

## src/app.py:

- Flask 애플리케이션 기본 및 oracle 드라이버 설정

## src/views.py:

- db 쿼리 및 사용자에게 보낼 html 을 정함
- 여기서 기능을 추가하면 됩니다

## src/test\_views.py:

- 예시 db 쿼리 입니다

## test\_views.py

```
@test.route('/test/employee', methods=['GET', 'POST'])
def employee():
```

여기서 모든 부서 관련 쿼리를 처리합니다

```
if request.method == 'POST': # 사용자 요청 처리
```

post 요청이 아니므로 일단은 무시 됩니다

```
if not employees: # 직원 목록이 비어있으면 모든 직원 조회
```

직원목록이 비어 있으므로 모든 직원이 조회됩니다

```
SELECT employee_id, employee_name, department_name, employee_address,
employee_phone_number, employee_email
FROM employee JOIN department
ON employee.department_id = department.department_id
ORDER BY employee_id
```

사용자에게 보낼 html 을 정하고 전달할 인수를 설정합니다  
모든 직원 목록을 보냅니다(다른 기능 추가시 salary.html)

```
render_template('test/employee.html', employees=employee
,employee_to_edit=employee_to_edit)
```

전달된 모든 db 내용을 표시합니다

```
<table>
  <thead>
    <tr>
```

```

        <th>직원 ID</th>
        <th>직원 이름</th>
        <th>부서 이름</th>
        <th>주소</th>
        <th>전화번호</th>
        <th>이메일</th>
    </tr>
</thead>
<tbody>
    {% for employee in employees %}
        <tr>
            <td>{{ employee[0] }}</td>
            <td>{{ employee[1] }}</td>
            <td>{{ employee[2] }}</td>
            <td>{{ employee[3] }}</td>
            <td>{{ employee[4] }}</td>
            <td>{{ employee[5] }}</td>
        </tr>
    {% endfor %}
</tbody>
</table>

```

사용자가 id 를 넣고 삭제 버튼을 누른다고 가정  
post 요청으로 delete\_employee 으로 요청합니다

```

<div class="section delete-employee" id="section">
    <h3>직원 삭제</h3>
    <form method="POST">
        <label for="delete_employee">직원 ID:</label>
        <input type="text" id="delete_employee_id" name="employee_id"
required>
        <br><br>
        <input type="submit" name="delete_employee" value="직원 삭제">
    </form>
</div>

```

삭제 쿼리를 진행

```

if request.method == 'POST': # 사용자 요청 처리
    여기서 POST 요청이므로 통과
    case 'delete_employee': # 직원 삭제 요청
        employee_id = request.form['employee_id']
        cursor.execute("""
            DELETE FROM employee
            WHERE employee_id = :employee_id
            """, {'employee_id': employee_id})

```

```
print(f"Deleted employee: {employee_id}")
connection.commit()
```

다시 모든 직원을 조회

```
cursor.execute("""
    SELECT employee_id, employee_name, department_name,
employee_address, employee_phone_number, employee_email
    FROM employee JOIN department
    ON employee.department_id = department.department_id
    ORDER BY employee_id
""")
```