Spring MVC 與 JdbcTemplate

段維瀚 老師

Session

- StringMVC 配置 JdbcTemplate
- 一對多資料表建立
- 一對多資料查詢-RowMapper
- 一對多資料查詢-SimpleFlatMapper
- Lab 實作練習

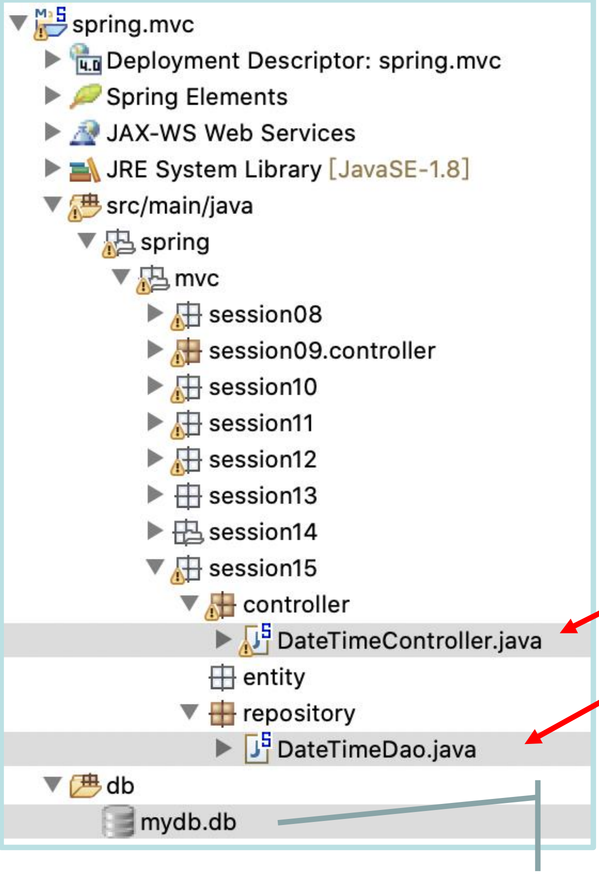
StringMVC 配置 JdbcTemplate

- pom.xml
  - 參考配置連結
- WEB-INF/springmvc-servlet.xml
  - 參考配置連結

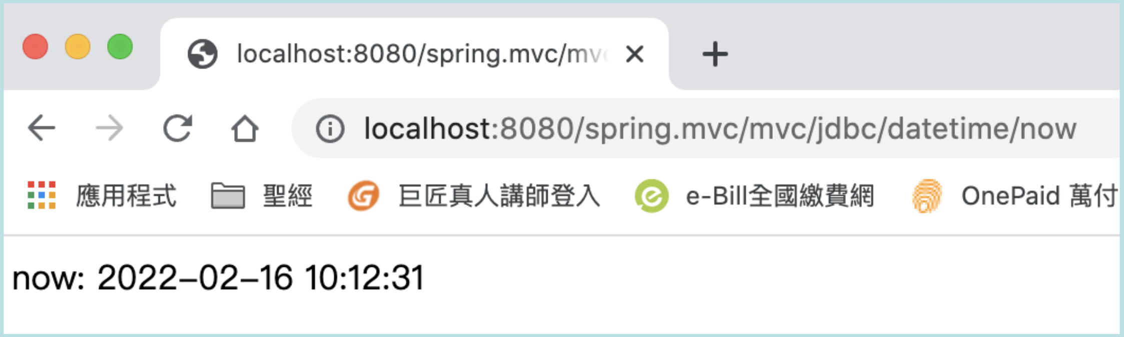
測試

SQL 語法：select datetime('now', '+8 hour')

+8 hour 表示將格林威治時區轉換成東八區



```
▼ spring.mvc
  ▶ Deployment Descriptor: spring.mvc
  ▶ Spring Elements
  ▶ JAX-WS Web Services
  ▶ JRE System Library [JavaSE-1.8]
  ▼ src/main/java
    ▼ spring
      ▼ mvc
        ▶ session08
        ▶ session09.controller
        ▶ session10
        ▶ session11
        ▶ session12
        ▶ session13
        ▶ session14
        ▼ session15
          ▼ controller
            ▶ DateTimeController.java
            entity
          ▼ repository
            ▶ DateTimeDao.java
  ▼ db
    mydb.db
```



localhost:8080/spring.mvc/mv

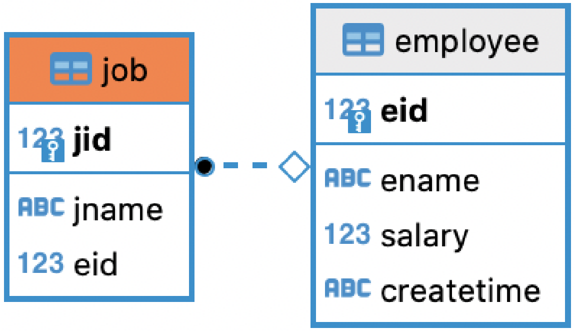
localhost:8080/spring.mvc/mvc/jdbc/datetime/now

```
now: 2022-02-16 10:12:31
```

寫一個 DateTimeController.java 與 DateTimeDao.java 進行測試

若測試成功會系統會自動建立 mydb.db 資料庫

E-R Model



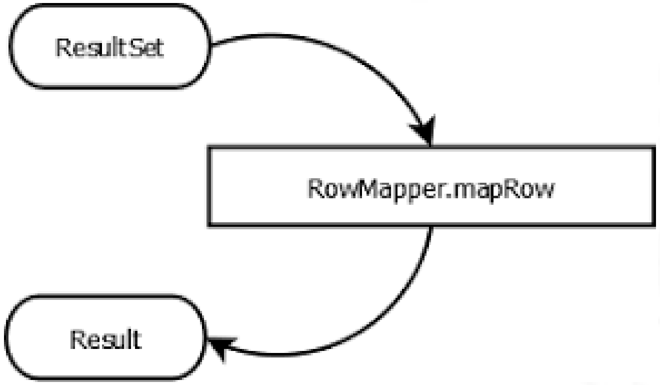
```
-- 建立 Employee 資料表
create table IF NOT EXISTS employee (
    eid integer PRIMARY KEY, -- 主鍵，員工 id (自行產生序號: 1, 2, 3, ...)
    ename text, -- 員工姓名
    salary integer, -- 員工薪資
    createtime datetime default current_timestamp -- 建檔時間
);

-- 建立 Job 資料表
create table IF NOT EXISTS job(
    jid integer PRIMARY KEY, -- 主鍵，工作 id
    jname text, -- 工作名稱
    eid integer not null, -- 員工 id
    foreign key(eid) references employee(eid) -- 外鍵關聯
);
```

建立範例資料

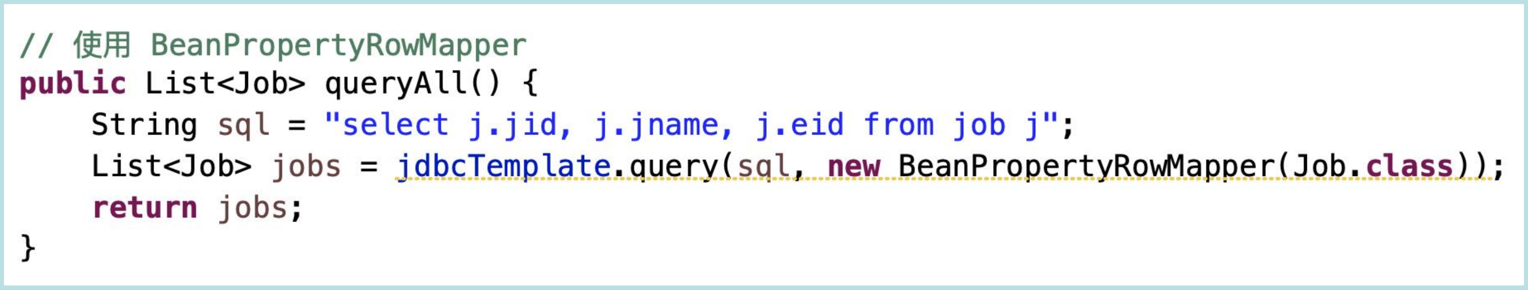
```
-- 建立 Employee 範例資料
insert into employee(ename, salary) values('John', 40000);
insert into employee(ename, salary) values('Mary', 50000);
insert into employee(ename, salary) values('Bobo', 60000);
insert into employee(ename, salary) values('Mark', 70000);

-- 建立 Job 範例資料
insert into job(jname, eid) values('Job A', 1);
insert into job(jname, eid) values('Job B', 1);
insert into job(jname, eid) values('Job C', 2);
insert into job(jname, eid) values('Job D', 2);
insert into job(jname, eid) values('Job E', 4);
insert into job(jname, eid) values('Job F', 4);
insert into job(jname, eid) values('Job G', 4);
```

一對多資料查詢-RowMapper

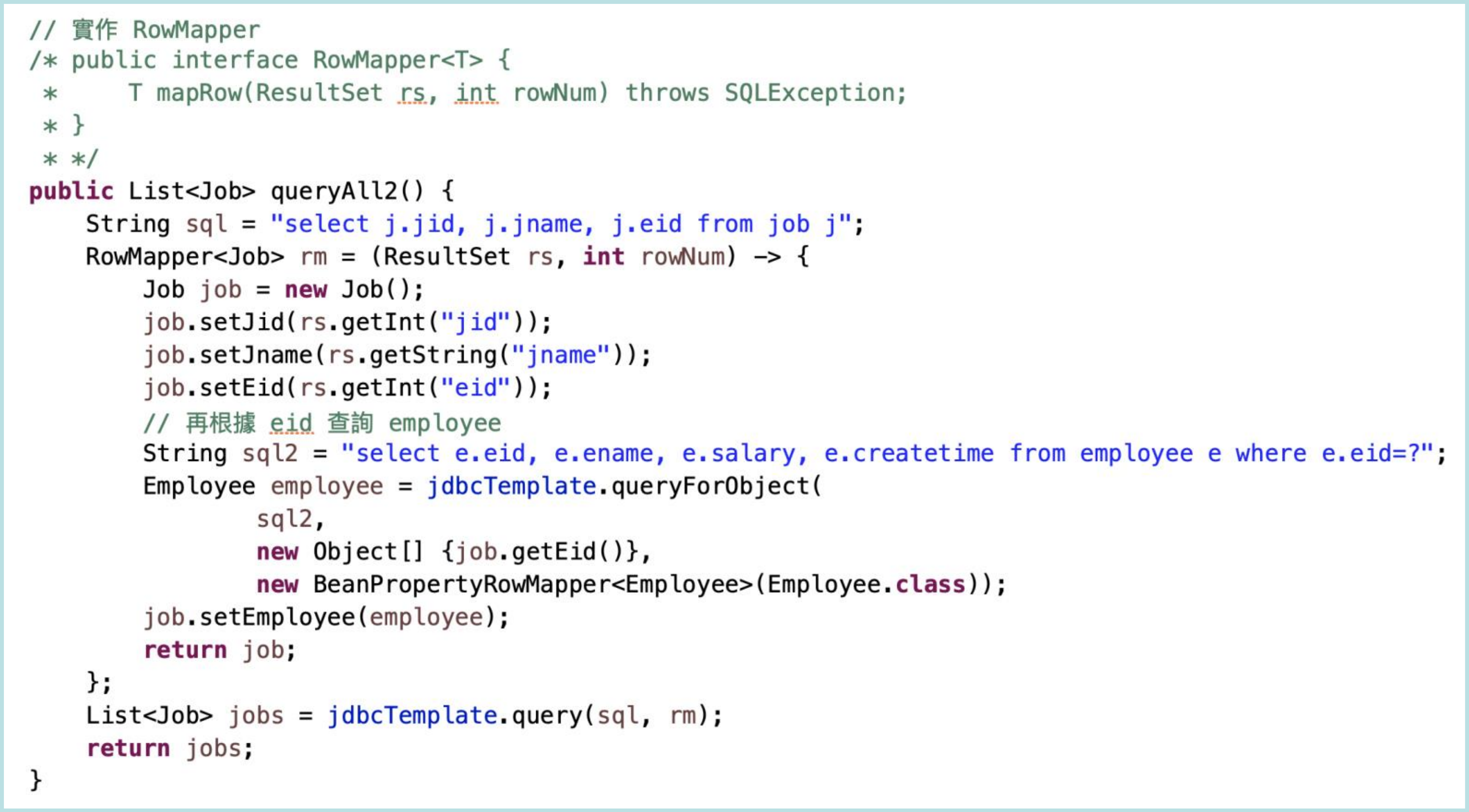
- ORM Mapper
  - Spring 提供 RowMapper 可將查詢結果封裝於Domain物件中。



```
List<Customer> list = jdbcTemplate.query(sql, (ResultSet rs, int i) -> {
            Customer customer = new Customer();
            // block of code
            return customer;
});
```



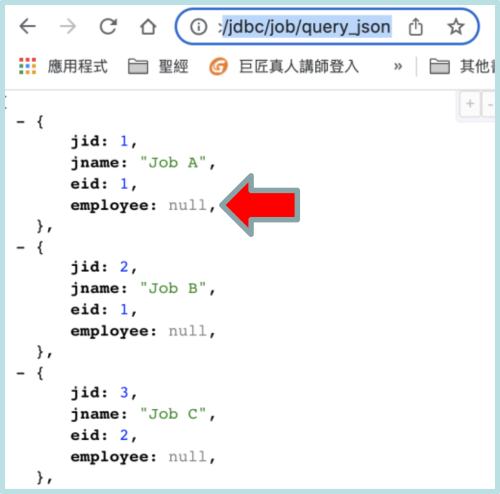
```java
// 使用 BeanPropertyRowMapper
public List<Job> queryAll() {
    String sql = "select j.jid, j.jname, j.eid from job j";
    List<Job> jobs = jdbcTemplate.query(sql, new BeanPropertyRowMapper(Job.class));
    return jobs;
}
```



```java
// 實作 RowMapper
/* public interface RowMapper<T> {
 *     T mapRow(ResultSet rs, int rowNum) throws SQLException;
 * }
 * */
public List<Job> queryAll2() {
    String sql = "select j.jid, j.jname, j.eid from job j";
    RowMapper<Job> rm = (ResultSet rs, int rowNum) -> {
        Job job = new Job();
        job.setJid(rs.getInt("jid"));
        job.setJname(rs.getString("jname"));
        job.setEid(rs.getInt("eid"));
        // 再根據 eid 查詢 employee
        String sql2 = "select e.eid, e.ename, e.salary, e.createtime from employee e where e.eid=?";
        Employee employee = jdbcTemplate.queryForObject(
                sql2,
                new Object[] {job.getEid()},
                new BeanPropertyRowMapper<Employee>(Employee.class));
        job.setEmployee(employee);
        return job;
    };
    List<Job> jobs = jdbcTemplate.query(sql, rm);
    return jobs;
}
```

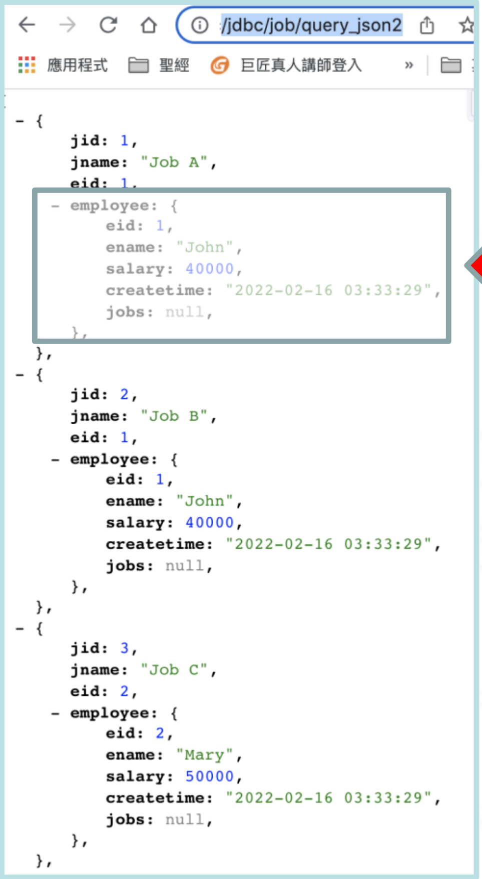
比較查詢結果

/jdbc/job/query_json
employee 無資料



```
- {
      jid: 1,
      jname: "Job A",
      eid: 1,
      employee: null,
  },
- {
      jid: 2,
      jname: "Job B",
      eid: 1,
      employee: null,
  },
- {
      jid: 3,
      jname: "Job C",
      eid: 2,
      employee: null,
  },
```

/jdbc/job/query_json2
employee 有資料



```
- {
      jid: 1,
      jname: "Job A",
      eid: 1,
    - employee: {
          eid: 1,
          ename: "John",
          salary: 40000,
          createtime: "2022-02-16 03:33:29",
          jobs: null,
      },
  },
- {
      jid: 2,
      jname: "Job B",
      eid: 1,
    - employee: {
          eid: 1,
          ename: "John",
          salary: 40000,
          createtime: "2022-02-16 03:33:29",
          jobs: null,
      },
  },
- {
      jid: 3,
      jname: "Job C",
      eid: 2,
    - employee: {
          eid: 2,
          ename: "Mary",
          salary: 50000,
          createtime: "2022-02-16 03:33:29",
          jobs: null,
      },
  },
```

一對多資料查詢-SimpleFlatMapper

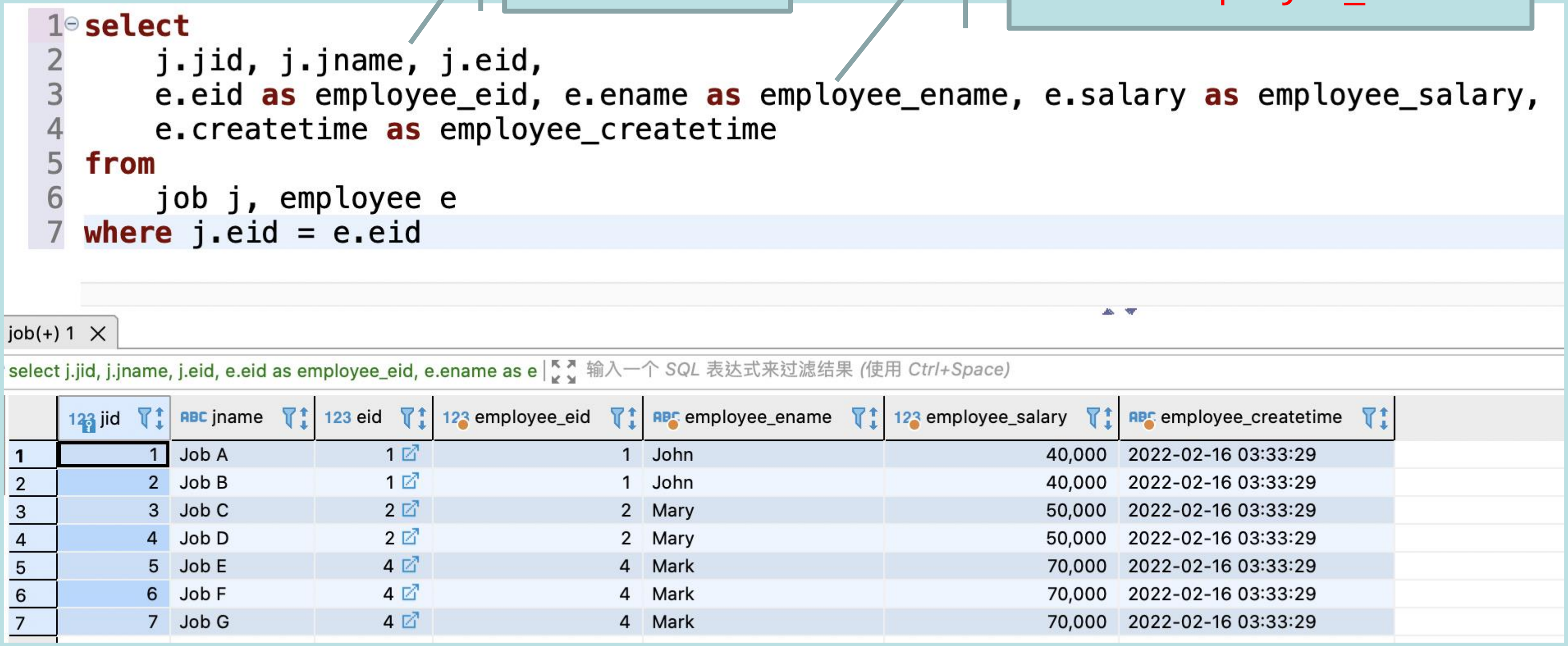
- 簡單數據庫映射
  - https://simpleflatmapper.org/

```
<dependency>
    <groupId>org.simpleflatmapper</groupId>
    <artifactId>sfm-springjdbc</artifactId>
    <version>3.11.7</version>
</dependency>
```

SQL 查詢

第一張表

第二張表的欄位別名要加上 employee_



```
select
    j.jid, j.jname, j.eid,
    e.eid as employee_eid, e.ename as employee_ename, e.salary as employee_salary,
    e.createtime as employee_createtime
from
    job j, employee e
where j.eid = e.eid
```

job(+) 1

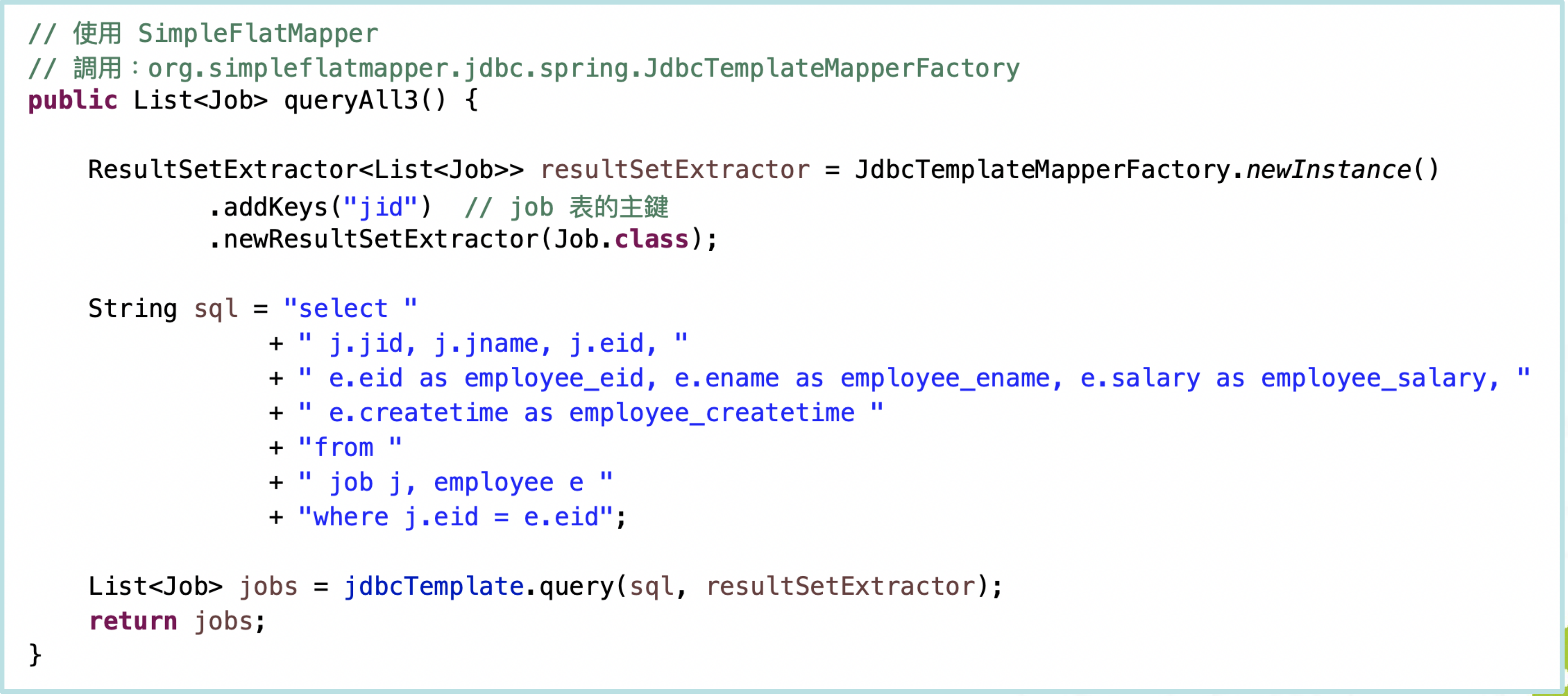
select j.jid, j.jname, j.eid, e.eid as employee_eid, e.ename as e | 输入一个 SQL 表达式来过滤结果 (使用 Ctrl+Space)

	jid	jname	eid	employee_eid	employee_ename	employee_salary	employee_createtime
1	1	Job A	1	1	John	40,000	2022-02-16 03:33:29
2	2	Job B	1	1	John	40,000	2022-02-16 03:33:29
3	3	Job C	2	2	Mary	50,000	2022-02-16 03:33:29
4	4	Job D	2	2	Mary	50,000	2022-02-16 03:33:29
5	5	Job E	4	4	Mark	70,000	2022-02-16 03:33:29
6	6	Job F	4	4	Mark	70,000	2022-02-16 03:33:29
7	7	Job G	4	4	Mark	70,000	2022-02-16 03:33:29

Job

Employee

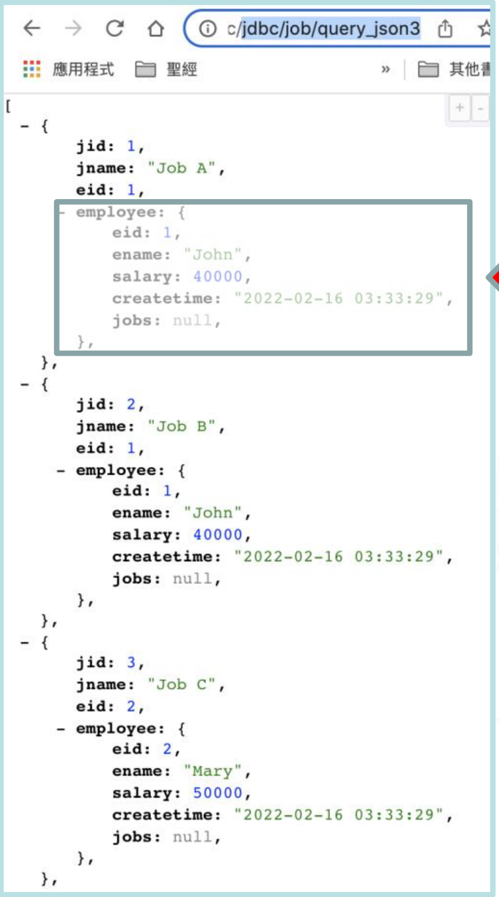
多對一資料查詢-SimpleFlatMapper



```
// 使用 SimpleFlatMapper
// 調用：org.simpleflatmapper.jdbc.spring.JdbcTemplateMapperFactory
public List<Job> queryAll3() {

    ResultSetExtractor<List<Job>> resultSetExtractor = JdbcTemplateMapperFactory.newInstance()
            .addKeys("jid")  // job 表的主鍵
            .newResultSetExtractor(Job.class);

    String sql = "select "
                + " j.jid, j.jname, j.eid, "
                + " e.eid as employee_eid, e.ename as employee_ename, e.salary as employee_salary, "
                + " e.createtime as employee_createtime "
                + "from "
                + " job j, employee e "
                + "where j.eid = e.eid";

    List<Job> jobs = jdbcTemplate.query(sql, resultSetExtractor);
    return jobs;
}
```

查詢結果

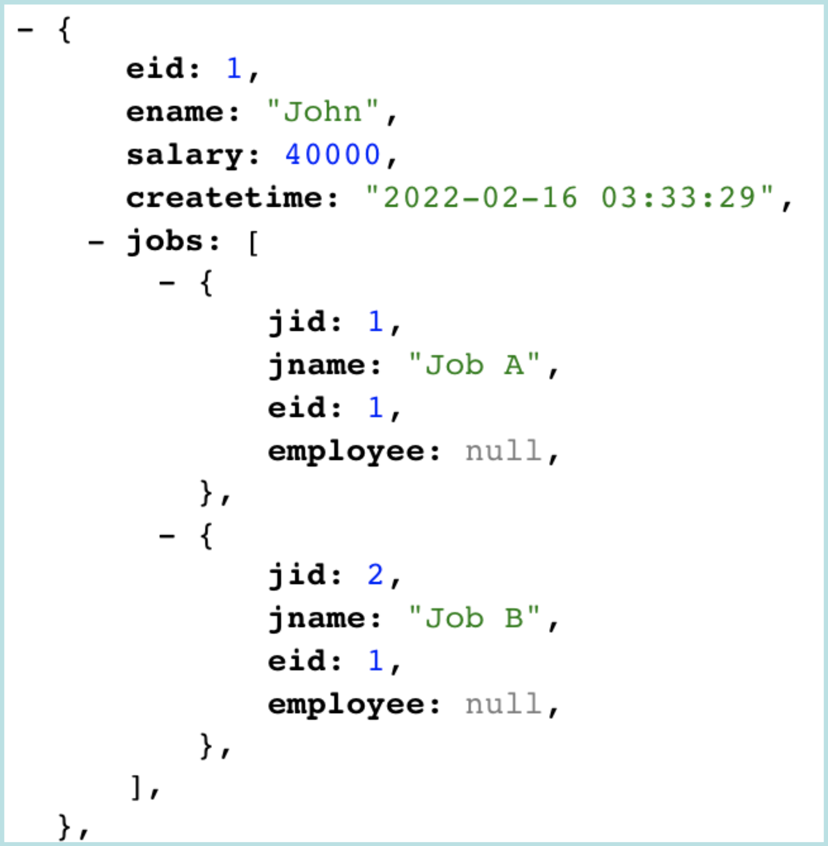
/jdbc/job/query_json3
employee 亦有資料



```
[
  - {
        jid: 1,
        jname: "Job A",
        eid: 1,
      - employee: {
            eid: 1,
            ename: "John",
            salary: 40000,
            createtime: "2022-02-16 03:33:29",
            jobs: null,
        },
    },
  - {
        jid: 2,
        jname: "Job B",
        eid: 1,
      - employee: {
            eid: 1,
            ename: "John",
            salary: 40000,
            createtime: "2022-02-16 03:33:29",
            jobs: null,
        },
    },
  - {
        jid: 3,
        jname: "Job C",
        eid: 2,
      - employee: {
            eid: 2,
            ename: "Mary",
            salary: 50000,
            createtime: "2022-02-16 03:33:29",
            jobs: null,
        },
    },
```

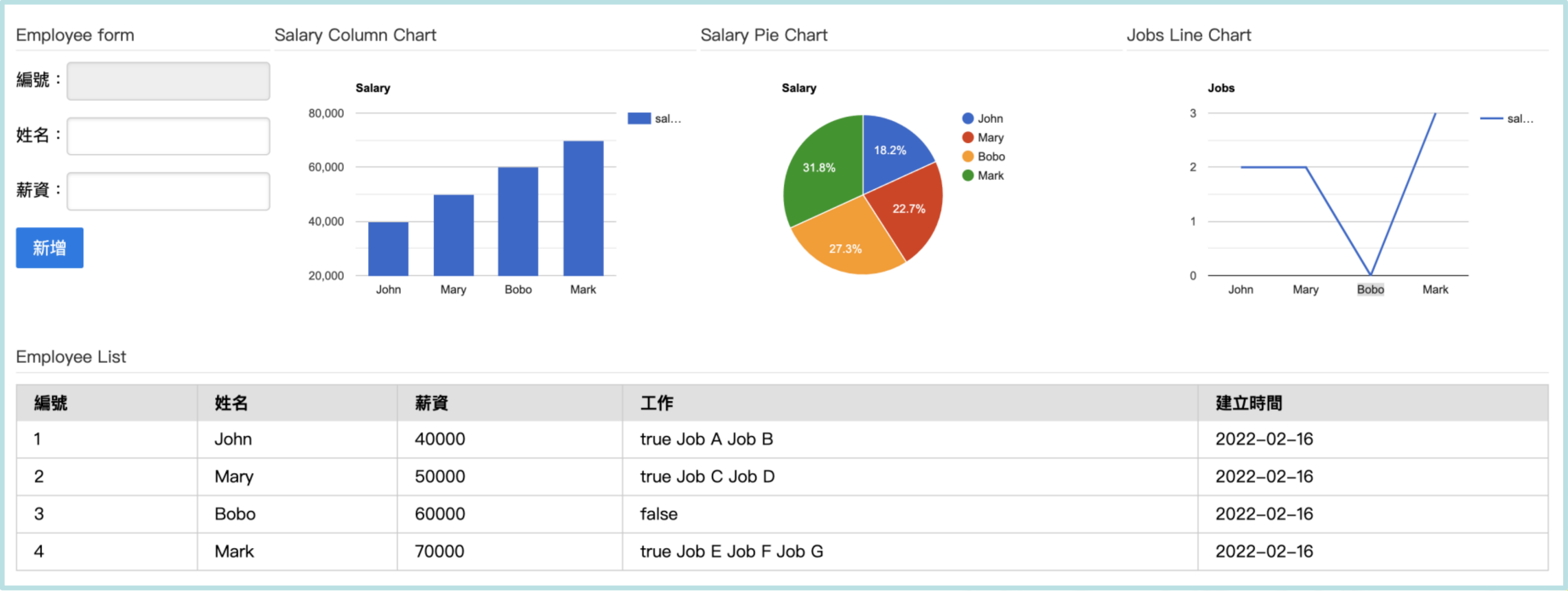
一對多查詢 Lab

- 查詢 Employee 連同 Job 也一並查出



```
- {
        eid: 1,
        ename: "John",
        salary: 40000,
        createtime: "2022-02-16 03:33:29",
      - jobs: [
          - {
                jid: 1,
                jname: "Job A",
                eid: 1,
                employee: null,
            },
          - {
                jid: 2,
                jname: "Job B",
                eid: 1,
                employee: null,
            },
        ],
  },
```

結合 Google Chart



Employee form

編號：	
姓名：	
薪資：	

新增

Salary Column Chart

Salary Pie Chart

Jobs Line Chart

Employee List

編號	姓名	薪資	工作	建立時間
1	John	40000	true Job A Job B	2022-02-16
2	Mary	50000	true Job C Job D	2022-02-16
3	Bobo	60000	false	2022-02-16
4	Mark	70000	true Job E Job F Job G	2022-02-16

Google Chart :
https://developers-dot-devsite-v2-prod.appspot.com/chart/interactive/docs/gallery/piechart.html

程式碼參考：
https://github.com/vincenttuan/SpringMVCExpert2022/blob/main/src/main/webapp/WEB-INF/views/session15/employee.jsp