

SpringMVC + JdbcTemplate

專題實作

說明

- 實作目的
 - 資料表 一對多、多對一 進行關聯設計
 - 資料表聯集查詢、分頁查詢
 - **REST** 風格設計
 - **CRUD** 介面操作
 - 利用 **SimpleFlatMapping** 進行關聯資料查詢對應
 - 透過 **Goggle Chart** 繪製統計圖表

畫面成果展示 1

Employee form | [Job form](#)

編號: 1

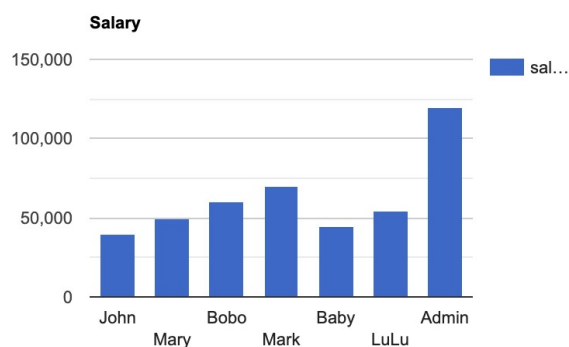
姓名: John

薪資: 40000

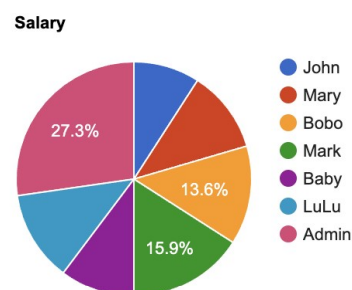
新增

修改

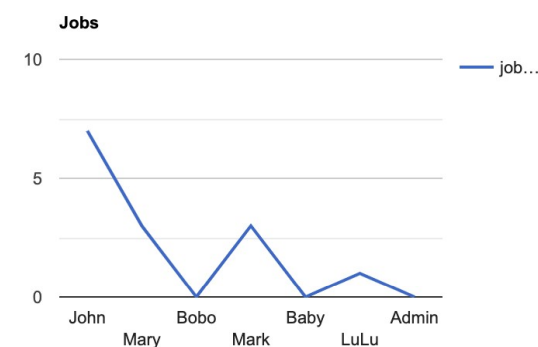
Salary Column Chart



Salary Pie Chart



Jobs Line Chart



Employee List

編號	姓名	薪資	工作	建立時間
1	John	40000	Job A Job B Job H Job I Job J Job M Job N	2022-02-16
2	Mary	50000	Job C Job D Job K	2022-02-16
3	Bobo	60000		2022-02-16
4	Mark	70000	Job E Job G Job L	2022-02-16
5	Baby	45000		2022-02-17
6	LuLu	55000	Job F	2022-02-17
7	Admin	120000		2022-02-17

畫面成果展示 2

Employee form | Job form

Job List | [全部查詢](#) | 分頁查詢： 1 2 3

Job Bar Chart

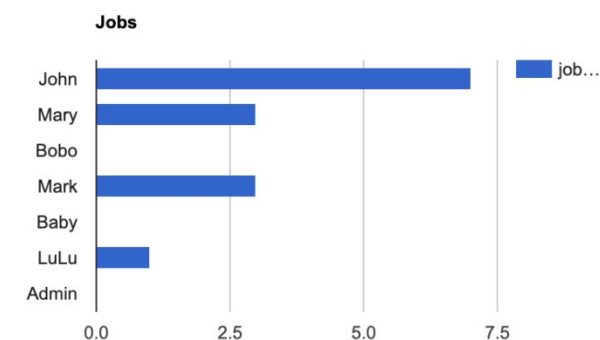
編號： 名稱：

員工： John ▾

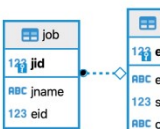
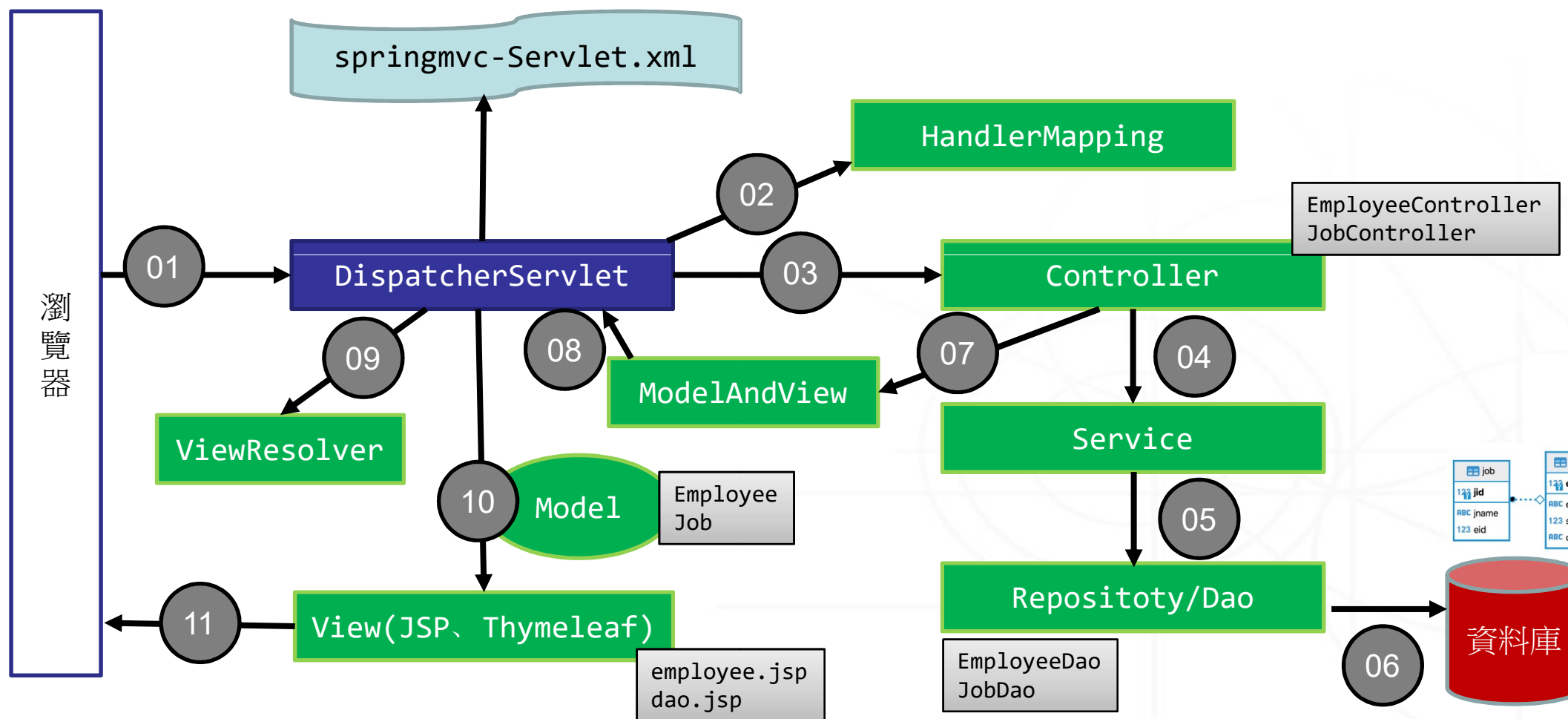
新增

修改

編號	名稱	員工資料	
		員編	姓名
1	Job A	1	John
2	Job B	1	John
3	Job C	2	Mary
4	Job D	2	Mary
5	Job E	4	Mark
6	Job F	6	LuLu
7	Job G	4	Mark
8	Job H	1	John
9	Job I	1	John
10	Job J	1	John
11	Job K	2	Mary
12	Job L	4	Mark
13	Job M	1	John
14	Job N	1	John



系統流程圖



資料表設計與 E-R圖

建立 employee 資料表

eid 主鍵，具有唯一性(自行產生序號：1、2、3、4 ...)

ename 員工姓名

salary 員工薪資

createtime 建檔時間 default current_timestamp 預設會注入現在時間

create table if not exists employee (

eid **integer primary key**,

ename text,

salary **integer**,

createtime datetime **default** current_timestamp

建立 job 資料表

jid 主鍵

jname 工作名稱

eid 員工編號

外鍵關聯 job.eid 對應到 employee.eid

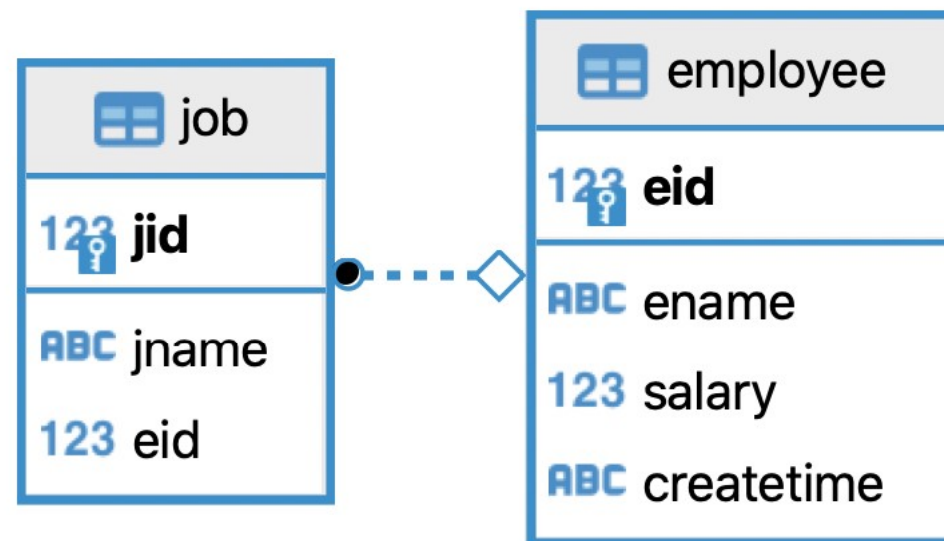
create table if not exists job (

jid **integer primary key**,

jname text,

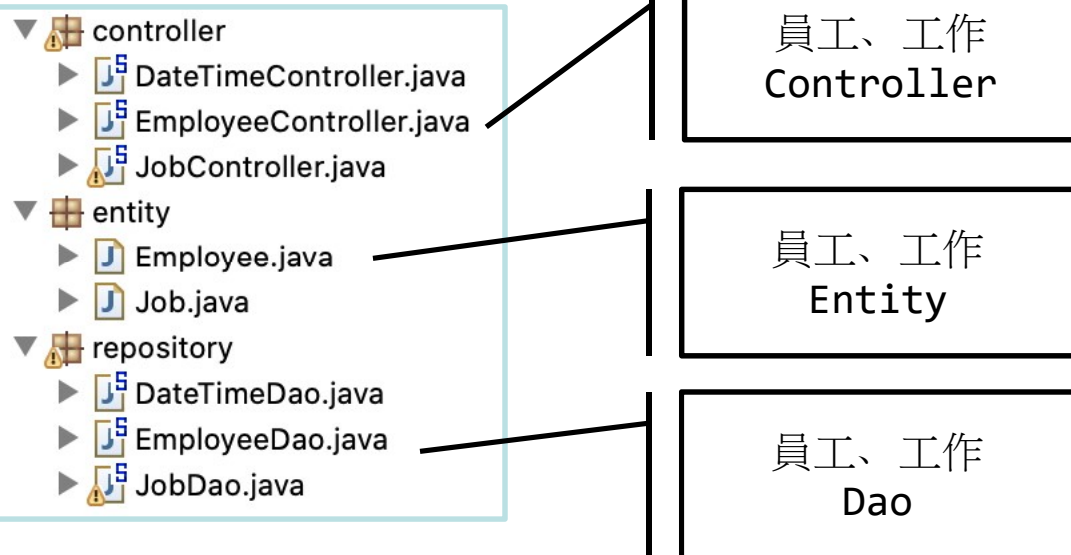
eid **integer not null**,

foreign key(eid) **references** employee(eid)

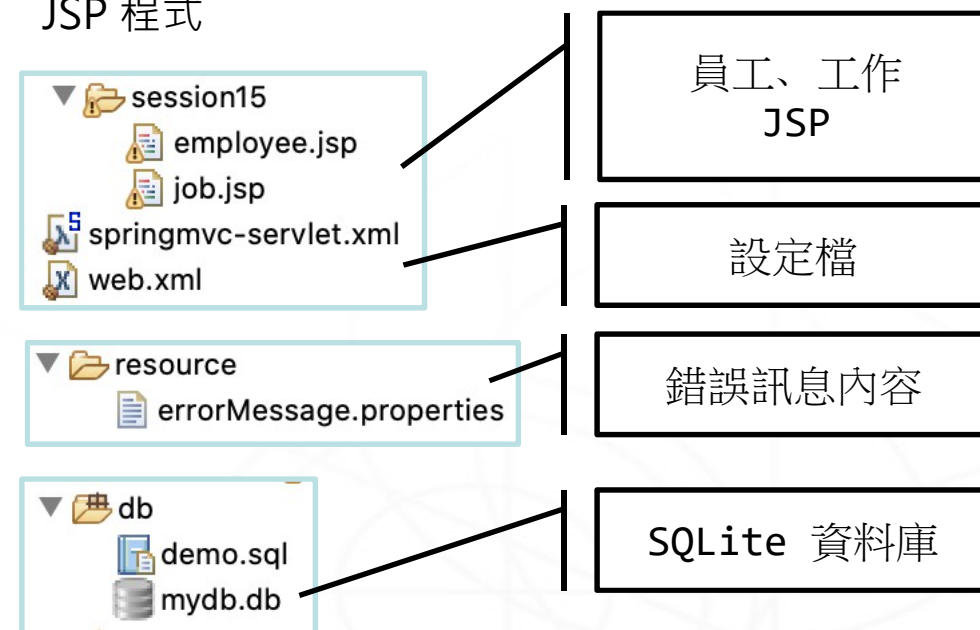


專案檔案結構

Java 程式



JSP 程式



Entity-Employee

```
public class Employee {

    private Integer eid;

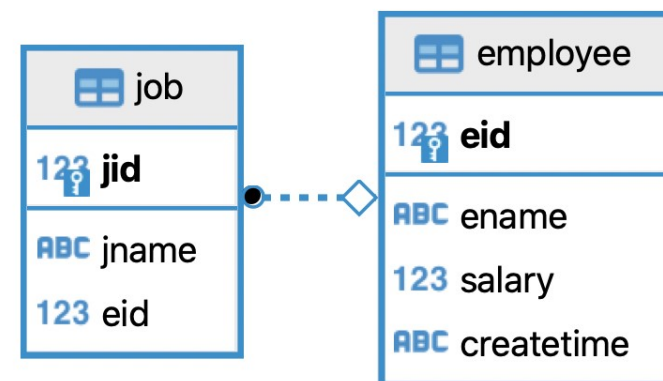
    @Size(min = 2, max = 50, message = "{employee.ename.size}")
    private String ename;

    @NotNull(message = "{employee.salary.empty}")
    @Range(min = 30000, max = 300000, message = "{employee.salary.range}")
    private Integer salary;

    @JsonFormat(pattern="yyyy-MM-dd HH:mm:ss",timezone="GMT+8") // 返回時間類型
    @DateTimeFormat(pattern="yyyy-MM-dd HH:mm:ss") //接收時間類型
    private Date createtime;

    // 加入 Job : 一位員工可以有多个工作
    private List<Job> jobs;

    // getter、setter
}
```



WEB-INF/resource/errorMessage.properties

employee.ename.size=員工姓名必須介於 {min}~{max} 個字之間
 employee.salary.empty=薪資不可空白
 employee.salary.range=薪資必須介於 {min}~{max} 之間

Entity-Job

```
public class Job {

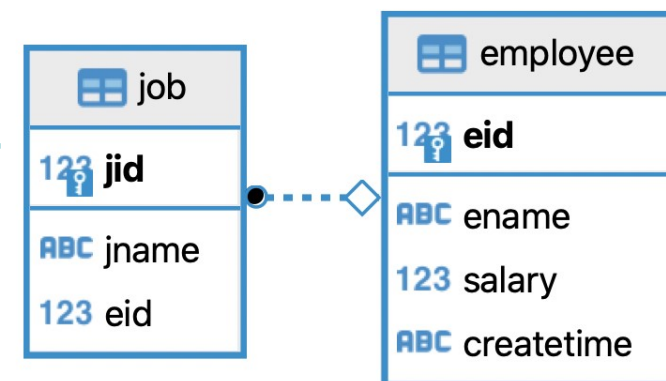
    private Integer jid;

    @Size(min = 2, max = 50, message = "{job.jname.size}")
    private String jname;

    private Integer eid;

    // 加入 Employee : 一位工作可以對應一個員工
    private Employee employee;

    // getter、setter
}
```



WEB-INF/resource/errorMessage.properties

job.jname.size=字數必須介於 {min}~{max} 個字之間

Repository / Dao

```
public interface IEmployeeDao {  
    // 建立 employee 資料表  
    public boolean createTable();  
    // 取得單筆 employee 資料  
    public Employee get(Integer eid);  
    // 新增 employee 資料  
    public int add(Employee emp);  
    // 修改 employee 資料  
    public int update(Employee emp);  
    // 查詢全部資料-使用 BeanPropertyRowMapper  
    public List<Employee> queryAll();  
    // 查詢全部資料-實作 RowMapper  
    public List<Employee> queryAll2();  
    // 查詢全部資料-使用 SimpleFlatMapper  
    public List<Employee> queryAll3();  
}
```

```
public interface IJobDao {  
    // 每頁筆數  
    int LIMIT = 5;  
    // 建立 job 資料表  
    public boolean createTable();  
    // 取得單筆 job 資料  
    public Job get(Integer eid);  
    // 新增 job 資料  
    public int add(Job job);  
    // 修改 job 資料  
    public int update(Job job);  
    // 查詢所有筆數  
    public int getCount();  
    // 不分頁全部查詢  
    public List<Job> query();  
    // 判斷 HttpSession 值決定是否要分頁  
    public List<Job> query(Object httpSessionValue);  
    // 分頁查詢  
    public List<Job> queryPage(int offset);  
    // 查詢全部資料-使用 BeanPropertyRowMapper  
    public List<Job> queryAll();  
    // 查詢全部資料-實作 RowMapper  
    public List<Job> queryAll2();  
    // 查詢全部資料-使用 SimpleFlatMapper  
    public List<Job> queryAll3();  
}
```

Controller

- **EmployeeController.java**

- **@RequestMapping("/session15/employee")**
 - @GetMapping("/") 多筆查詢
 - @GetMapping("/{eid}") 單筆查詢
 - @PostMapping("/") 新增
 - @PutMapping("/") 修改

- **JobController.java**

- **@RequestMapping("/session15/job")**
 - @GetMapping("/") 多筆查詢
 - @GetMapping("/{jid}") 單筆查詢
 - @GetMapping("/page/{num}") 分頁查詢
 - @PostMapping("/") 新增
 - @PutMapping("/") 修改

畫面成果展示 1

薪資統計比較表

工作數量累計

Employee form | [Job form](#)

Salary Column Chart

Salary Pie Chart

Jobs Line Chart

編號:

姓名:

員工姓名必須介於 2~50 個字之間

薪資:

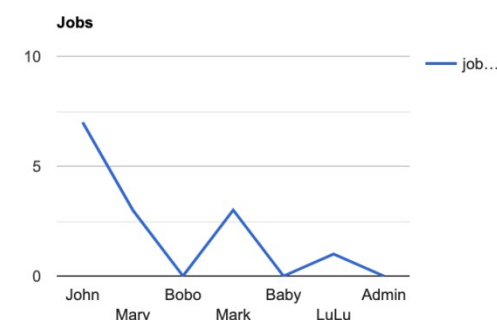
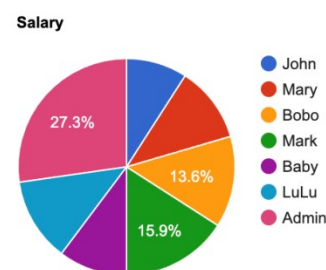
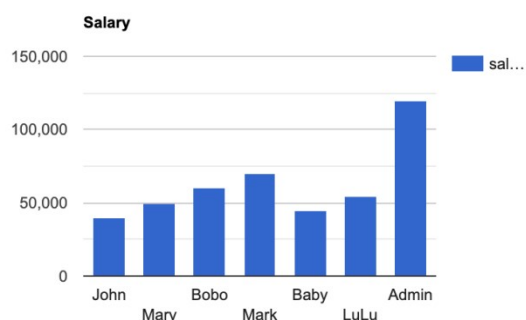
薪資不可空白

新增

修改

薪資不可空白
員工姓名必須介於 2~50 個字之間

錯誤訊息



Employee List

編號	姓名	薪資	工作	建立時間
1	John	40000	Job A Job B Job H Job I Job J Job M Job N	2022-02-16
2	Mary	50000	Job C Job D Job K	2022-02-16
3	Bobo	60000		2022-02-16
4	Mark	70000	Job E Job G Job L	2022-02-16
5	Baby	45000		2022-02-17
6	LuLu	55000	Job F	2022-02-17
7		00000		2022-02-17

修改員工資料

畫面成果展示 2

分頁查詢

修改員工資料

員工工作數量累計

Employee form | Job form

Job List | [全部查詢](#) | 分頁查詢： 1 2 3

Job Bar Chart

編號： 名稱：

員工： John ▾

新增

修改

編號	名稱	員工資料	
		員編	姓名
1	Job A	1	John
2	Job B	1	John
3	Job C	2	Mary
4	Job D	2	Mary
5	Job E	4	Mark
6	Job F	6	LuLu
7	Job G	4	Mark
8	Job H	1	John
9	Job I	1	John
10	Job J	1	John
11	Job K	2	Mary
12	Job L	4	Mark
13	Job M	1	John
14	Job N	1	John

修改工作資料

