

■■■ Spring MVC ■■■

■■■ Spring MVC 프로젝트 실습04 ■■■

- Perspective
 - JavaEE
- 다이나믹 웹 프로젝트 생성
 - mvcApp04
 - (STS - 스프링 / 스프링 레거시 / 스프링 부트 / 스프링 MVC ...)
- **mvcApp00** 으로 부터 **src, WebContent** 디렉터리 복사&붙여넣기로 대체
 - 톰캣 서버 등록 및 포트(port)설정
 - 기타 환경 설정,
 - Spring MVC 환경 설정을 위한 기본적인 라이브러리 등록 및 설정,
 - dispatcher-servlet.xml 파일 구성
 - web.xml 파일 구성
- 톰캣 서버 등록 및 포트(port)설정
 - port 8080 → 8090 → 3306 / URIEncoding="UTF-8" → **server.xml**
 - Tomcat_server > server.xml (#63 ~ 67)

```
<Connector URIEncoding="UTF-8" connectionTimeout="20000" maxParameterCount="1000" port="3306"/>
```
- 기타 환경 설정
 - 폰트, 줄번호, 편집 포맷, 인코딩 방식, 템플릿 등... → **epf 파일 import**
- 실습 성격
 - 데이터 송수신 관련 실습(Spring MVC 버전)
 - JDBC 모듈 추가 활용
 - SimpleDriverDataSource 객체 활용
- 기능 구현(데이터 각 유형별 CRUD 실습)
 - 직원 정보 출력, 입력, 수정, 삭제 기능 구현.
 - 지역 정보 출력, 입력, 수정, 삭제 기능 구현.
 - 부서 정보 출력, 입력, 수정, 삭제 기능 구현.
 - 직위 정보 출력, 입력, 수정, 삭제 기능 구현.
 - 로그인 / 로그아웃 기능 구현
 - 관리자 및 일반 직원 로그인 구분.
 - ==> Spring MVC, JSP/Servlet, Oracle, HTML/CSS/Javascript, AJAX, jQuery, EL, JSTL 표현 등 사용.
- 사전 작업
 - → 테이블 및 시퀀스, 주요 쿼리문, 샘플 데이터 등 구성
 - EMPLOYEE → 직원
 - REGION → 지역
 - DEPARTMEN → 부서
 - POSITION → 직위
 - 샘플 데이터 → 입력...

- 주요 쿼리문 구성
- ○ Spring MVC 환경 설정을 위한 기본적인 라이브러리 등록(14개 파일)
 - **WebContent > WEB-INF > lib** 경로에 등록
 - 경로1 → **C:\spring-framework-3.0.2-with-docs\dist**
 - 파일1 → **org.springframework.aop-3.0.2.RELEASE.jar** → aop
 - 파일2 → **org.springframework.asm-3.0.2.RELEASE.jar** → asm
 - 파일3 → **org.springframework.beans-3.0.2.RELEASE.jar** → beans
 - 파일4 → **org.springframework.context-3.0.2.RELEASE.jar** → context
 - 파일5 → **org.springframework.core-3.0.2.RELEASE.jar** → core
 - 파일6 → **org.springframework.expression-3.0.2.RELEASE.jar** → expression
 - 파일7 → **org.springframework.jdbc-3.0.2.RELEASE.jar** → jdbc
 - 파일8 → **org.springframework.transaction-3.0.2.RELEASE.jar** → trasaction
 - 파일9 → **org.springframework.web.servlet-3.0.2.RELEASE.jar** → servlet
 - 파일10 → **org.springframework.web-3.0.2.RELEASE.jar** → web
 - 경로2 → **C:\spring-framework-3.0.2-dependencies\org.apache.commons\com.springsource.org.apache.commons.logging\1.1.1**
 - 파일11 → **com.springsource.org.apache.commons.logging-1.1.1.jar** → logging
 - 경로3 → **C:\oraclexe\app\oracle\product\11.2.0\server\jdbc\lib** (오라클 설치 경로)
 - 파일12 → **ojdbc6.jar** → ojdbc6
 - 경로4 → **C:\Downloads\13_JSTL\jakarta-taglibs-standard-1.1.2\lib**
 - 파일13 → **jstl.jar** → jstl
 - 파일14 → **standard.jar** → standard

※ WebContent > WEB-INF > lib 경로에 xml 파일 생성

- → 『dispatcher-servlet.xml』
- 『dispatcher-servlet.xml』 파일을 WebContent > WEB-INF 경로에 생성한다
- 그리고 이 파일의 기본 설정 내용(템플릿)을 구성하기 위해
- 스프링 DOC의 내용을 참고해서 작성할 수 있도록 한다.
 - 경로 → **C:\spring-framework-3.0.2-with-docs\docs\spring-framework-reference\pdf**
 - 파일 → **spring-framework-reference.pdf**
 - 내용(위치) → 문서 구조성 425 페이지 (실제 내용 상 450 페이지)
 - 해당 위치로 이동하면 XML 파일에 대한 샘플 코드가 있다.
 - 이 내용을 복사하여 dispatcher-servelet.xml 파일의 기본 템플릿으로 작성한다.
 - 내용(위치)
 - V.The Web → P.441
 - 15.Web MVC framework → P.442
 - 15.3 Implementing Controllers → P.449
 - Defining a controller with @Controller → P.450

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:p="http://www.springframework.org/schema/p"
xmlns:context="http://www.springframework.org/schema/context"
xsi:schemaLocation="
http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.0.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context-3.0.xsd">
<context:component-scan base-package="org.springframework.samples.petclinic.web"/>
// ...
</beans>
```

○ 물리적 파일 구성

- #00. mvcApp04_scott.sql
 - → 데이터베이스 관련 객체 구성 및 실습 데이터 준비
- #01. Employee.java
 - DTO 개념 활용 (→ DTO 계층)
 - 직원 데이터 자료형 클래스.
- #02. Department.java
 - DTO 개념 활용 (→ DTO 계층)
 - 부서 데이터 자료형 클래스.
- #03. Region.java
 - DTO 개념 활용 (→ DTO 계층)
 - 지역 데이터 자료형 클래스.
- #04. Position.java
 - DTO 개념 활용 (→ DTO 계층)
 - 직위 데이터 자료형 클래스.
- #05. IEmployeeDAO.java
 - → 인터페이스.
- #06. IDepartmentDAO.java
 - → 인터페이스.
- #07. IRegionDAO.java
 - → 인터페이스.
- #08. IPositionDAO.java
 - → 인터페이스.
- #09 EmployeeDAO.java
 - → 데이터베이스 액션 처리 클래스
 - 직원 데이터 CRUD 액션(직원 데이터 출력, 입력, 수정, 삭제 액션)
 - Connection 객체에 대한 의존성 주입 준비
 - (인터페이스 속성 구성 / setter 구성)
- #10 DepartmentDAO.java
 - → 데이터베이스 액션 처리 클래스
 - 부서 데이터 CRUD 액션
 - Connection 객체에 대한 의존성 주입 준비
 - (인터페이스 속성 구성 / setter 구성)
- #11. RegionDAO.java
 - → 데이터베이스 액션 처리 클래스
 - 지역 데이터 CRUD 액션
 - Connection 객체에 대한 의존성 주입 준비
 - (인터페이스 속성 구성 / setter 구성)

- #12. PositionDAO.java
 - → 데이터베이스 액션 처리 클래스
 - 직위 데이터 CRUD 액션
 - Connection 객체에 대한 의존성 주입 준비
 - (인터페이스 속성 구성 / setter 구성)
- #13. EmployeeMenu.jsp
 - → 메인 메뉴 뷰 페이지.
 - 직원 관리, 부서 관리, 지역 관리, 직위 관리
 - 로그아웃 기능 포함.
- #14. menuStyle.css
 - → 메뉴 스타일 구성.
- #15. EmployeeList.jsp
 - → 직원 리스트 출력 페이지.
- #16. main.css
 - → 메인 스타일 구성.
- #17. EmployeeListController.java
 - → 사용자 정의 컨트롤러 클래스
 - 직원 리스트 컨트롤러
 - 리스트 페이지 요청에 대한 액션 처리
 - (employeeList.action) 리스트 출력 액션
 - (의존성 주입을 위한 준비 필요)
 - DAO 객체에 대한 의존성 주입(DI)을 위한 준비
 - 인터페이스 형태의 자료형 속성으로 구성
 - setter 구성
- #17. IMemberDAO.java → 인터페이스
 - MemberDAO.java
 - → 데이터베이스 액션 처리 클래스
 - Connection 객체에 대한 의존성 주입
 - (의존성 주입을 위한 준비 필요)
- #18. web.xml
 - → DispatcherServlet 객체 등록
 - URL 매핑 주소 등록
- #19. dispatcher-servlet.xml
 - → 컨트롤러 객체 등록
 - URL 매핑 주소 등록
 - SimpleDriverDataSource 객체 등록
 - 의존 객체 주입 설정
 - DAO 객체 등록
- #20. EmployeeInsertFormController.java
 - (employeeinsertform.action)
 - → 직원 데이터 입력 폼 컨트롤러
 -

- #21. EmployeeInsertForm.jsp
 - 직원 데이터 입력 폼 뷰 페이지
- #22. AjaxController.java
 - (ajax.action)
 - AJAX 요청 컨트롤러
- #23. Ajax.jsp
- #24. EmployeeInsertController.java
 - (employeeinsert.action)
 - 직원 데이터 입력 액션 컨트롤러
- #25. EmployeeUpdateForm.jsp
 - 직원 데이터 수정 폼 뷰 페이지
- #26. EmployeeUpdateFormController.java
 - (employeeupdateform.action)
 - 직원 데이터 수정 폼 컨트롤러
- #27. EmployeeUpdateController.java
 - (employeeupdate.action)
 - 직원 데이터 수정 액션 컨트롤러
- #28. EmployeeDeleteController.java
 - (employeedelete.action)
 - 직원 데이터 삭제 액션 컨트롤러
- #29. LoginForm.jsp
 - 관리자, 일반직원 로그인 폼 뷰 페이지
 - ID(employeeid), PW(주민번호 뒤 7자리) 입력
- #30. LoginFormController.java
 - 로그인 폼 컨트롤러
 - loginform.action
- #31. LoginController.java
 - 로그인 액션 컨트롤러
 - login.action

로그인 → 관리자 → employeelist.action
 일반직원 → emplist.action

- employeelist.action → 이클립스 실행 과정에서 요청을 위한 빈 파일
- Ajax.jsp
- #32. LogoutController.java
 - 로그아웃 액션 컨트롤러
 - logout.action

- #33. EmpList.jsp
 - 직원 리스트 출력 페이지(일반직원 전용)
 - 일반 직원이 접근하는 직원 데이터 출력 페이지
 - emplist.action
- #34. EmpListController.java
 - 직원 리스트 출력 액션 컨트롤러
 - emplist.action
- #35. RegionListController.java
 - 지역 리스트 출력 액션 컨트롤러
 - regionList.action
- #36. RegionList.jsp
 - 지역 리스트 뷰
 - regionlist.action
- #37. RegionInsertFormController.java
 - 지역 등록 폼 컨트롤러
 - regionInsertFrom.action
- #38. RegionInsertForm.jsp
 - 지역 등록 폼 뷰
- #39. RegionInsertController.java
 - 지역 등록 액션 컨트롤러
 - regionInsert.action
- #40. RegionUpdateFormController.java
 - 지역 변경 폼 컨트롤러
 - regionUpdateForm.action
- #41. RegionUpdateForm.jsp
 - 지역 변경 폼 뷰
- #42. RegionUpdateController.java
 - 지역 변경 액션 컨트롤러
 - regionUpdate.action
- #43. RegionDeleteController.java
 - 지역 삭제 액션 컨트롤러
 - regionDelete.action
- #44. DepartmentListController.java
 - 부서 리스트 조회 액션 컨트롤러
 - departmentlist.action
- #45. DepartmentList.jsp
 - 부서 리스트 뷰

- departmentlist.action
- #46. DepartmentInsertFormController.java
 - 부서 등록 폼 컨트롤러
 - departInsertForm.action
- #47. DepartmentInsertForm.jsp
 - 부서 등록 폼 뷰
- #48. DepartmentInsertController.java
 - 부서 등록 액션 컨트롤러
 - departInsert.action
- #49. DepartmentUpdateFormController.java
 - 부서 변경 폼 액션 컨트롤러
 - departUpdateForm.action
- #50. DepartmentUpdateForm.jsp
 - 부서 변경 폼 뷰
- #51. DepartmentUpdateController.java
 - 부서 변경 액션 컨트롤러
 - departUpdate.action
- #52. DepartmentDeleteController.java
 - 부서 삭제 액션 컨트롤러
 - departDelete.action
- #53. PositionListController.java
 - 직위 리스트 컨트롤러
 - positionlist.action
- #54. PositionList.jsp
 - 직위 리스트 뷰
 - positionlist.action
- #55. PositionInsertFormController.java
 - 직위 등록 액션 컨트롤러
 - positionInsertForm.action
- #56. PositionInsertForm.jsp
 - 직위 등록 폼 뷰
- #57. PositionInsertController.java
 - 직위 등록 액션 컨트롤러
 - positionInsert.action
- #58. PositionUpdateFormController.java
 - 직위 변경 폼 컨트롤러
 - positionUpdateForm.action

- #59. PositionUpdateForm.jsp
 - 직위 변경 폼 뷰
- #60. PositionUpdateController.java
 - 직위 변경 액션 컨트롤러
 - positionUpdate.action
- #61. PositionDeleteController.java
 - 직위 삭제 액션 컨트롤러
 - positionDelete.action
- #62. RegListController.java
 - 지역 정보 리스트 조회 액션 컨트롤러(일반직원 전용)
 - reglist.action
- #63. RegList.jsp
 - 지역 정보 리스트 뷰(일반직원 전용)
 - reglist.action
- #64. DeptListController.java
 - 부서 정보 리스트 조회 액션 컨트롤러(일반직원 전용)
 - deptlist.action
- #65. DeptList.jsp
 - 부서 정보 리스트 뷰(일반직원 전용)
 - deptlist.action
- #66. PosListController.java
 - 직위 정보 리스트 조회 액션 컨트롤러(일반직원 전용)
 - poslist.action
- #67. PosList.jsp
 - 직위 정보 리스트 뷰(일반직원 전용)
 - poslist.action

-
- dispatcher-servlet.xml
 - → 컨트롤러 객체 등록
 - → URL 매핑 주소 등록
 - → SimpleDriverDataSource 객체 등록
 - → 의존 객체 주입 설정
 - → DAO 객체 등록

```

<!-- DataSource -->
<!-- → SimpleDataSource -->
<bean id="localDataSource" class="org.springframework.jdbc.datasource.SimpleDriverDataSource"
    <property name="driverClass" value="oracle.jdbc.driver.OracleDriver"></property>
    <property name="url" value="jdbc:oracle:thin:@localhost:1521:xe"></property>
    <property name="username" value="scott"></property>
    <property name="password" value="tiger"></property>
</bean>

```


- web.xml
 - → DispatcherServlet 객체 등록
 - → URL 매핑 주소 등록
- memberlist.do
 - → 이클립스 실행 과정에서 요청을 위한 빈 파일

○ 사용자 최초 요청 주소

- <http://localhost:3306/mvcApp04/employeelist.action> → 관리자
 - <http://localhost:3306/mvcApp04/emplist.action> → 일반직원
 - 관리자 로그인 → 일반직원용 뷰 → 로그아웃 → 로그인 페이지
 - 일반직원 로그인 → 관리자용 뷰 → 로그아웃 → 로그인 페이지
-