

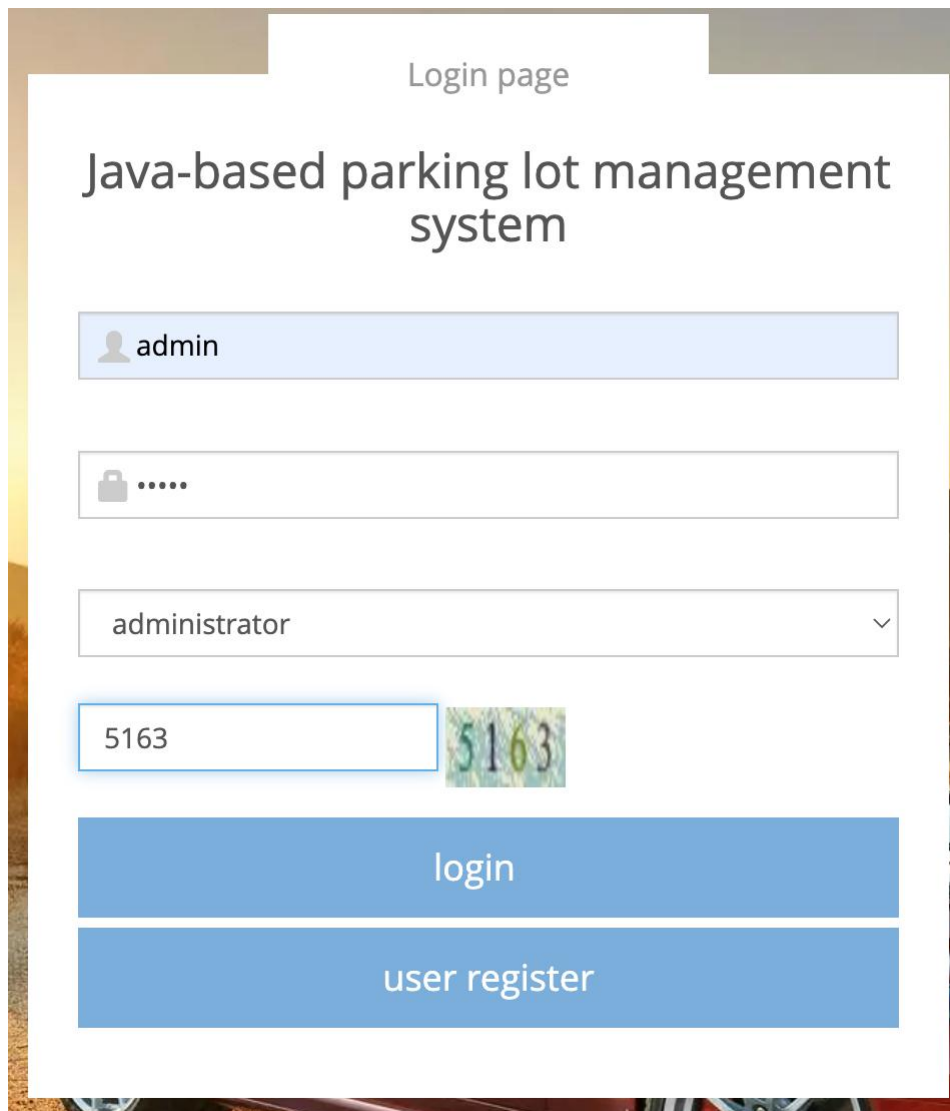
## Test Login

First, you need to enter account and password. Secpndly, you need to choose administrator and user. Then you need to enter a verification code.

administrator

account: admin

password: admin



The screenshot shows a web application interface for a "Java-based parking lot management system". The page is titled "Login page" at the top. It features several input fields and buttons. The first input field is for the account name, with a user icon and the text "admin". The second input field is for the password, with a lock icon and five dots. The third input field is a dropdown menu for the role, currently showing "administrator". The fourth input field is for a verification code, with the text "5163" and a corresponding image of the code "5163". Below these fields are two blue buttons: "login" and "user register".

Login page

Java-based parking lot management system

admin

.....

administrator

5163

5163

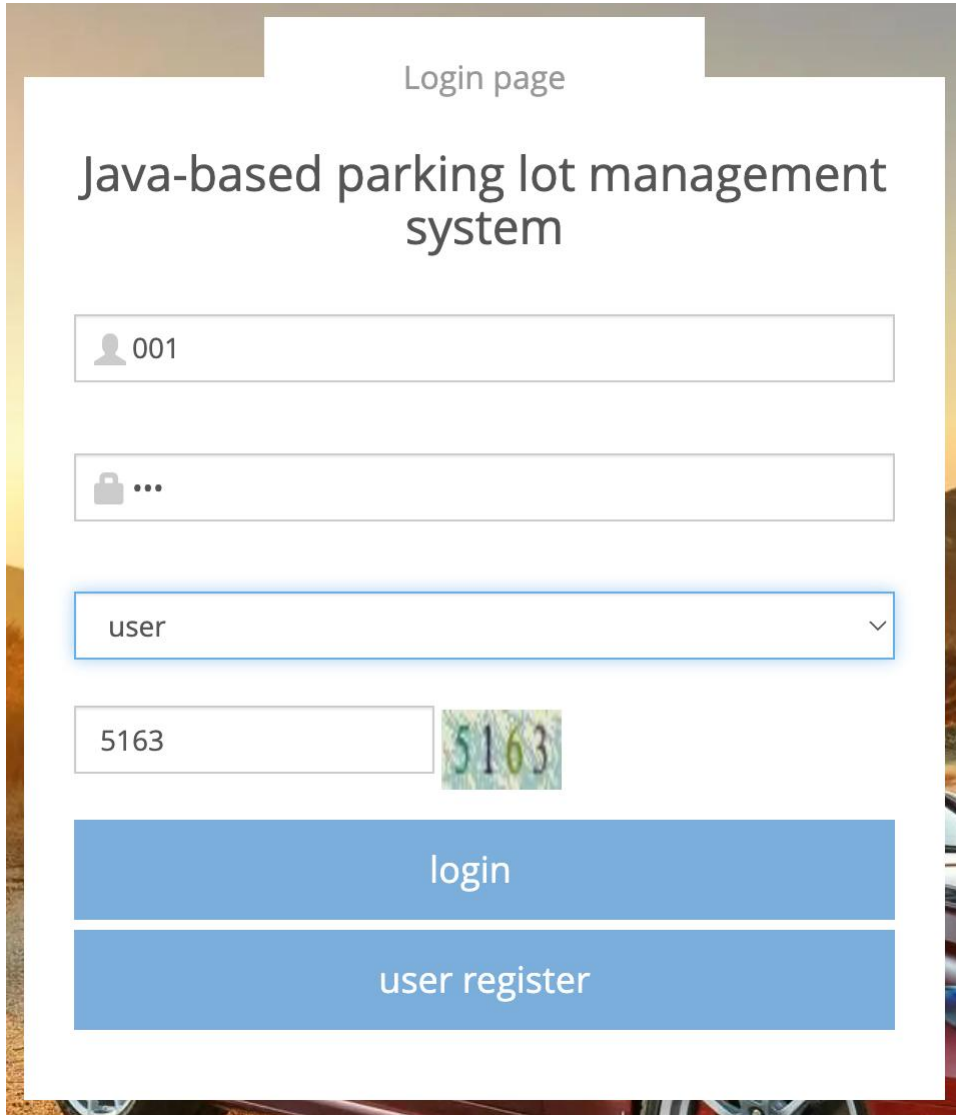
login

user register

user

account: 001

password: 001



The image shows a web application interface for a parking management system. At the top, it says "Login page". Below that, the title "Java-based parking lot management system" is displayed. There are three input fields: the first contains "001" with a user icon, the second is empty with a lock icon and three dots, and the third contains "user" with a dropdown arrow. Below these is a CAPTCHA field showing "5163" next to a distorted image of the same numbers. At the bottom, there are two blue buttons labeled "login" and "user register".

Login page

Java-based parking lot management system

001

...

user

5163

login

user register

This is a Java-based parking management system development project

This project is developed and implemented based on

Servlet, Mysql, Javascript and other back-end technologies, and uses Tomcat to run and deploy. The functions of this project mainly include parking space management, parking information management, parking lot management, parking lot management, etc.

Recharge management, feedback management, complaint management, user management, IC card management, administrator management, etc., the background system is divided into two kinds of users, ordinary users and administrator users, we do the rights management, the administrator's rights are greater than the user

The permissions you have.

Project code:

1, src folder is the backend provided by the Java language development interface service code

- \* dao is a class that is written to interact with the database, and the business table of the database is added, deleted, and checked

- \* The entity package contains some entity classes used by the Parking management system, including User, Admins, Parking, ParkingLotInformation, and so on

- \* service package is business code, used to write some code logic when business processing
- \* The servlet package is the interface service code, which is the entrance of the back-end service, and it writes the service interface that the entire project needs to provide to the front-end, such as the interface for querying parking information, login, and registration
- \* The test package contains the test code, which uses JUnit and Mock to test the code
- \* util package is some commonly used tool classes, such as uploading files and so on

2, the WebRoot folder is the web front-end required files, including HTML, JSP files

3. In the database folder is the sql file of the project, park.sql

Running/testing instructions:

This project needs to be deployed on tomcat, and the environment needs to have jdk1.8, mysql, and tomcat installed. You need to run the park.sql file before running the project. After the execution is complete, you need to modify the JDBC connection parameters until they are

correct

Use Tomcat to run the code,

[http://localhost:8181/jspmvc\\_war/login.jsp](http://localhost:8181/jspmvc_war/login.jsp)