\equiv 국비지원과정 교육원소개 마이페이지 취업지원센터 채용정보 프로젝트 커뮤니티 산학연계 상담센터 국비무료교육 온라인상담 개강과정 시간표조회 방문상담예약 전화상담예약 국비대상자 간편조회 온라인 수강신청 FAQ 배기원 스마트 콘텐츠 융합 응용SW 엔지니어 양성과정(888H) [강남 M] 2020. 12.02 ~ 2021.07.08 15:30~22:00 김동현 강사 | 진은지 취업담임 남은 시간 11:33:39 수강생 평가 * 첨부파일의 확장자를 소문자로 등록하세요. - 예시: test.jpg(0), test.JPG(X) [NCS전공교과] 인터페이스 구현 (서술형(신)) 1회차 ~ 총점: 100.0 1. resultType과 resultMap의 차이점에 대해 설명하시오. (10점) 결과 점수 (10.0점) resultType - 클래스명 전체 또는 alias를 입력 즉 매핑하려는 자바 클래스의 전체 경로를 입력함 - 예 : com.test.Student 객체로 쿼리 실행 결과값을 받고자 할 때 <select id="selectTest" resultType="com.test.Student"> </select> - 예 : int 형 객체로 쿼리 실행 결과값을 받을 때 <select id="selectTest" resultType="int"> </select> 수강생 답 resultMap - resultMap 선언 당시 참조로 사용한 이름을 입력 - resultType을 이용하면 자동 매핑되기 때문에 편리하지만 제한이 있으나, resultMap을 사용하면 개발자가 직접 원하는 POJO 클래스에 매핑 가능 <resultMap id="test" type="com.test.Student"> <result property="name" column="name"> </resultMap> <select id="selectTest" resultMap="test"> </select> resultType - 클래스명 전체 또는 alias를 입력 즉 매핑하려는 자바 클래스의 전체 경로를 입력함 - 예: com.test.Student 객체로 쿼리 실행 결과값을 받고자 할 때 <select id="selectTest" resultType="com.test.Student"> </select> - 예: int 형 객체로 쿼리 실행 결과값을 받을 때 <select id="selectTest" resultType="int"> </select> resultMap 답안첨삭 - resultMap 선언 당시 참조로 사용한 이름을 입력 - resultType을 이용하면 자동 매핑되기 때문에 편리하지만 제한이 있으나, resultMap을 사용하면 개발자가 직접 원하는 POJO 클래스에 매핑 가능 -예: <resultMap id="test" type="com.test.Student"> <result property="name" column="name"> </resultMap> <select id="selectTest" resultMap="test"> </select> mybatis 구현을 위한 기능 설정에 필요한 속성을 정확하게 설명하였다. resultType: resultset 결과를 매핑해서 반환되는 타입의 전체 클래스명이나 별명 모범답안 resultMap: resultset 결과를 매핑할 resultMap 참조 아이디명 평가기준 mybatis 구현을 하기 위해 정의된 기능을 식별할 수 있는지 평가한다. 수행준거 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계가 필요한 인터페이스의 기능을 식별할 수 있다. 2. (scott/tiger 계정)DEPT 테이블의 특정 row의 dname과 loc를 수정하는 기능을 작성 할 때, mapper xml 에 알맞은 태그를 작성하시오. (15점) 앨리먼트 아이디: updateDept 매개변수자료형 별칭: deptDto 사용할 테이블명 : DEPT 객체의 필드명과 테이블의 컬럼명은 동일한 COMMAND 객체임. deptDto의 field∶deptno, dname, loc 수정할 기준이 되는 column: DEPTNO 결과 점수 (15.0점) 앨리먼트 아이디 : updateDept 매개변수자료형 별칭 : deptDto 사용할 테이블명: DEPT 객체의 필드명과 테이블의 컬럼명은 동일한 COMMAND 객체임. deptDto의 field: deptno, dname, loc 수강생 답 수정할 기준이 되는 column: DEPTNO <update id="updateDept" resultType="deptDto" > update dept set dname = #{dname}, $loc = \#\{loc\}$ where deptno = #{deptno} </update> 앨리먼트 아이디 : updateDept 매개변수자료형 별칭 : deptDto 사용할 테이블명: DEPT 객체의 필드명과 테이블의 컬럼명은 동일한 COMMAND 객체임. deptDto의 field∶deptno, dname, loc 수정할 기준이 되는 column: DEPTNO 답안첨삭 <update id="updateDept" resultType="deptDto" > dept set dname = #{dname}, $loc = \#\{loc\}$ where deptno = #{deptno} </update> mybatis 설계서를 확인하여 update element를 정확히 기술하였다. <update id="selectDept" parameterType="deptDto"> 모범답안 UPDATE DEPT SET DNAME=#{dname}, LOC=#{loc} WHERE DEPTNO=#{deptno} </update> 평가기준 update element를 정확히 작성할 수 있는지 평가한다. 수행준거 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 세부 설계서를 확인하여 공통적인 인터페이스를 구현할 수 있다. 3. Servlet을 이용한 웹 프로젝트에서 Mybatis를 사용하기 위한 순서를 기술하시오. (15점) 결과 점수 (15.0점) 1. MyBatis 라이브러리 준비 https://github.com/mybatis 에 접속한다. Pinned repositories에서 mybatis-3을 클릭한다. README.md에서 Download Latest를 클릭한다. Assets에서 mybatis-3.5.4.zip을 클릭한다. 압축을 풀면 JAR 파일과 사용 설명 PDF 파일, 의존 라이브러리 JAR파일이 있다. mybatis-3.5.4.jar 파일과 의존 라이브러리 파일들을 웹 프로젝트에 추가한다. 2. DAO 변경 MyBatis 프레임워크를 사용함으로써 이제 DB 연동은 MyBatis가 대신한다. JDBC 프로그래밍 대신, MyBatis의 SqlSessionFactory 컴포넌트를 사용하도록 변경하고 DAO의 SQL문을 별도의 파일로 분리할 차례이다. SqlSessionFactory 우선 DAO 클래스에 SqlSessionFactory 인스턴스 변수와 setter를 선언한다. MyBatis에서는 SqlSession 객체가 실제 SQL을 실행하는데, SqlSession 객체는 SqlSessionFactory로부터 얻을 수 있다. DAO가 필요로하는 SqlSessionFactory는 외부에서 주입될 수 있도록 setter를 선언하는 것이다. 먼저 MyBatis를 이용한 select 구현 방법을 알아보자. SqlSessionFactory 객체의 openSession()을 호출해서 SqlSession을 얻는다. SqlSession은 MyBatis에서 실제 SQL 실행을 담당하는 컴포넌트이다. 즉 SQL을 실행하려면 SqlSession이 필요하며, 이 객체가 JDBC 드라이버를 사용한다. SqlSession 객체의 selectList()를 호출해서 select문을 실행할 수 있다. 파라미터로는 SQL mapper의 네임 스페이스와 SQL문 id를 합친 문자열을 넘긴다. 위 코드에서 com.atoz_develop.spms.dao.ProjectDao가 네임 스페이스, selectList가 SQL문 id에 해당한 마지막으로 finally 블록에서 SqlSession 객체의 close()를 호출해서 SQL문을 실행할 때 사용한 자원을 해제 해준다. 한 개 데이터를 select 해울 때는 selectOne() 메소드를 호출한다. selectList()와 달리 두 번째 파라미터에 조건절에서 사용할 데이터를 넘긴다. 이 두 번째 파라미터의 타입은 Object, 즉 객체를 넘겨야 하지만 int와 같은 기본 데이터 타입을 넘겨도 컴파일 시 auto boxing이 수행돼서 변환 없이 그대로 넘겨도 된다. 3) Insert Insert문은 SqlSession 객체의 insert()를 호출해서 실행할 수 있다. selectList()와 마찬가지로 첫 번째 파라미터는 SQL mapper의 네임 스페이스와 SQL id를 합친 문자열을 넘 긴다. 두 번째 파라미터로는 insert문을 실행할 때 필요한 프로퍼티를 가진 VO 객체를 넘긴다. insert() 호출 후 commit()을 호출한다. SqlSession 객체를 얻을 때 openSession(true)와 같이 호출하면 INSERT, UPDATE, DELETE문 실행 시 수강생 답 auto commit을 수행하는 SqlSession 객체를 얻을 수 있다. 4) Update UPDATE문은 SqlSession 객체의 update()를 호출해서 실행할 수 있다. 문법은 이전과 동일하므로 생략한다. 5) Delete DELETE문은 SqlSession 객체의 delete()를 호출해서 실행할 수 있다. 문법은 이전과 동일하므로 생략한다. 이렇게 SQL문 명령어와 SqlSession 객체의 메소드명이 거의 동일해서 호출 시 파라미터만 유의해서 넘겨주 면 크게 어려움 없이 DAO를 변경할 수 있다. 3. SQL Mapper 파일 작성 SqlSession이 SQL문을 실행할때 참조하는 SQL mapper 파일을 작성한다. 즉 원래 DAO 클래스에 내장돼있던 SQL문을 이 SQL mapper 파일로 분리해주는 것이다. DAO 패키지에 xml 파일을 생성하고 아래와 같이 작성한다. 필요한 resultMap과 SQL문은 각자 프로젝트에 맞게 변경한다. SQL mapper 파일에서 <mapper>의 namespace값 과 <select>, <insert>, <update>, <delete>의 id 값은 SqlSession 객체가 SQL문을 찾을 때 사용한다. DAO 클래스에서 SqlSession의 selectList(), selectOne(), insert(), update(), delete() 메소드를 호출할 때 넘 기는 문자열 값과 매핑되는 것이다. 4. MyBatis 설정 파일 작성 MyBatis 설정 파일에는 DB 커넥션을 생성하는 data source에 대한 정보, 트랜잭션 관리자, MyBatis 동작을 제어하는 환경값, SQL mapper 파일 경로 등을 설정한다. DAO 패키지에 xml 파일을 생성하고 다음과 같이 작성한다. 상세한 설정값은 각자 프로젝트에 맞게 변경한다. 5, 프로퍼티 파일 작성 MyBatis 설정 파일의 roperties resource="...">에 지정한 프로퍼티 파일을 작성한다. 이 파일은 DB 접속 정보를 담는다. 6. SqlSessionFactory 객체 생성 DAO에서 SqlSessionFactory의 setter를 선언해 주었다. 외부에서 SqlSessionFactory 객체를 생성하고 주입해주어야 한다. SqlSessionFactory는 SqlSessionFactoryBuilder 객체를 통해 생성할 수 있는데 build()를 호출해서 위에서 작성한 MyBatis 설정 파일의 입력 스트림을 넘겨주어야 한다. MyBatis 설정 파일은 보통 CLASSPATH에 위치시키는데, CLASSPATH에 있는 파일의 입력스트림은 Resources 클래스의 getResourceAsStream()을 이용해서 손쉽게 얻을 수 있다. 이렇게 생성한 sqlSessionFactory를 DAO에 주입시켜주면 MyBatis 사용에 필요한 모든 구현이 완료된다. 1. MyBatis 라이브러리 준비 https://github.com/mybatis 에 접속한다. Pinned repositories에서 mybatis-3을 클릭한다. README.md에서 Download Latest를 클릭한다. Assets에서 mybatis-3.5.4.zip을 클릭한다. 압축을 풀면 JAR 파일과 사용 설명 PDF 파일, 의존 라이브러리 JAR파일이 있다. mybatis-3.5.4.jar 파일과 의존 라이브러리 파일들을 웹 프로젝트에 추가한다. 2. DAO 변경 MyBatis 프레임워크를 사용함으로써 이제 DB 연동은 MyBatis가 대신한다. JDBC 프로그래밍 대신, MyBatis의 SqlSessionFactory 컴포넌트를 사용하도록 변경하고 DAO의 SQL문을 별도의 파일로 분리할 차례이다. SqlSessionFactory 우선 DAO 클래스에 SqlSessionFactory 인스턴스 변수와 setter를 선언한다. MyBatis에서는 SqlSession 객체가 실제 SQL을 실행하는데, SqlSession 객체는 SqlSessionFactory로부터 얻을 수 있다. DAO가 필요로하는 SqlSessionFactory는 외부에서 주입될 수 있도록 setter를 선언하는 것이다. 먼저 MyBatis를 이용한 select 구현 방법을 알아보자. SqlSessionFactory 객체의 openSession()을 호출해서 SqlSession을 얻는다. SqlSession은 MyBatis에서 실제 SQL 실행을 담당하는 컴포넌트이다. 즉 SQL을 실행하려면 SqlSession이 필요하며, 이 객체가 JDBC 드라이버를 사용한다. SqlSession 객체의 selectList()를 호출해서 select문을 실행할 수 있다. 파라미터로는 SQL mapper의 네임 스페이스와 SQL문 id를 합친 문자열을 넘긴다. 위 코드에서 com.atoz_develop.spms.dao.ProjectDao가 네임 스페이스, selectList가 SQL문 id에 해당한 마지막으로 finally 블록에서 SqlSession 객체의 close()를 호출해서 SQL문을 실행할 때 사용한 자원을 해제 해준다. 한 개 데이터를 select 해울 때는 selectOne() 메소드를 호출한다. selectList()와 달리 두 번째 파라미터에 조건절에서 사용할 데이터를 넘긴다. 이 두 번째 파라미터의 타입은 Object, 즉 객체를 넘겨야 하지만 int와 같은 기본 데이터 타입을 넘겨도 컴파일 시 auto boxing이 수행돼서 변환 없이 그대로 넘겨도 된다. 3) Insert Insert문은 SqlSession 객체의 insert()를 호출해서 실행할 수 있다. selectList()와 마찬가지로 첫 번째 파라미터는 SQL mapper의 네임 스페이스와 SQL id를 합친 문자열을 넘 두 번째 파라미터로는 insert문을 실행할 때 필요한 프로퍼티를 가진 VO 객체를 넘긴다. insert() 호출 후 commit()을 호출한다. SqlSession 객체를 얻을 때 openSession(true)와 같이 호출하면 INSERT, UPDATE, DELETE문 실행 시 답안첨삭 auto commit을 수행하는 SqlSession 객체를 얻을 수 있다. 4) Update UPDATE문은 SqlSession 객체의 update()를 호출해서 실행할 수 있다. 문법은 이전과 동일하므로 생략한다. 5) Delete DELETE문은 SqlSession 객체의 delete()를 호출해서 실행할 수 있다. 문법은 이전과 동일하므로 생략한다. 이렇게 SQL문 명령어와 SqlSession 객체의 메소드명이 거의 동일해서 호출 시 파라미터만 유의해서 넘겨주 면 크게 어려움 없이 DAO를 변경할 수 있다. 3. SQL Mapper 파일 작성 SqlSession이 SQL문을 실행할때 참조하는 SQL mapper 파일을 작성한다. 즉 원래 DAO 클래스에 내장돼있던 SQL문을 이 SQL mapper 파일로 분리해주는 것이다. DAO 패키지에 xml 파일을 생성하고 아래와 같이 작성한다. 필요한 resultMap과 SQL문은 각자 프로젝트에 맞게 변경한다. SQL mapper 파일에서 <mapper>의 namespace값 과 <select>, <insert>, <update>, <delete>의 id 값은 SqlSession 객체가 SQL문을 찾을 때 사용한다. DAO 클래스에서 SqlSession의 selectList(), selectOne(), insert(), update(), delete() 메소드를 호출할 때 넘 기는 문자열 값과 매핑되는 것이다. 4. MyBatis 설정 파일 작성 MyBatis 설정 파일에는 DB 커넥션을 생성하는 data source에 대한 정보, 트랜잭션 관리자, MyBatis 동작을 제어하는 환경값, SQL mapper 파일 경로 등을 설정한다. DAO 패키지에 xml 파일을 생성하고 다음과 같이 작성한다. 상세한 설정값은 각자 프로젝트에 맞게 변경한다. 5. 프로퍼티 파일 작성 MyBatis 설정 파일의 roperties resource="...">에 지정한 프로퍼티 파일을 작성한다. 이 파일은 DB 접속 정보를 담는다. 6. SqlSessionFactory 객체 생성 DAO에서 SqlSessionFactory의 setter를 선언해 주었다. 외부에서 SqlSessionFactory 객체를 생성하고 주입해주어야 한다. SqlSessionFactory는 SqlSessionFactoryBuilder 객체를 통해 생성할 수 있는데 build()를 호출해서 위에서 작성한 MyBatis 설정 파일의 입력 스트림을 넘겨주어야 한다. MyBatis 설정 파일은 보통 CLASSPATH에 위치시키는데, CLASSPATH에 있는 파일의 입력스트림은 Resources 클래스의 getResourceAsStream()을 이용해서 손쉽게 얻을 수 있다. 이렇게 생성한 sqlSessionFactory를 DAO에 주입시켜주면 MyBatis 사용에 필요한 모든 구현이 완료된다. mybatis를 확인하여 정형화된 인터페이스 기능 구현을 정확하게 정의할 수 있다 1) ojdbc6.jar를 lib 폴더에 추가 2) mybatis-x.x.x.jar를 다운로드 후 lib 폴더에 추가 모범답안 3) config.xml 작성 4) mapper.xml 작성 5) dao 작성 평가기준 mybatis를 확인하여 정형화된 인터페이스 기능 구현을 이해하는지 평가한다. 수행준거 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 세부 설계서를 확인하여 공통적인 인터페이스를 구현할 수 있다. 4. MyBatis가 제공하는 동적 SQL 구문 중에서 자바의 if문과 같은 역할을 수행하는 if 구문을 사용하여 아래 명시된 내용을 처리하는 select 앨리먼트를 완성하시오. (25점) Select 앨리먼트 아이디: findActiveBlogWithTitleLike 반환자료형과 매개변수자료형 별칭 : Blog 조회할 테이블명 : blog의 모든 컬럼 조회 객체의 필드명과 테이블의 컬럼명은 동일한 COMMAND 객체임. 조건절의 공통조건 : state 컬럼값이 'ACTIVE' 추가조건: title 컬럼값이 널이 아니면 title 컬럼값 like 비교값 결과 점수 (25.0점) <select id="findActiveBlogWithTitleLike" resultType="Blog"> select from blog 수강생 답 where state = 'ACTIVE' <if test="titel != null"> title like '%' || #{title} || '%' </if> </select> <select id="findActiveBlogWithTitleLike" resultType="Blog"> select from blog where 답안첨삭 state = 'ACTIVE' <if test="titel != null"> title like '%' || #{title} || '%' </if> </select> mybatis 설계서를 확인하여 select element를 정확히 기술하였다. <select id="findActiveBlogWithTitleLike" parameterType="Blog" resultType="Blog"> SELECT * FROM BLOG WHERE state = 'ACTIVE' 모범답안 <if test="title != null"> AND title like #{title} </if> </select> 평가기준 select element를 정확히 이해하고 사용할 수 있는지 평가한다. 수행준거 인터페이스 구현 검증을 위하여 외부 시스템과의 연계 모듈 상태를 확인할 수 있다. 5. mapper.xml 에 작성하는 namespace의 역할을 설명하시오. (10점) 결과 점수 (10.0점) namespace 는 구분하는 식별자 이며 클래스에서는 패키지 와 같은역할로 한다. 수강생 답 Mybatis 내에서 원하는 SQL문 찾아서 실행할때 동작을 한다. namespace속성은 이 mapper.xml과 매핑될 mapper.java(인터페이스)를 의미한다. namespace 는 구분하는 식별자 이며 클래스에서는 패키지 와 같은역할로 한다. Mybatis 내에서 원하는 SQL문 찾아서 실행할때 동작을 한다. 답안첨삭 namespace속성은 이 mapper.xml과 매핑될 mapper.java(인터페이스)를 의미한다. mybatis 구현을 위해 내부 모듈 간의 연계에 필요한 namespace 에 대해 정확하게 설명하였다. 패키지경로를 포함한 전체 이름을 가진 구문을 구분하기 위해 필수로 사용해야 한다. 모범답안 인터페이스 바인딩을 가능하게 한다. 평가기준 mapper의 namespace를 이해하고 있는지 평가한다 수행준거 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계가 필요한 인터페이스의 기능을 식별할 수 있다. 6. config.xml에서 domain.blog.Comment 클래스를 Comment 라는 별칭으로 사용하려고 할 때, 필요한 앨리먼트와 해당 앨리먼트의 속성에 값을 넣어서 작성하시오. (25점) 결과 점수 (25.0점) <typeAliases> <typeAlias type="domain.blog.Comment" alias="Comment"/> 수강생 답 </typeAliases> <typeAliases> <typeAlias type="domain.blog.Comment" alias="Comment"/> 답안첨삭 </typeAliases> mybatis 명세서를 참조하여 구현 검증에 필요한 typeAliases 앨리먼트를 정확하게 작성할 수 있다 <typeAliases> 모범답안 <typeAlias alias="Commant" type="domain.blog.Comment"/> </typeAliases> typeAliases 앨리먼트를 정확히 작성할 수 있는지 평가한다. 평가기준 수행준거 인터페이스 구현 검증을 위하여 외부 시스템과의 연계 모듈 상태를 확인할 수 있다. 평가 메인으로

KH 정보교육원

국비대상자 간편조회 | 국비무료교육 온라인상담