

1. 컨트롤러에 Service인터페이스를 받은ServiceImpl이 아니라 그냥 Service로 객체 service를 만드는 이유?
(Service를 받은 ServiceImpl에 repository의 create, list메소드를 쓸 수 있으니 ServiceImpl를 써야하는것 아닌지 궁금합니다.
그렇게 해보니까 에러가 나긴했는데 그 이유도 궁금합니다.)

2. service와 serviceImpl을 만드는 명확한 이유를 잘 모르겠습니다.

그냥 컨트롤러에 repository객체를 만들어서 repository에 있는 create나 list메소드를 써도 되는게 아닌지? 헛갈립니다.

(이 방식도 예러가 나던데 그 이유가 궁금합니다.)

3. self.location을 하는 이유는 버튼 작동방식에서 정보 전송 없이 그냥 페이지의 전환일 때 사용되는지?
왜 이름이 self.location인가요?

4. #의 의미

5. Model의 정확한 쓰임 (그냥 컨트롤러에서 addAttribute를 하기위해 쓰이는건지?)

을 아직 잘 모르겠습니다 ㅠ.

```
</tr>
<tr th:each="memberElem : ${memberList}"> <!-- 컨트롤러에서 addAttr한 attrName이 {}에 들어간다. -->
  <td align="center" th:text="${memberElem.memberNo}"></td>
  <td align="center" th:text="${memberElem.id}"></td>
  <td align="center" th:text="${memberElem.pw}"></td>
  <td align="center" th:text="${memberElem.name}"></td>
  <td align="center" th:text="${memberElem.gender}"></td>
  <td align="center" th:text="${memberElem.age}"></td>
  <td align="center" th:text="${memberElem.location}"></td>
```

```
<!-- <td align="center" th:text="${memberElem.interestedGenre}"></td> -->
<td align="center" th:href="@{|read?memberNo=${memberElem.memberNo}|}" th:text="${memberElem.interestedGenre}"></td>
<!-- 위에 다른 표기방식과 같은 결과를 출력하던데, 용도적인 측면에서 어떤 차이가 있는지 궁금합니다 -->
```

```
<td align="center" th:text="${memberElem.interestedArtist}"></td>
<td align="center" th:text="${#dates.format(memberElem.regDate, 'yyyy-MM-dd HH:mm:ss')}"></td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```