Gallery
- Flickr -

Gallery – Flickr

❖ Flickr

- ○ 메뉴

```
   :
<c:url value="/" var="root"/>
   :
        <li class="nav-item">
           <a class="nav-link" href="${root}gallery/flickr">
              <i class="fa fa-images"></i> Flickr</a>
        </li>
   :
```

Gallery – Flickr

❖ GalleryController

```
@Controller
@RequestMapping("/gallery")
public class GalleryController {

   @RequestMapping("/flickr")
   public void flickr() {
   }

}
```

Gallery - Flickr

❖ FlickrApi.js

- fila의 FlickrApi.js
  - webapp/resources/js/FlickrApi.js 배치

Gallery – Flickr

❖ 뷰

- ○ gallery/flickr.jsp

```
<%@ contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<c:url value="/resources/js/FlickrApi.js" var="flickr"/>
<script src="${flickr}"></script>

<script>
$(function(){
   var api = new FlickrApi();
   api.search('superman');      // api.recent();

   $('#search').click(function(){
      var keyword = $('#keyword').val();
      api.search(keyword);
   });
});
</script>
```

Gallery - Flickr

❖ 뷰

- ○ gallery/flickr.jsp

```
<h1>Flickr Gallery</h1>

<div class="mb-4">
   <input type="text" id="keyword">
   <button id="search">검색</button>
</div>

<div class="row" id="panel">

</div>
```

Gallery
- lightbox -

Gallery - lightbox

- **lightbox 플러그인**
  - webapp/resoruces/bowercomponents/lightbox2 준비

Gallery - lightbox

❖ 메뉴

```
   :
<c:url value="/" var="root"/>
   :
         <li class="nav-item">
            <a class="nav-link" href="${root}gallery/lightbox">
               <i class="fa fa-images"></i> Gallery</a></li>
         <li class="nav-item">
   :
```

❖ fila의 Image 모듈 준비

- mpaper/image-mapper.xml
    - namespace 조정
    - selectList에 Pagination 타입 지정
- Image 모델
- ImageDao 인터페이스

❖ ImageService 인터페이스

```
public interface ImageService {
   public final static String IMAGE_DIR = "c:/temp/images";
   public final static String THUMB_DIR = "c:/temp/images/thumb";

   Pagination getPagination(int page) throws Exception ;

   List<Image> getList(Pagination pagination) throws Exception;

   Image getImage(int id) throws Exception ;

}
```

❖ ImageServiceImpl

```
@Service
public class ImageServiceImpl implements ImageService {

   @Autowired
   ImageDao dao;

   @Override
   public Pagination getPagination(int page) throws Exception {
      int total = dao.getCount();
      Pagination pagination = new Pagination(page, total);
      return pagination;
   }

   @Override
   public List<Image> getList(Pagination pagination)
                           throws Exception {
      return dao.selectList(pagination);
   }
```

❖ ImageServiceImpl

```
   @Override
   public Image getImage(int id)  throws Exception {
      return dao.selectOne(id);
   }

}
```

❖ 컨트롤러

- GalleryController : 목록 추출

```
public class GalleryController {
   :
   @RequestMapping("/lightbox")
   public void lightbox(
         @RequestParam(value="page", defaultValue="1") int page,
         Model model) throws Exception {
      Pagination pagination = service.getPagination(page);
      model.addAttribute("pagination", pagination);
      model.addAttribute("list", service.getList(pagination));
   }
```

- ❖ **뷰**
  - ○ gallery/lightbox.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>

<c:url value="/" var="root"/>
<link
href="${root}resources/bower_components/lightbox2/dist/css/lightbox.css" rel="stylesheet">
<script
src="${root}resources/bower_components/lightbox2/dist/js/lightbox.js"
></script>

<h1>Gallery</h1>

<a href="carousel"
   class="btn btn-primary btn-sm">carousel</a>
<hr>
```

❖ 뷰

- gallery/lightbox.jsp

```
<div class="row">
   <c:forEach var="image" items="${list}" varStatus="status">
   <div class="col-md-3 col-sm-4">
      <a href="image/${image.imageId}" data-lightbox="roadtrip">
         <img src="thumb/${image.imageId}" alt="${image.title}"/>
      </a>
      <p>
         <a href="download/${image.imageId}">
            <i class="fa fa-download"></i>
            ${image.title}
         </a>
      </p>
   </div>
   </c:forEach>
</div>
```

❖ 컨트롤러

- GalleryController : thumbnail 이미지 출력

```
@RequestMapping("/thumb/{imageId}")
public String thumb(@PathVariable int imageId, Model model)
                    throws Exception {
   Image image = service.getImage(imageId);
   String path = ImageService.THUMB_DIR +"/" +
                    image.getThumbName();

   model.addAttribute("type", image.getMimeType());
   model.addAttribute("path", path);

   return "fileView";     // 사용자 정의 뷰 이름
}
```

❖ 컨트롤러

- ○ GalleryController : 원본 이미지 보기

```
@RequestMapping("/image/{imageId}")
public String image(@PathVariable int imageId, Model model)
                              throws Exception {
    Image image = service.getImage(imageId);
    String path = ImageService.IMAGE_DIR +"/" +
                                         image.getFileName();

    model.addAttribute("type", image.getMimeType());
    model.addAttribute("path", path);

    return "fileView";      // 사용자 정의 뷰 이름
}
```

❖ 컨트롤러

- GalleryController

```
@RequestMapping("/download/{imageId}")
public String download(@PathVariable int imageId, Model model)
                    throws Exception {
   Image image = service.getImage(imageId);
   String path = ImageService.IMAGE_DIR +"/" +
                          image.getFileName();
   model.addAttribute("type", image.getMimeType());
   model.addAttribute("path", path);
   model.addAttribute("fileName", image.getFileName());

   return "download";     // 사용자 정의 뷰 이름

}
```

Gallery
- 사용자 정의 뷰 -

Gallery - 사용자 정의 뷰

❖ 사용자 정의 뷰를 해석할 BeanNameViewResolver 빈 등록

- servlet-context.xml
  - 복수 개의 ViewResolver 운영시 순서 중요!

```
<beans:bean
   class="org.springframework.web.servlet.view.BeanNameViewResolver">
   <beans:property name="order" value="1" />
</beans:bean>

   // 다른 뷰리졸버 적용 순서 수정
```

  - 컨트롤러가 리턴한 뷰의 이름을 사용자 정의 빈의 이름으로 해석하여 뷰를 찾음
  - 매칭되는 뷰가 없으면 다음 뷰 리졸버로 전달

Gallery - 사용자 정의 뷰

❖ 사용자 정의 View

- AbsctractView를 상속받아 구현
- 구현해야 할 추상 메서드

```
protected void renderMergedOutputModel(
                  Map<String, Object> model,
                  HttpServletRequest request,
                  HttpServletResponse response
) throws Exception
```

  - model : 컨트롤러가 저장한 모델
  - request : 요청 객체
  - response : 응답 객체 - 스트림으로 파일 다운로드 진행

Gallery - 사용자 정의 뷰

❖ 사용자 정의 뷰 : FileView

- 패키지 : edu.iot.butter.view

```
:
import edu.iot.butter.util.FileUtil;

@Component("fileView") // 뷰 이름 설정
public class FileView extends AbstractView {

  @Override
  protected void renderMergedOutputModel(
        Map<String, Object> model, HttpServletRequest request,
        HttpServletResponse response) throws Exception {

     String path = (String) model.get("path");
     String type = (String) model.get("type");

     File file = new File(path);
```

Gallery – 사용자 정의 뷰

❖ 사용자 정의 뷰 : FileView

```
      // 응답 헤더 설정
      response.setContentType(type);
      response.setContentLength((int) file.length());
      response.setHeader("Content-Transfer-Encoding", "binary");

      FileUtil.copy(path, response);
   }
}
```

Gallery – 사용자 정의 뷰

❖ 사용자 정의 뷰 : DownloadView

- ㅇ 패키지 : edu.iot.butter.view

```
@Component("download") // 뷰 이름 설정
public class DownloadView extends AbstractView {

   @Override
   protected void renderMergedOutputModel(
         Map<String, Object> model, HttpServletRequest request,
         HttpServletResponse response) throws Exception {

      String path = (String) model.get("path");
      String type = (String) model.get("type");
      String fileName = (String) model.get("fileName");
      fileName =URLEncoder.encode(fileName, "utf-8");

      File file = new File(path);
```

Gallery – 사용자 정의 뷰

❖ 사용자 정의 뷰 : DownloadView

```
      response.setContentType(type);
      response.setContentLength((int) file.length());
      response.setHeader("Content-Disposition",
               "attachment; filename=\"" + fileName + "\";");
      response.setHeader("Content-Transfer-Encoding", "binary");

      FileUtil.copy(path, response);
   }
}
```

Gallery
- carousel -

Gallery - carousel

❖ GalleryController

```
@RequestMapping("/carousel")
public void carousel(
      @RequestParam(value="page", defaultValue="1") int page,
      Model model) throws Exception {
   Pagination pagination = service.getPagination(page);
   model.addAttribute("pagination", pagination);
   model.addAttribute("list", service.getList(pagination));
}
```

Gallery - carousel

❖ 뷰

- ○ gallery/carousel.jsp

```
<style>
#demo {
   margin : 10px auto;
   width : 80%;
}
#demo img{
   width : 100%;
}
</style>

<h1>Gallery</h1>

<a href="lightbox"
   class="btn btn-primary btn-sm">lightbox</a>

<hr>
```

Gallery - carousel

❖ 뷰

- ○ gallery/carousel.jsp

```
<!--  Carousel 슬라이드 구성 -->
<div id="demo" class="carousel slide" data-ride="carousel">

  <!-- Indicators -->
  <ul class="carousel-indicators">
   <c:forEach items="${list}" varStatus="status">
      <li data-target="#demo" data-slide-to="${status.index}"
         <c:if test="${status.first}">class="active"</c:if>
      ></li>
    </c:forEach>
  </ul>
```

Gallery – carousel

❖ 뷰

- ○ gallery/carousel.jsp

```
  <!-- The slideshow -->
  <div class="carousel-inner">
   <c:forEach var="image" items="${list}" varStatus="status">
      <div class="carousel-item
            <c:if test="${status.first}">active</c:if>">
            <img src="image/${image.imageId}"
            alt="${image.title}"/>
      </div>
    </c:forEach>
  </div>

  <!-- Left and right controls -->
  <a class="carousel-control-prev" href="#demo" data-slide="prev">
    <span class="carousel-control-prev-icon"></span>
  </a>
  <a class="carousel-control-next" href="#demo" data-slide="next">
    <span class="carousel-control-next-icon"></span>
  </a>
</div>
```