수강생 평가

\* 첨부파일의 확장자를 소문자로 등록하세요. - 예시: test.jpg(O), test.JPG(X)

커뮤니티

산학연계

상담센터

마이페이지

채용정보

취업지원센터

[강남 M] 2020. 12. 02 ~ 2021. 07. 08 15:30~22:00 김동현 강사 | 진은지 취업담임 남은 시간 11:38:22

교육원소개

스마트 콘텐츠 융합 응용SW 엔지니어 양성과정(888H)

```
[NCS전공교과] 통합 구현 (평가자체크리스트)
                                                                                                                         1회차 ~
                                                           총점: 40.0
명시된 요구사항의 내용에 따라 오라클 데이터베이스와 MyBatis 연동 처리 및 스프링 웹 애플리케이션과의 연계 모듈 작성에 대한 내용들을 작성하시오.
아래의 그림을 참조하여 mybatis-config.xml 파일 안에 작성되는 앨리먼트중 클래스패스의 상대경로로 리소스를 지정하는 mappers 엘리먼트를 작성하시오 벌칭은 클래스명과 동일함.

→ B org.mybatis.builder

      AuthoMapper.xml
      BlogMapper.xml
      PostMapper.xml
요구사항
               mappers 앨리먼트 안에 Type 들을 등록하시오.
               <!-- org.mybatis.builder 하위의 모든 클래스를 별칭등록 ->
수강생 답
                  <package name="org.mybatis.builder"/>
                </typeAliases>
               <!- org.mybatis.builder 하위의 모든 클래스를 별칭등록 ->
                <typeAliases>
                  <package name="org.mybatis.builder"/>
답안첨삭
                </typeAliases>
               ⇒ typeAlias 아니고, mapper입니다 ㅠ
               <mappers>
               <mapper resource="org/mybatis/builder/AuthorMapper.xml"/>
모범답안
               <mapper resource="org/mybatis/builder/BlogMapper.xml"/>
               <mapper resource="org/mybatis/builder/PostMapper.xml"/>
               </mappers>
(scott/tiger) DEPT table의 모든 데이터를 return하는 메소드가 다음과 같을 때, mybatis를 사용하여 같은 기능을 할 수 있도록 스프링 MVC 구조의 웹애플리케이션에서의 DeptDao 클래
스의 selectAll 메소드를 기술하고 deptMapper.xml 파일 안에 작성할 내용들도 작성하시오.
조건
1) mapper.xml♀l namespace : deptmapper
2) mapper.xml select 엘리먼트의 id: selectList
3) vo Type 별칭: DeptDto
4) SqlSessionFactory 객체는 전역변수로 선언
되어 있으며, getSqlSessionFactory() 라는 메소드로
객체를 생성한다.
public List<DeptDto> selectList(){
    List<DeptDto> list = new ArrayList<DeptDto>();
    Connection con = getConnection();
    PreparedStatement pstm = null;
    ResultSet rs = null;
    String sql = "SELECT DEPTNO, DNAME, LOC FROM DEPT";
    try {
        pstm = con.prepareStatement(sql);
        rs = pstm.executeQuery();
        while(rs.next()) {
           DeptDto dto = new DeptDto();
           dto.setDeptno(rs.getInt("DEPTNO"));
           dto.setDname(rs.getString("DNAME"));
           dto.setLoc(rs.getString("LOC"));
           list.add(dto);
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        close(rs);
        close(pstm);
        close(con);
    return list;
요구사항
              Dao 의 마이바티스를 사용하는 코드로 selectAll() 메소드 코드를 수정 작성하고, mapper.xml 에 기록할 내용도 작성하시오.
```

```
<mapper namespace="deptmapper">
                       <select id="selectAll" resultType="DeptDto">
수강생 답
                       SELEECT DEPTNO, DNAME, LOC FROM DEPT
                       </select>
                 </mapper>
                 <mapper namespace="deptmapper">
                       <select id="selectAll" resultType="DeptDto">
                       SELEECT DEPTNO, DNAME, LOC FROM DEPT
답안첨삭
                       </select>
                 </mapper>
                 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 데이터 연계 요구사항을 고려하여 마이바티
                스 Mapper 의 내용을 일부만 기술하였음
                // DeptDao.java
                public List<DeptDto> selectAll() {
                SqlSession session = null;
                List<Dept> list = null;
                try {
                 session = getSqlSessionFactory().openSession(false);
                list = session.selectList("deptmapper.selectList");
                } catch (Exception e) {
                e.printStackTrace();
                } finally {
모범답안
                session.close();
                return list;
                // mapper
                <mapper namespace="deptmapper">
                <select id="selectList" resultType="DeptDto">
                 SELECT DEPTNO, DNAME, LOC FROM DEPT
                 </select>
                </mapper>
스프링에서 파일을 업로드 처리 하려고 할 때의 절차이다. 1, 2번 항목의 요구사항에 들어갈 것들을 작성하시오.
<!- 파일 업로드 설정 -->
<beans:bean id="multipartResolver"</pre>
class="org.springframework.web.multipart.commons.CommonsMultipartResolver">
<beans:property name="maxUploadSize" value="100000" />
```

```
전송온 이름: photo, 필수항목임.
2. Service 인터페이스에 추가: public int writeFileName(String filename)
요구사항
                1, 2 번 항목에 대한 내용을 작성하시오.
                1번
                @GetMapping(
                          value = "/fileDownload.do",
                          produces = MediaType.APPLICATION\_OCTET\_STREAM\_VALUE
```

1. 컨트롤러에서 파일업로드 처리 : 메소드명 - public void fileUpMethod

```
@ResponseBody // 응답메세지에 return객체를 직접 출력
                    public void fileUpMethod(@RequestParam int photo, HttpServletResponse response) throws UnsupportedEncodingException {
                            //1. 업무로직 : db에서 첨부파일 정보 조회
                           Attachment attach = boardService.writeFileName(photo);
                            log.debug("attach = {}", attach);
                            if(attach = null) {
                               throw new IllegalArgumentException("해당 첨부파일은 존재하지 않습니다:" + photo);
                            //2. Resource객체를 리턴 : 응답메세지에서 출력은 spring-container가 처리
                            String originalFilename = attach.getOriginalFilename();
                            String renamedFilename = attach.getRenamedFilename();
                            String saveDirectory = application.getRealPath("/resources/upload/board");
                            File downFile = new File(saveDirectory, renamedFilename);
수강생 답
                           //웹상자원, 서버컴퓨터자원을 모두 다룰수 있는 스프링의 추상화 layer
                            String location = "file:" + downFile.toString();
                            log.debug("location = {}", location);
                            //3.응답헤더
                            //한글깨짐방지처리
                           originalFilename = new String(originalFilename.getBytes("utf-8"), "iso-8859-1");
                           response.setContentType(MediaType.APPLICATION_OCTET_STREAM_VALUE);
                           response.addHeader(HttpHeaders.CONTENT_DISPOSITION, "attachment; filename=" + originalFilename);
                        } catch(Exception e) {
                           log.error("파일 다운로드 오류!", e);
                            throw e;
                 2번
                 public int writeFileName(String filename) {
                        return boardDao. writeFileName(filename);
                 1번
                 @GetMapping(
                           value = "/fileDownload.do",
```

```
produces = MediaType.APPLICATION_OCTET_STREAM_VALUE
                    @ResponseBody // 응답메세지에 return객체를 직접 출력
                    public void fileUpMethod(@RequestParam int photo, HttpServletResponse response) throws UnsupportedEncodingException {
                           //1. 업무로직: db에서 첨부파일 정보 조회
                           Attachment attach = boardService.writeFileName(photo);
                           log.debug("attach = {}", attach);
                            if(attach = null) {
                               throw new IllegalArgumentException("해당 첨부파일은 존재하지 않습니다:" + photo);
                           //2. Resource객체를 리턴 : 응답메세지에서 출력은 spring-container가 처리
                           String originalFilename = attach.getOriginalFilename();
                           String renamedFilename = attach.getRenamedFilename();
                           String saveDirectory = application.getRealPath("/resources/upload/board");
                           File downFile = new File(saveDirectory, renamedFilename);
                           //웹상자원, 서버컴퓨터자원을 모두 다룰수 있는 스프링의 추상화 layer
답안첨삭
                           String location = "file:" + downFile.toString();
                           log.debug("location = {}", location);
                           //3.응답헤더
                           //한글깨짐방지처리
                           originalFilename = new String(originalFilename.getBytes("utf-8"), "iso-8859-1");
                           response.setContentType(MediaType.APPLICATION_OCTET_STREAM_VALUE);
                           response.addHeader(HttpHeaders.CONTENT_DISPOSITION, "attachment; filename=" + originalFilename);
                       } catch(Exception e) {
                           log.error("파일 다운로드 오류!", e);
                           throw e;
                ⇒ 파일업로드를 처리할 수 있는 controller의 handler를 작성해야합니다.
                 2번
                 public int writeFileName(String filename) {
                        return boardDao. writeFileName(filename);
                 //controller
                 public void fileUpMethod(@RequestParam("photo", required="true") MultipartFile mfile){
                 int\ result\ service.write Fie Name (mfile.get Original Filename ());
모범답안
                 public int writeFileName(String filename);
                                                                      평가항목
```

평가내용	평가기준	배점	평가결과
연계 요구사항 분석	개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 데이터 연계 요구사항을 분석할 수 있다.	10.0	예
연계 필요 데이터 식별	개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계가 필요한 데이터를 식별할 수 있다.	10.0	예
연계 표준 설계	개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계를 위한 데이터 표준을 설계할 수 있다.	10.0	예
연계 대상 데이터 송수신 방법 정의	개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 특성을 고려하여 효율적 데이터 송수신 방법을 정의할 수 있다.	10.0	예
연계 모듈간 연계주기 정의	개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 데이터 연계 요구사항을 고려하여 연계주기를 정의할 수 있다.	10.0	아니오
연계 실패 시 처리방안 정의	개발하고자하는 응용소프트웨어와 연계 대상 내외부 모듈 간의 연계 목적을 고려하여 데이터 연계 실패 시 처리방안을 정의할 수 있다.	10.0	아니오
연계 모듈간 보안 적용	응용소프트웨어와 관련된 내외부 모듈 간의 연계 데이터의 중요성을 고려하여 송수신 시 보안을 적용할 수 있다.	10.0	아니오
연계모듈 구현환경 준비	구성된 연계 매카니즘에 대한 명세0서를 참조하여 연계모듈구현을 위한 논리적, 물리적 환경을 준비할 수 있다.	10.0	아니오
연계 모듈 구현	구성된 연계 매카니즘에 대한 명세서를 참조하여 외부 시스템과의 연계 모듈을 구현할 수 있다.	10.0	아니오
연계 데이터 무결성 검증	연계모듈의 안정적인 작동여부와 모듈 간 인터페이스를 통해 연동된 데이터와 무결성을 검증할 수 있다.	10.0	아니오