

Method name:
login

Class name: information
regarding login.

ID: 01

clients :- users

Associated username: login, home

Description of Responsibilities:

verifying the login information
provided by the user.

Arguments Received: It gets responses about the
users login details, including the
forget password feature.

Type of value returned:

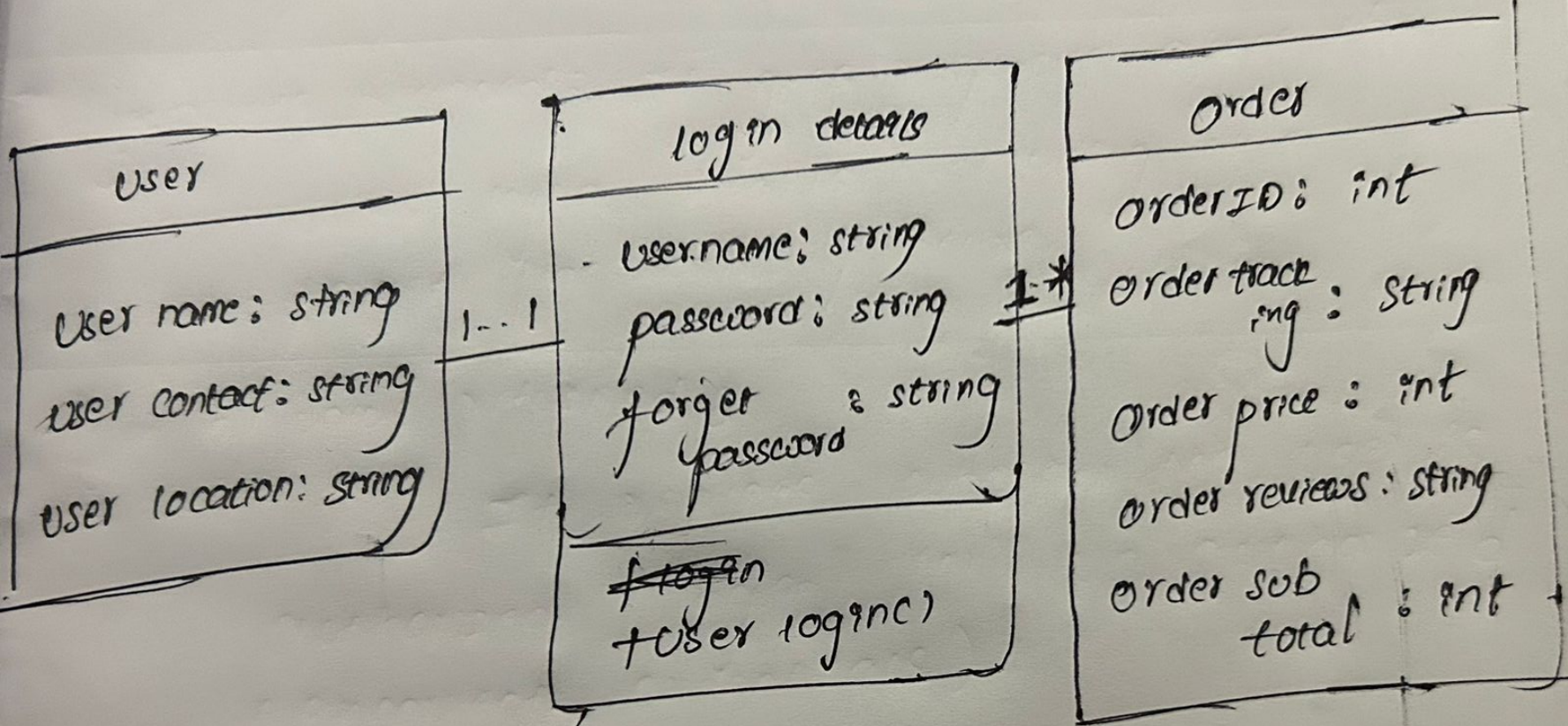
It replies whether the user is
authorized (login) (or) un-authorized
(incorrect credentials).

pre-condition:

The pre-condition is the user
login information.

post-condition:

The post-condition is verification
process done with database regarding
user login information.



<< invariant >>
{ Driven password = password.password() }

//This is a method for one use case.

```
public class Login{

    private static final long serialVersionUID = 1 L;
    private String username;
    private String password;

    // Pre condition for checking the details from the db.
    public boolean validate(LoginBean loginBean) throws ClassNotFoundException {
        boolean status = false;

        Class.forName("com.mysql.jdbc.Driver");

        try {Connection connection = DriverManager
            .getConnection("jdbc:mysql://localhost:3306/mysql_database?useSSL=false", "root", "root");

            // post condition for checking the details from the db.

            PreparedStatement preparedStatement = connection
                .prepareStatement("select * from login where username = ? and password = ? ") {
            preparedStatement.setString(1, loginBean.getUsername());
            preparedStatement.setString(2, loginBean.getPassword());

            System.out.println(preparedStatement);
            ResultSet rs = preparedStatement.executeQuery();
            status = rs.next();

        } catch (SQLException e) {
            // process sql exception
            printSQLException(e);
        }
        return status;
    }
}
```

//invariant

```
private void printSQLException(SQLException ex) {
    for (Throwable e: ex) {
        if (e instanceof SQLException) {
            e.printStackTrace(System.err);
            System.err.println("SQLState: " + ((SQLException) e).getSQLState());
            System.err.println("Error Code: " + ((SQLException) e).getErrorCode());
            System.err.println("Message: " + e.getMessage());
            Throwable t = ex.getCause();
            while (t != null) {
                System.out.println("Cause: " + t);
                t = t.getCause();
            }
        }
    }
}
```