포트를 할당하고, 클라이언트의 요청을 연결하여 통신 |

| | | | | | |
|---|--|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | 12 | <ul style="list-style-type: none"> * 송신 시스템 및 송신 모듈 1. 연계 데이터 생성 및 추출 <li style="padding-left: 20px;">2. 오류 발생 확인 → Yes일 경우, Log 기록 3. 코드 매핑 및 데이터 변환 <li style="padding-left: 20px;">4. 오류 발생 확인 → Yes일 경우, Log 기록 5. I/F 테이블 또는 파일 생성 6. 연계 서버 또는 송신 Adapter 7. 전송 | 연계 메커니즘 수행 절차 | <ul style="list-style-type: none"> * 수신 시스템 및 수신 모듈 1. 수신 Adapter 2. I/F 테이블 또는 파일 생성 <li style="padding-left: 20px;">3. 오류 발생 확인 → Yes일 경우, Log 기록 4. 코드 매핑 및 데이터 변환 <li style="padding-left: 20px;">5. 오류 발생 확인 → Yes일 경우, Log 기록 6. 운영 DB에 연계 데이터 반영 |
| | | 13 | 로그 기록의 로그 테이블 정의서 내용 | 연계 메커니즘 수행 절차 | 한글명 영문명 데이터 타입 길이 PK Description 송신일시 SEND_DATE DATETIME 20 Y 송신 일자 명시 |
| | | 1 | 연계 서버 장애 시 대응 주요 점검 사항 | 연계 장애 및 오류처리 | <ul style="list-style-type: none"> 1. 연계 서버 엔진에서 기록되는 로그 파일의 내용 확인 2. 연계 서버의 정상적 실행 여부 3. 연계 서버의 데이터 전송 변환 과정의 오류 여부 4. 송/수신 대상 시스템의 정상 연결 여부(아이피 및 포트 접근 가능 여부) 5. 송/수신 연계 테이블(파일) 접근 권한 등 |
| 2 | | 연계 장애 및 오류 유형 | 연계 장애 및 오류처리 | <ul style="list-style-type: none"> * 연계 시스템 오류 - 연계 서버의 실행 여부 및 송/수신, 전송 형식 변환 등의 오류 - 서버의 기능과 관련된 장애 및 오류 * 연계 프로그램 오류 - 데이터 추출을 위한 DB 접근 시 권한 불충분(송신) - 운영 DB에 반영하기 위한 DB 접근 권한 불충분(수신) - 데이터 변환 및 반영 시 예외 처리(Exception) 미처리(공통) * 연계 데이터 오류 - 송신 시스템에서 추출된 연계 데이터가 유효하지 않은 값으로 인한 오류 | |

| | | | | | |
|--|--------------------|---|----------------------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | 2. 연계 장애 및 오류처리 구현 | 3 | 연계 장애 사례 및 처리방안 | 연계 장애 및 오류처리 | <ul style="list-style-type: none"> * 연계 시스템 오류 <ul style="list-style-type: none"> ① 사례 <ul style="list-style-type: none"> - 연계 서버 미 구동 - 송/수신 시스템의 아이피(IP) 및 포트(Port) 접속 불가 ② 처리 방안 <ul style="list-style-type: none"> - 연계 서버의 로그를 확인하여 원인 분석 후 처리 * 연계 프로그램 오류 <ul style="list-style-type: none"> ① 사례 <ul style="list-style-type: none"> - 등록되지 않은 코드로 매핑 불가 - 등록된 데이터가 존재하지 않음 ② 처리 방안 <ul style="list-style-type: none"> - 송/수신 연계 프로그램 기록 로그 확인 및 처리 - 데이터 재전송 및 재반영 * 연계 데이터 오류 <ul style="list-style-type: none"> ① 사례 <ul style="list-style-type: none"> - 유효하지 않은 일자 ② 처리 방안 <ul style="list-style-type: none"> - 연계 프로그램 로그를 확인하여 데이터 보정 후 재전송 |
| | | 1 | 연계 모듈 기능 구현 방식 2가지 | 연계 모듈 기능 구현 | <ul style="list-style-type: none"> * EAI / ESB 방식 * 웹 서비스 방식 |
| | | 2 | 연계 모듈 기능 구현 방법 중, "EAI 방식" 개념 설명 | 연계 모듈 기능 구현 | 기업에서 운영되는 서로 다른 플랫폼 및 애플리케이션들 간의 정보 전달, 연계, 통합을 가능하게 해주는 솔루션이다 |
| | | 3 | 연계 모듈 기능 구현 방법 중, "ESB 방식" 개념 설명 | 연계 모듈 기능 구현 | <ul style="list-style-type: none"> * 기업에서 운영되는 서로 다른 플랫폼 및 애플리케이션들 간을 하나의 시스템으로 관리 운영할 수 있도록 서비스 중심의 통합을 지향하는 아키텍처 또는 기술이다 * ESB는 버스를 중심으로 각각 프로토콜이 호환이 가능하도록 애플리케이션의 통합을 느슨한 결합 방식으로 지원하는 방식이다 |
| | | 4 | 느슨한 결합(Loosely Coupled) 개념 | ESB 방식 | 특정 서비스를 변경하더라도 연결된 다른 서비스에는 영향을 주지 않는 유연한 구조이다 |

| | | | | |
|------------------------|----|--------------------------------------------------------------|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. 연계 모듈 구현 하겨 그서 미 개바 | 5 | 트리거 작성 문법 (1) | EAI / ESB 방식 | 1. CREATE OR REPLACE TRIGGER 프로그램명 → [프로그램명]으로 트리거 생성 2. AFTER INSERT OR UPDATE OR DELETE ON 테이블명 FOR EACH ROW → 연계 데이터를 추출할 테이블 각 행에 INSERT, UPDATE, DELET가 발생할 때마다 수행 3. DECLARE AFTER_CODE VARCHAR2(5); → 선언부 → 사용할 변수 선언 4. BEGIN → 트리거 시작 5. IF INSERTING THEN (...) → 데이터가 추가된 경우 |
| | 6 | 트리거 작성 문법 (2) | EAI / ESB 방식 | 6. ELSIF DELETING THEN (...) → 데이터가 삭제된 경우 7. ELSE IF UPDATING("") → 데이터가 수정된 경우 8. EXCEPTION WHEN NO_DATA_FOUND THEN (...) → 오류 처리 9. END; → 트리거 종료 |
| | 7 | 연계 모듈 기능 구현 방법 중, "웹 서비스 방식" 개념 설명 | 연계 모듈 기능 구현 | 네트워크에 분산된 정보를 서비스 형태로 개방하여 표준화된 방식으로 공유하는 기술로써, 서비스 지향 아키텍처 개념을 실현하는 대표적인 기술이다 |
| | 8 | 웹 서비스 방식 유형 | 웹 서비스 방식 | * SOAP (Simple Object Access Protocol) * WSDL (Web Service Description Language) * UDDI (Universal Description Discovery and Integration) |
| | 9 | SOAP 개념 | 웹 서비스 방식 | HTTP, HTTPS, SMTP 등을 사용하여 XML 기반의 메시지를 네트워크 상태에서 교환하는 프로토콜이다 |
| | 10 | WSDL(Web Service Description Language) 개념 서술 | 웹 서비스 방식 | 웹 서비스명, 제공 위치, 메시지 포맷, 프로토콜 정보 등 웹 서비스에 대한 상세 정보를 XML 형식으로 기술한 파일이다 |
| | 11 | UDDI (Universal Description Discovery and Integration) 개념 서술 | 웹 서비스 방식 | 웹 서비스에 대한 정보인 WSDL을 등록하고 검색하기 위한 저장소로 공개적으로 접근, 검색이 가능한 레지스트리이다 |

| | | | | | |
|------------------|------------------------|----|----------------------------------------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 03. 내외부 연계 모듈 구현 | 1. 연계 모듈 구현 환경 구성 및 개발 | 12 | 웹 서비스 방식 유형 간단 총 정리 | 웹 서비스 방식 | <ul style="list-style-type: none"> * SOAP (Simple Object Access Protocol) <ul style="list-style-type: none"> - XML 기반 메시지를 네트워크 상태에서 교환하는 프로토콜 - HTTP, HTTPS, SMTP 등을 이용 * WSDL (Web Service Description Language) <ul style="list-style-type: none"> - 웹 서비스에 대한 상세 정보를 XML로 기술한 파일 * UDDI (Universal Description Discovery and Integration) <ul style="list-style-type: none"> - 웹 서비스에 대한 정보인 WSDL을 등록하고 검색하기 위한 저장소 - 공개적 접근, 검색이 가능한 레지스트리 |
| | | 13 | 웹 서비스 방식 연계 모듈 구현 절차 | 웹 서비스 방식 | <ol style="list-style-type: none"> 1. 송/수신 파일 경로 및 파일명 정의 2. 송신 연계 응용 프로그램 구현 3. 파일 전송 4. 수신 DB 반영 서비스 호출 처리 5. 수신 연계 응용 프로그램 구현 |
| | | 14 | REST (Representational State Transfer) | 웹 서비스 방식 | HTTP URI를 통해 자원을 명시하고, HTTP 메소드(POST, GET, PUT, DELETE)를 통해 해당 자원에 대한 생성, 조회, 갱신, 삭제 등의 명령을 적용하는 것을 의미하는 기술이다 |
| | | 15 | SOAP 활용한 XML 방식 구현 | 웹 서비스 방식 | <ul style="list-style-type: none"> * Header : 선택사항 <pre><SOAP-ENV:Envelope xmlns:SOAP-ENV="" SOAP-ENV:encodingStyle=""> <soap:header> (...) </soap:header></pre> * Body : 필수사항 <pre><SOAP-ENV:body> <m:NoteSrch xmlns:m=""> </SOAP-ENV:Body> </SOAP-ENV:Envelope></pre> |
| | | 16 | WSDL 문법구조 | 웹 서비스 방식 | <ul style="list-style-type: none"> * 서비스 인터페이스 <pre><wsdl:definitions> <wsdl:types> 데이터 타입 선언 </wsdl:types> <wsdl:message> 메시지 인자와 리턴값 선언 </wsdl:message> <wsdl:portType> 인터페이스 정의 <wsdl:operation> 메서드 선언 </wsdl:operation> </wsdl:portType></pre> * 서비스 구현 <pre><wsdl:binding> 네트워크 프로토콜 및 포트매핑 </wsdl:binding> <wsdl:service> 서비스 정의 <wsdl:port> 웹 서비스 URL </wsdl:port> </wsdl:service> </wsdl:definitions></pre> |

| | | | | |
|--|----|--------------------------------------------------------------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | 17 | WSDL 문법구조 상세 설명 | 웹 서비스 방식 | <ul style="list-style-type: none"> * WSDS 1.1 WSDL 2.0 설명 * service service 웹 기반 프로토콜에 노출되는 시스템 함수의 집합을 포함 * port endpoint 웹 서비스로의 주소 또는 연결 지점을 정의 * binding binding 인터페이스를 규정하고 SOAP 바인딩 스타일(RPC/Document) 과 트랜스포트(SOAP 프로토콜)을 정의 * porttype interface 웹 서비스를 정의 * operation operation SOAP의 동작 및 메시지 인코딩 방식을 정의 * message 없음 보통 message는 operation에 상응 * types types 데이터를 기술 |
| | 18 | 수신 연계 응용 프로그램 구현 코드 XML의 정보를 추출하여 운영 DB에 INSERT하는 연계 응용 프로그램 작성 | 웹 서비스 방식 | 수제비 정보처리기사 실기 1의 pp. 3-23 ~ 3-24 참고 |
| | 1 | 송신 연계 테스트 검증 사례 | 송신 연계 테스트 검증 | <ul style="list-style-type: none"> * 프로세스 : 연계 데이터 생성 ① 테스트 결과 <ul style="list-style-type: none"> - 선택 SQL, 프로그램 오류 - 데이터 객체 생성 오류 ② 검증 방법 <ul style="list-style-type: none"> - 사용자에게 오류 발생 시 알람이 발생되도록 프로그램 개선 (오류 원인을 포함) - 예외 케이스가 발생하지 않도록 프로그램 개선 * 프로세스 : 연계 테이블에 입력 ① 테스트 결과 <ul style="list-style-type: none"> - 입력 SQL 오류 - 데이터 정합성 오류 ② 검증 방법 <ul style="list-style-type: none"> - 입력 실패 결과와 원인을 연계 테이블에 기록 - 예외 케이스를 분석하여 발생하지 않도록 프로그램 개선 * 프로세스 : 연계 데이터 전송 ① 테스트 결과 <ul style="list-style-type: none"> - DB Connection(네트워크, 서버 설정 등) 오류 ② 검증 방법 <ul style="list-style-type: none"> - 통신 결과를 읽어서 실패 결과와 원인을 연계 테이블에 등록 - DB Connection 재설정 후 다시 테스트 수행 |

2. 연계 테스트 및 검증

| | | | |
|---|-----------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2 | 송신 연계 테스트 검증 사례 | 수신 연계 테스트 검증 | <p>* 프로세스 : 연계 데이터 읽음</p> <p>① 테스트 결과</p> <ul style="list-style-type: none"> - 데이터 선택 시 오류(SQL, 프로그램) <p>② 검증 방법</p> <ul style="list-style-type: none"> - 수신 측 사용자에게 예외 발생 알람(원인 포함) - 연계 테이블에 예외 사항 기록 - 재발되지 않도록 프로그램 개선 <p>* 프로세스 : 데이터 트랜잭션</p> <p>① 테스트 결과</p> <ul style="list-style-type: none"> - 데이터 트랜잭션 시 프로그램의 논리상 오류 <p>② 검증 방법</p> <ul style="list-style-type: none"> - 사용자에게 예외 알람 - 연계 테이블에 예외 사항 기록 - 재발되지 않도록 프로그램 개선 <p>* 프로세스 : 처리 결과 응답</p> <p>① 테스트 결과</p> <ul style="list-style-type: none"> - DB Connection(네트워크, 서버 설정 등) 오류 <p>② 검증 방법</p> <ul style="list-style-type: none"> - 연계 테이블에 예외 사항 기록 - DB Connection 재설정 |
| 3 | | | |

57 개