

# 后端改造

---

实体类数据类型要由Int 改成 Integer

Employee.java代码如下:

```
package com.lz.hrsys.entity;

public class Employee {
    private Integer id;
    private Integer number;
    private String name;
    private String gender;
    private Integer age;
    private Department dep;

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public Integer getNumber() {
        return number;
    }

    public void setNumber(Integer number) {
        this.number = number;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getGender() {
        return gender;
    }

    public void setGender(String gender) {
        this.gender = gender;
    }

    public Integer getAge() {
        return age;
    }

    public void setAge(Integer age) {
        this.age = age;
    }
}
```

```

    }

    public Department getDep() {
        return dep;
    }

    public void setDep(Department dep) {
        this.dep = dep;
    }

    @Override
    public String toString() {
        return "Employee{" +
            "id=" + id +
            ", number=" + number +
            ", name='" + name + '\'' +
            ", gender='" + gender + '\'' +
            ", age=" + age +
            ", dep=" + dep +
            '}';
    }
}

```

Department.java代码如下:

```

package com.lz.hrsys.entity;

public class Department {
    private Integer id;
    private String name;
    private Integer number;

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public Integer getNumber() {
        return number;
    }

    public void setNumber(Integer number) {
        this.number = number;
    }
}

```

```

@Override
public String toString() {
    return "Department{" +
        "id=" + id +
        ", name='" + name + '\'' +
        ", number=" + number +
        '}';
}
}

```

Controller层代码要加上@CrossOrigin, public List search( Employee condition) 不要加上其他参数。

EmployeeController代码如下:

```

package com.lz.hrsys.controller;

import com.lz.hrsys.entity.Employee;
import io.swagger.v3.oas.annotations.Operation;
import io.swagger.v3.oas.annotations.Parameter;
import io.swagger.v3.oas.annotations.tags.Tag;
import jakarta.annotation.Nullable;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;
import com.lz.hrsys.service.EmployeeService;
import com.lz.hrsys.service.DepartmentService;

import java.util.List;

@CrossOrigin
@RestController
@RequestMapping("emp")
@Tag(name="Employee Controller",description = "员工接口")
public class EmployeeController {
    @Autowired
    EmployeeService empService;
    @Autowired
    DepartmentService departmentService;
    Logger logger= LoggerFactory.getLogger(EmployeeController.class);
    @Operation(summary = "search employee",description = "根据条件查询所有员工")
    @GetMapping
    public List<Employee> search( Employee condition){

        List<Employee> list = empService.search(condition);
        return list;
    }
    // 根据ID查询员工
    @Operation(summary = "search employee by id",description = "根据ID查询员工")
    @GetMapping("/{id}")
    public Employee search(@Parameter(description="员工id",name="id",required=true) @PathVariable Integer id){
        Employee emp = empService.searchById(id);
        return emp;
    }
}

```

```
//    新增用户
@Operation(summary = "add employee",description = "新增员工")
@PostMapping
public boolean add(@RequestBody Employee emp){
    boolean flag = empService.add(emp);
    return flag;
}

//    修改用户
@Operation(summary = "update employee",description = "修改员工")
@PutMapping
public boolean update(@RequestBody Employee emp){
    boolean flag = empService.update(emp);
    return flag;
}

//    删除用户
@Operation(summary = "delete employee",description = "删除员工")
@DeleteMapping("{id}")
public boolean delete(@Parameter(description="员工id",name="id",required=true) @PathVariable Integer id){
    boolean flag = empService.delete(id);
    return flag;
}

}
```