

# Gallery

## - 다중 파일 업로드 -

## ❖ update.jsp

```
<h4>이미지 파일 추가</h4>
<form action="add_image" method="post"
  enctype="multipart/form-data">
  :
  <input type="file" name="files"
    multiple="multiple" accept="image/*" required>
  <p>
    <button type="submit" class="btn btn-primary">
      <i class="fas fa-check"></i> 확인
    </button>
    <button type="reset" class="btn btn-success">
      <i class="fas fa-undo"></i> 취소
    </button>
  </p>
</form>

</div>
</div>
```

## 파일 업로드

### ❖ MultipartHttpServletRequest를 이용한 업로드 파일 접근

- MultipartHttpServletRequest 인터페이스 파일 관련 주요 메서드

메서드	설명
Iterator<String> getFileNames( )	업로드 된 파일들의 이름 목록을 제공하는 iterator 리턴
MultipartFile getFile(String name)	파라미터 이름이 name인 업로드 파일 정보 리턴
<b>List&lt;MultipartFile&gt;</b> <b>getFiles(String name)</b>	파라미터 이름이 name인 업로드 파일 정보 목록 리턴
Map<String, MultipartFile> getFileMap( )	파라미터 이름을 키로 파라미터에 해당하는 파일 정보를 값으로 하는 Map 리턴

```
List<MultipartFile> fileList = request.getFiles("files");
for(MultipartFile imageFile: fileList) {
    // file 로 업로드 처리
}
```

# 파일 업로드

---

## ❖ 파일 업로드

```
@RequestMapping(value="/update/add_images",
                method=RequestMethod.POST)
public String addImagesSubmit(GalleryImage image,
                             MultipartHttpServletRequest request)
    throws Exception {

    List<MultipartFile> fileList = request.GetFiles("files");
    for(MultipartFile imageFile : fileList) {
        uploadImage(image, imageFile);
        service.addImage(image);
    }

    return "redirect:/gallery/update/" + image.getGalleryId();
}
```