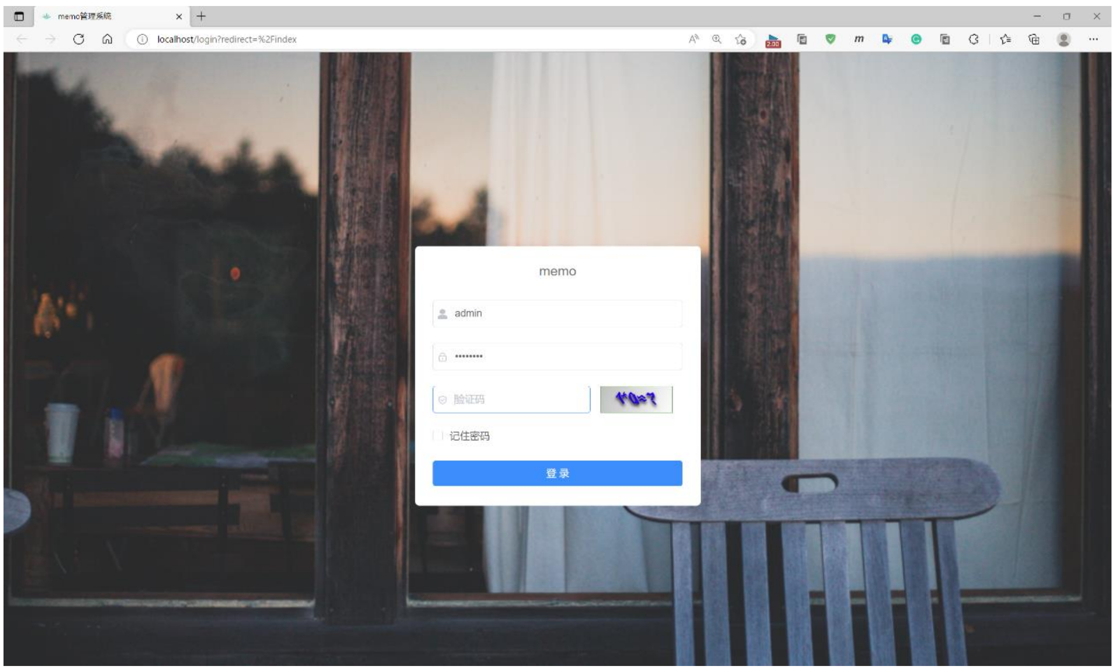
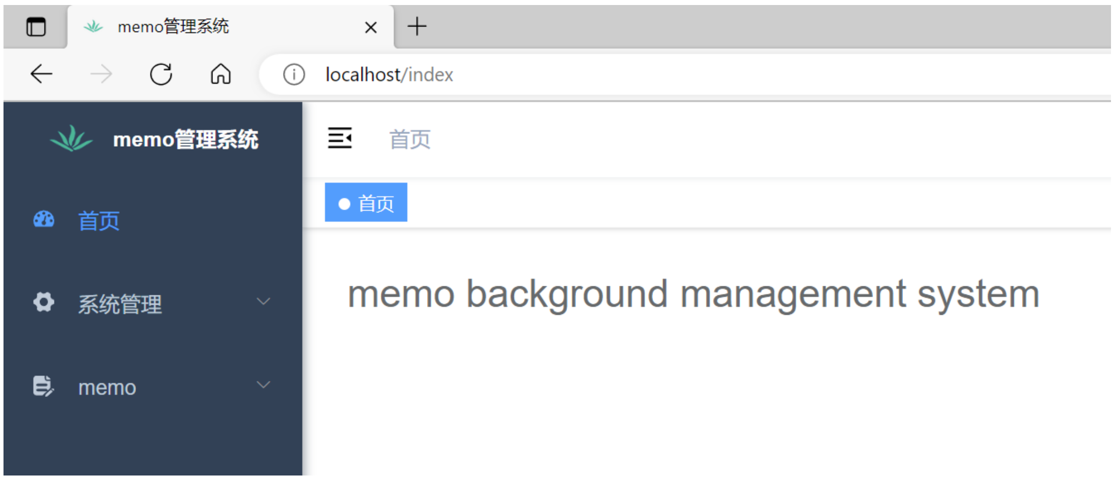
Todo list

This document is for the use of the todolist_memo system.

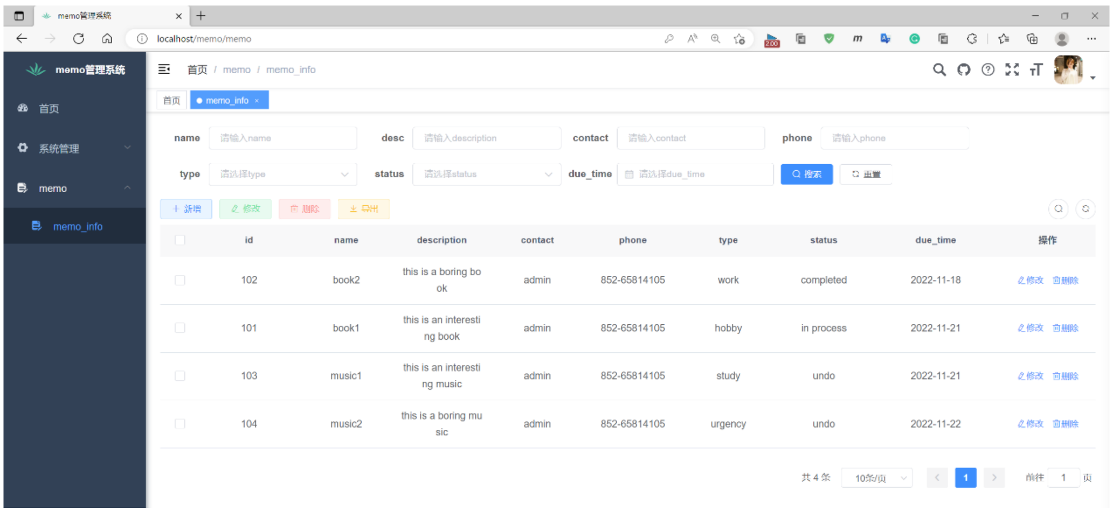
How to use



Enter the verification code to access the login page.



Login Page

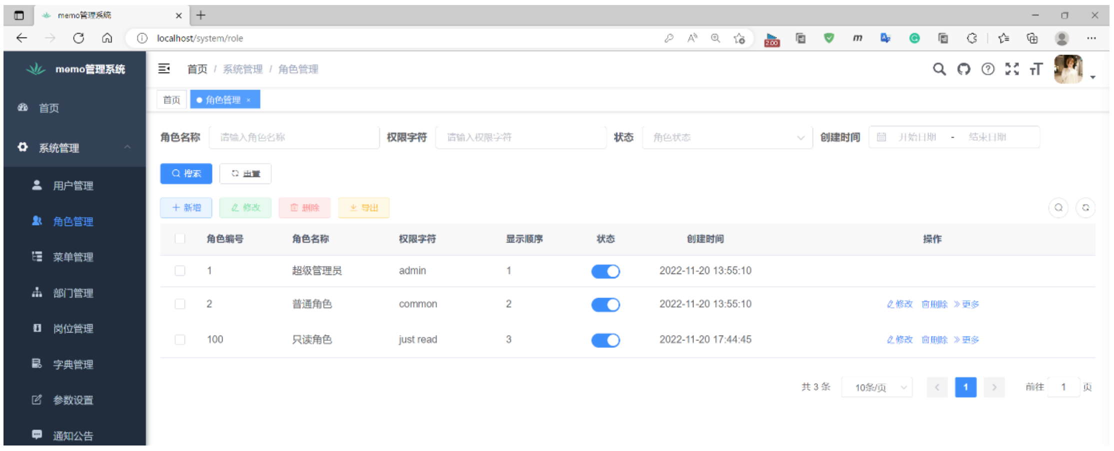


id	name	description	contact	phone	type	status	due_time	操作
102	book2	this is a boring book	admin	852-65814105	work	completed	2022-11-18	修改 删除
101	book1	this is an interesting book	admin	852-65814105	hobby	in process	2022-11-21	修改 删除
103	music1	this is an interesting music	admin	852-65814105	study	undo	2022-11-21	修改 删除
104	music2	this is a boring music	admin	852-65814105	urgency	undo	2022-11-22	修改 删除

共 4 条

Memo_list Page

All todolist functions are implemented in this page, and the display of due_time is sorted by date. (order by due_time in TodoMemoMapper.xml)

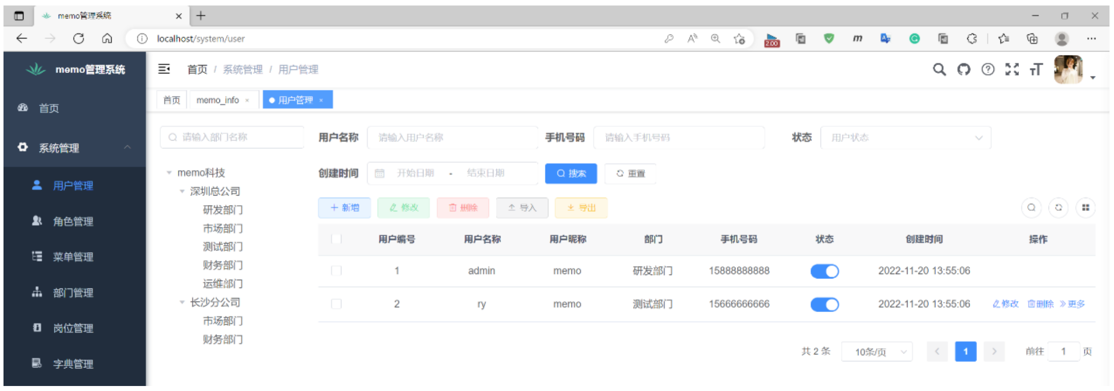


角色编号	角色名称	权限字符	显示顺序	状态	创建时间	操作
1	超级管理员	admin	1	●	2022-11-20 13:55:10	
2	普通角色	common	2	●	2022-11-20 13:55:10	修改 删除 更多
100	只读角色	just read	3	●	2022-11-20 17:44:45	修改 删除 更多

共 3 条

Role Management Page

You can create different types of roles that have different permissions.



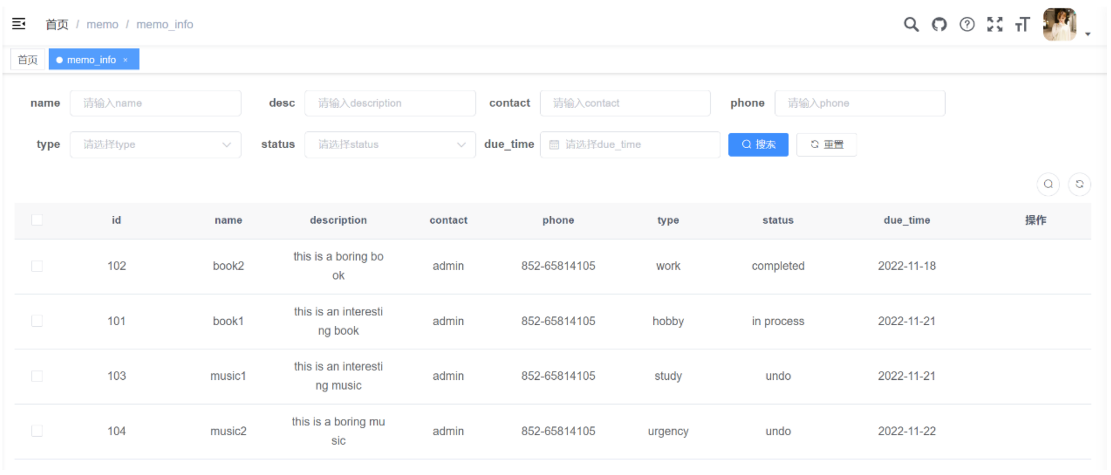
- memo科技
  - 深圳总公司
    - 研发部门
    - 市场部门
    - 测试部门
    - 财务部门
    - 运维部门
  - 长沙分公司
    - 市场部门
    - 财务部门

用户编号	用户名称	用户昵称	部门	手机号码	状态	创建时间	操作
1	admin	memo	研发部门	15888888888	●	2022-11-20 13:55:06	
2	ry	memo	测试部门	15666666666	●	2022-11-20 13:55:06	修改 删除 更多

共 2 条

User Management Page

You can add different users that belong to the specified role, The read-only role can only view the memo and cannot modify the todolist.

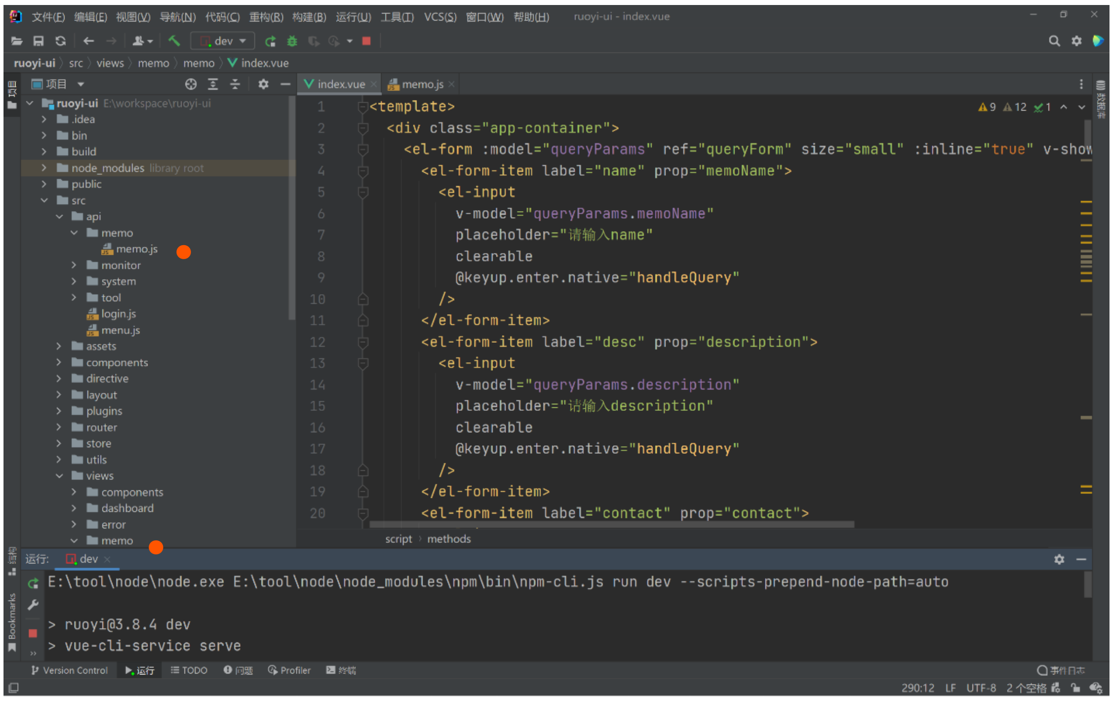


	id	name	description	contact	phone	type	status	due_time	操作
	102	book2	this is a boring book	admin	852-65814105	work	completed	2022-11-18	
	101	book1	this is an interesting book	admin	852-65814105	hobby	in process	2022-11-21	
	103	music1	this is an interesting music	admin	852-65814105	study	undo	2022-11-21	
	104	music2	this is a boring music	admin	852-65814105	urgency	undo	2022-11-22	

The Read-only Role Page

How build

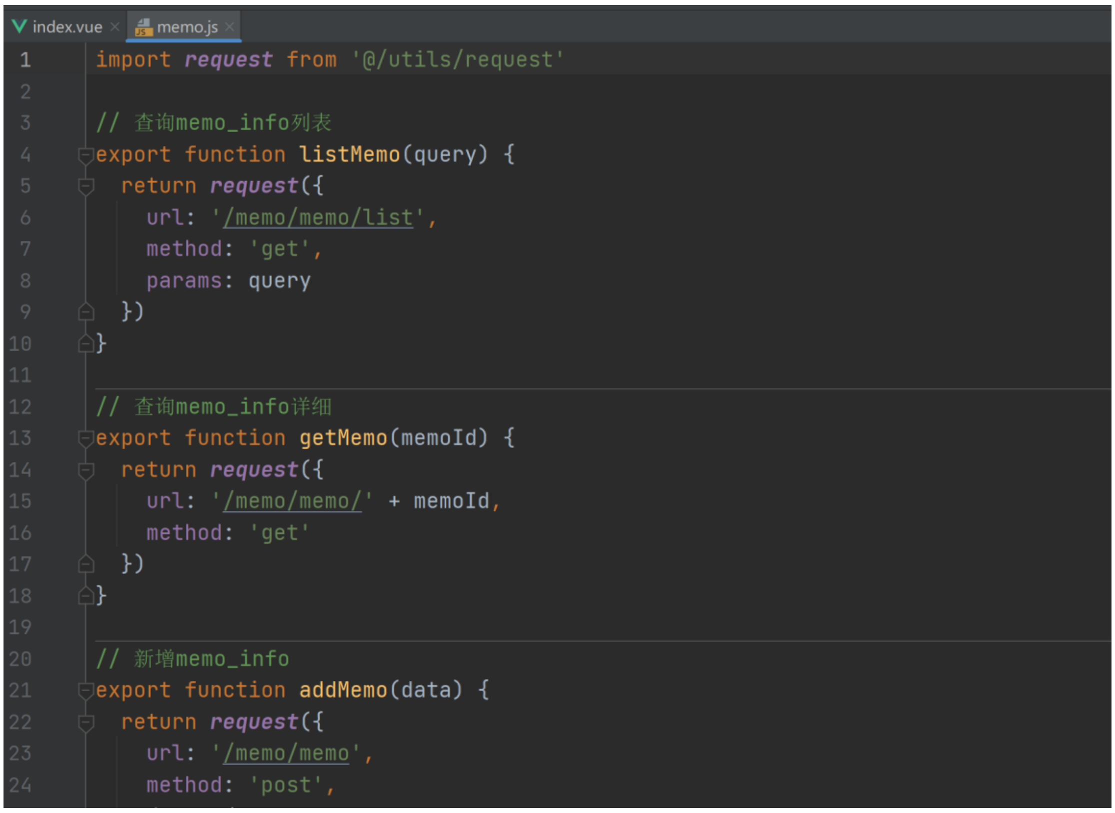
Front end part



```
<template>
  <div class="app-container">
    <el-form :model="queryParams" ref="queryForm" size="small" :inline="true" v-show
      <el-form-item label="name" prop="memoName">
        <el-input
          v-model="queryParams.memoName"
          placeholder="请输入name"
          clearable
          @keyup.enter.native="handleQuery"
        />
      </el-form-item>
      <el-form-item label="desc" prop="description">
        <el-input
          v-model="queryParams.description"
          placeholder="请输入description"
          clearable
          @keyup.enter.native="handleQuery"
        />
      </el-form-item>
      <el-form-item label="contact" prop="contact">
```

```
E:\tool\node\node.exe E:\tool\node\node_modules\npm\bin\npm-cli.js run dev --scripts-prepend-node-path=auto

> ruoyi@3.8.4 dev
> vue-cli-service serve
```

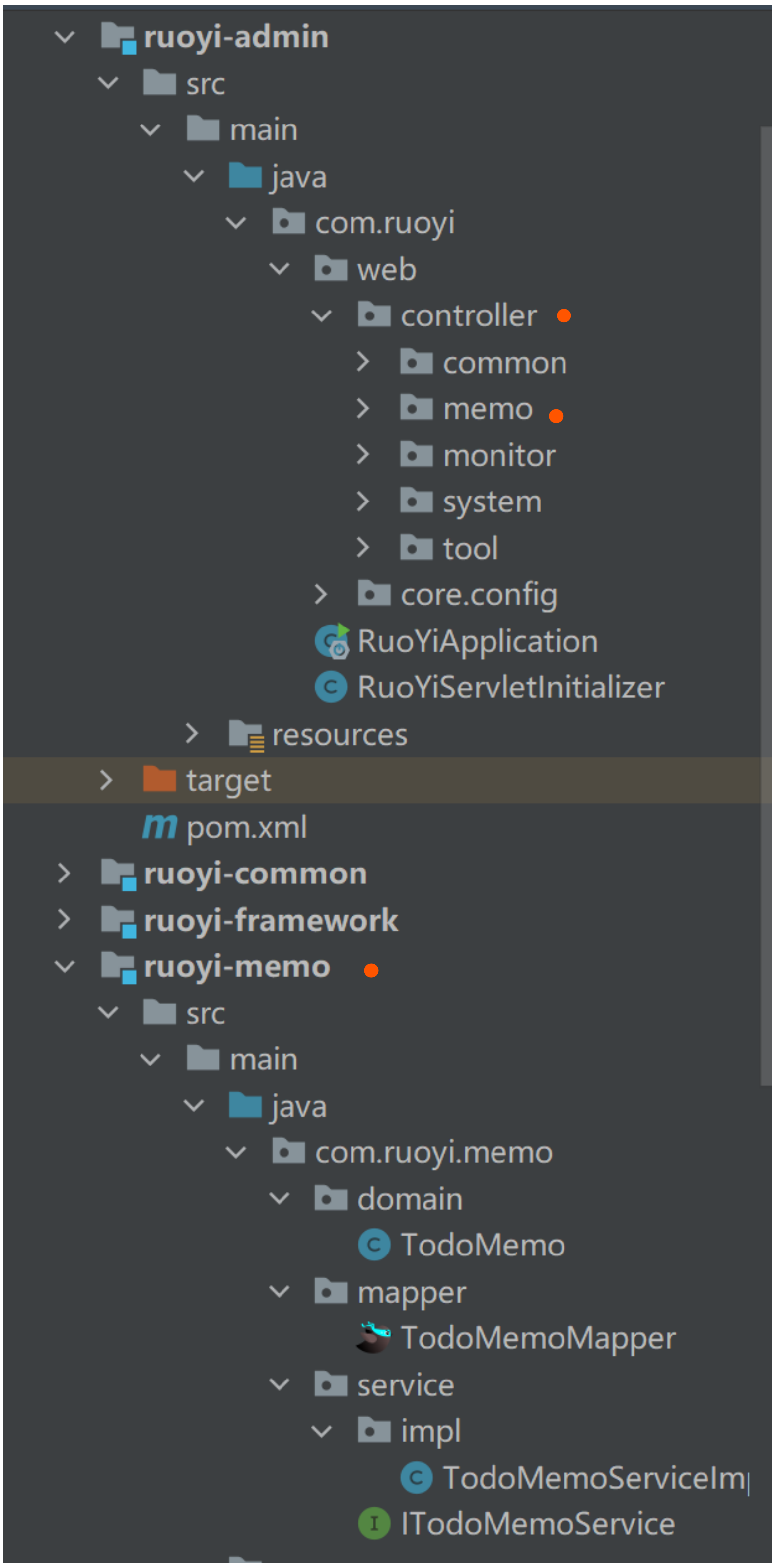
File directory and page preparation.



```
import request from '@/utils/request'

// 查询memo_info列表
export function listMemo(query) {
  return request({
    url: '/memo/memo/list',
    method: 'get',
    params: query
  })
}

// 查询memo_info详细
export function getMemo(memoId) {
  return request({
    url: '/memo/memo/' + memoId,
    method: 'get'
  })
}

// 新增memo_info
export function addMemo(data) {
  return request({
    url: '/memo/memo',
    method: 'post',
```

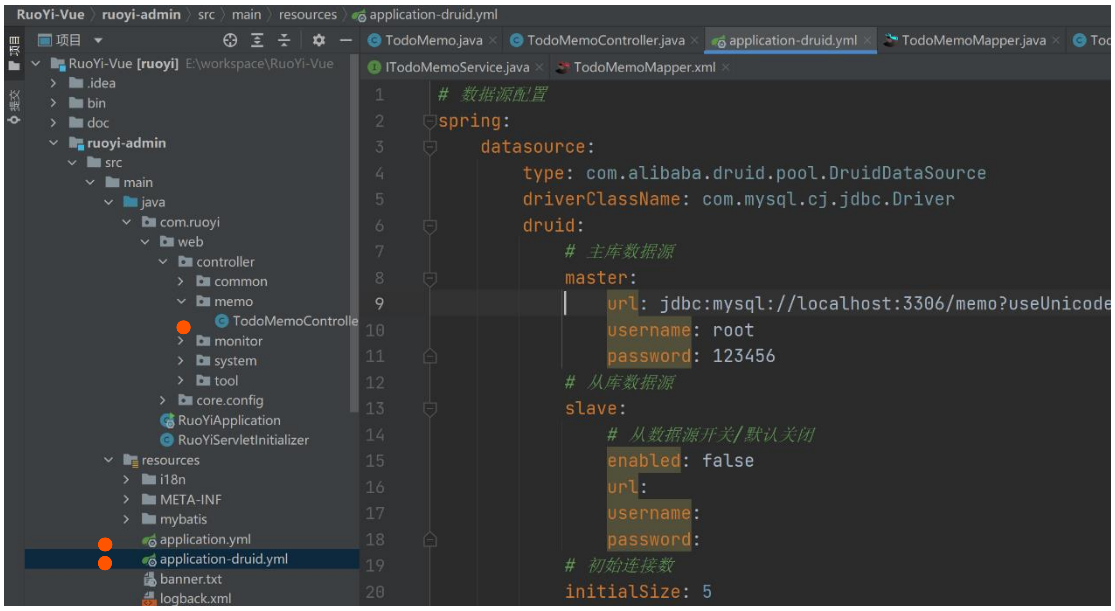
Mapping of the request path.

Back end part



```
ruoyi-admin
  src
    main
      java
        com.ruoyi
          web
            controller
              common
              memo
              monitor
              system
              tool
            core.config
            RuoYiApplication
            RuoYiServletInitializer
      resources
  target
  pom.xml
ruoyi-common
ruoyi-framework
ruoyi-memo
  src
    main
      java
        com.ruoyi.memo
          domain
            TodoMemo
          mapper
            TodoMemoMapper
          service
            impl
              TodoMemoServiceIm
            ITodoMemoService
```

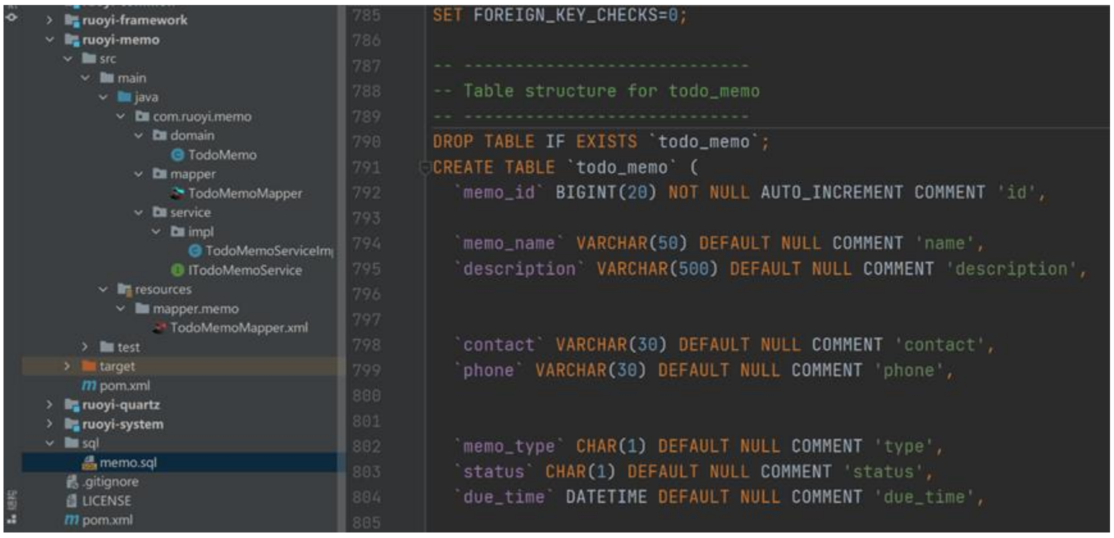
The main API implementation directory, this project relies on the ruoyi-framework for development. This framework helps us to implement the system management module, which includes user management, role management and other modules.



```
# 数据源配置
spring:
    datasource:
        type: com.alibaba.druid.pool.DruidDataSource
        driverClassName: com.mysql.cj.jdbc.Driver
        druid:
            # 主库数据源
            master:
                url: jdbc:mysql://localhost:3306/memo?useUnicode
                username: root
                password: 123456
            # 从库数据源
            slave:
                # 从数据源开关/默认关闭
                enabled: false
                url:
                username:
                password:
            # 初始连接数
            initialSize: 5
```

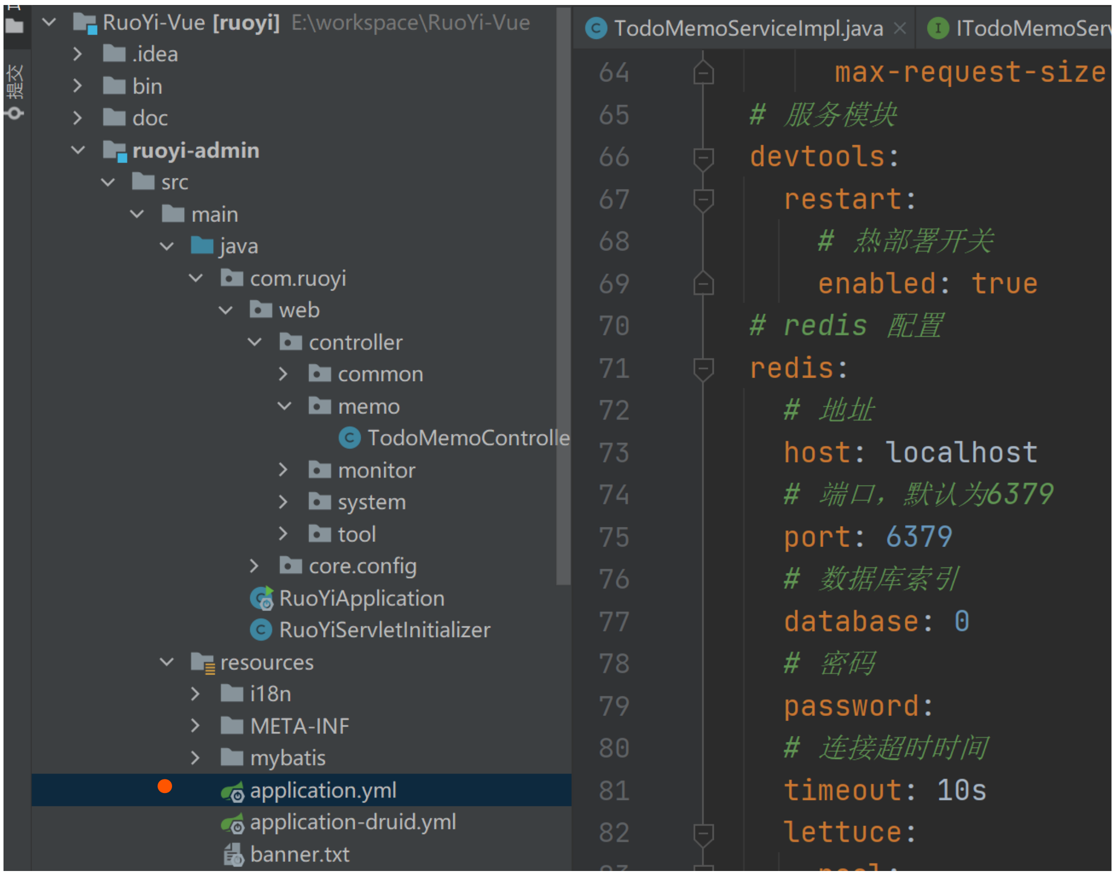
Database Configuration

The url links to the specific database that was created, username, password set by yourself



```
SET FOREIGN_KEY_CHECKS=0;

-- ----------------------------
-- Table structure for todo_memo
-- ----------------------------
DROP TABLE IF EXISTS `todo_memo`;
CREATE TABLE `todo_memo` (
  `memo_id` BIGINT(20) NOT NULL AUTO_INCREMENT COMMENT 'id',

  `memo_name` VARCHAR(50) DEFAULT NULL COMMENT 'name',
  `description` VARCHAR(500) DEFAULT NULL COMMENT 'description',


  `contact` VARCHAR(30) DEFAULT NULL COMMENT 'contact',
  `phone` VARCHAR(30) DEFAULT NULL COMMENT 'phone',


  `memo_type` CHAR(1) DEFAULT NULL COMMENT 'type',
  `status` CHAR(1) DEFAULT NULL COMMENT 'status',
  `due_time` DATETIME DEFAULT NULL COMMENT 'due_time',
```

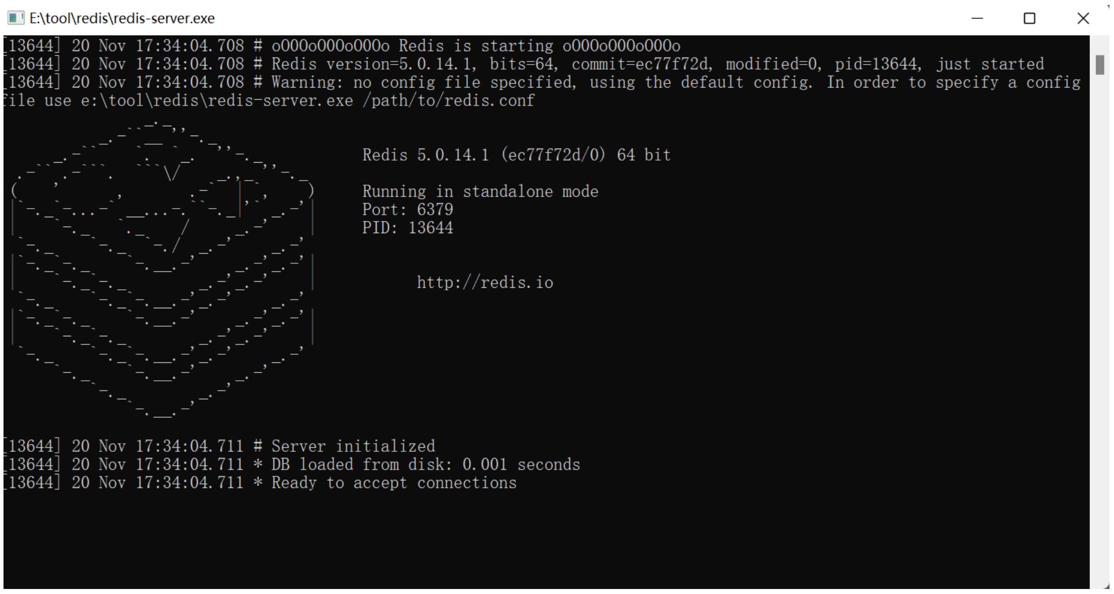
SQL statement to create the database.



```
RuoYi-Vue [ruoyi] E:\workspace\RuoYi-Vue
  .idea
  bin
  doc
  ruoyi-admin
    src
      main
        java
          com.ruoyi
            web
              controller
                common
                memo
                  TodoMemoControlle
                monitor
                system
                tool
              core.config
              RuoYiApplication
              RuoYiServletInitializer
        resources
          i18n
          META-INF
          mybatis
          application.yml
          application-druid.yml
          banner.txt
```

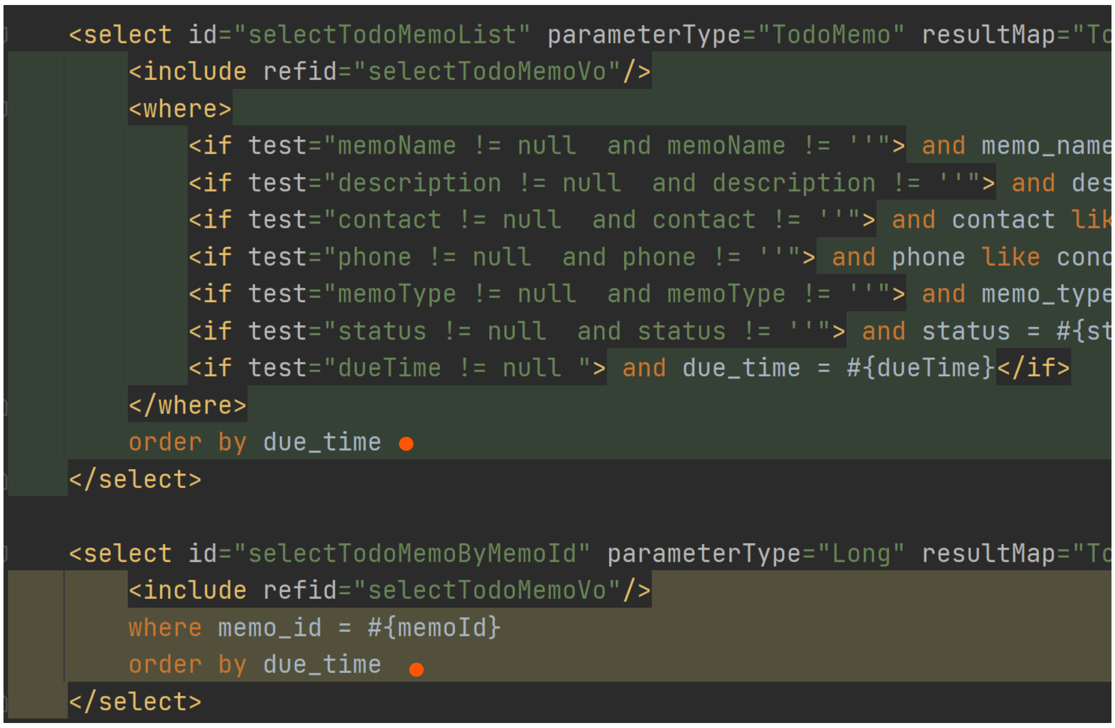
```yaml
      max-request-size
# 服务模块
devtools:
  restart:
    # 热部署开关
    enabled: true
# redis 配置
redis:
  # 地址
  host: localhost
  # 端口，默认为6379
  port: 6379
  # 数据库索引
  database: 0
  # 密码
  password:
  # 连接超时时间
  timeout: 10s
  lettuce:
```

To implement the CAPTCHA functionality, we used the redis database as a cache.



```
[13644] 20 Nov 17:34:04.708 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo
[13644] 20 Nov 17:34:04.708 # Redis version=5.0.14.1, bits=64, commit=ec77f72d, modified=0, pid=13644, just started
[13644] 20 Nov 17:34:04.708 # Warning: no config file specified, using the default config. In order to specify a config
file use e:\tool\redis\redis-server.exe /path/to/redis.conf

                    Redis 5.0.14.1 (ec77f72d/0) 64 bit

                    Running in standalone mode
                    Port: 6379
                    PID: 13644

                          http://redis.io

[13644] 20 Nov 17:34:04.711 # Server initialized
[13644] 20 Nov 17:34:04.711 * DB loaded from disk: 0.001 seconds
[13644] 20 Nov 17:34:04.711 * Ready to accept connections
```

Start the redis service.



```
<select id="selectTodoMemoList" parameterType="TodoMemo" resultMap="To
    <include refid="selectTodoMemoVo"/>
    <where>
        <if test="memoName != null  and memoName != ''"> and memo_name
        <if test="description != null  and description != ''"> and des
        <if test="contact != null  and contact != ''"> and contact lik
        <if test="phone != null  and phone != ''"> and phone like conc
        <if test="memoType != null  and memoType != ''"> and memo_type
        <if test="status != null  and status != ''"> and status = #{st
        <if test="dueTime != null "> and due_time = #{dueTime}</if>
    </where>
    order by due_time
</select>

<select id="selectTodoMemoByMemoId" parameterType="Long" resultMap="To
    <include refid="selectTodoMemoVo"/>
    where memo_id = #{memoId}
    order by due_time
</select>
```

The memo sorting was realized by the order of the due_time.