

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - FoodAddaToTraineeApplication.java
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - com.infy.utility
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
- mvnw
- mvnw.cmd
- pom.xml
- RuleSet2015.xml

FoodAddaToTraineeApplication.java

```
1 package com.infy;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5
6 @SpringBootApplication
7 public class FoodAddaToTraineeApplication {
8
9     //Main class
10    public static void main(String[] args) {
11        SpringApplication.run(FoodAddaToTraineeApplication.class, args);
12    }
13
14 }
15 }
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:32 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

UserAPI.java

```
1 package com.infy.api;
2
3 import java.security.NoSuchAlgorithmException;
4
5 import javax.validation.Valid;
6
7 import org.springframework.beans.factory.annotation.Autowired;
8 import org.springframework.core.env.Environment;
9 import org.springframework.http.HttpStatus;
10 import org.springframework.http.ResponseEntity;
11 import org.springframework.validation.annotation.Validated;
12 import org.springframework.web.bind.annotation.CrossOrigin;
13 import org.springframework.web.bind.annotation.PostMapping;
14 import org.springframework.web.bind.annotation.RequestBody;
15 import org.springframework.web.bind.annotation.RequestMapping;
16 import org.springframework.web.bind.annotation.RestController;
17
18 import com.infy.exception.FoodAddaException;
19 import com.infy.model.Login;
20 import com.infy.model.SuccessInfo;
21 import com.infy.model.Users;
22 import com.infy.service.UserService;
23 import com.infy.utility.HashingUtility;
24 @CrossOrigin
25 @RestController
26 @RequestMapping("userAPI")
27 @Validated
28 public class UserAPI {
29
30     @Autowired
31     private UserService userService;
32
33     @Autowired
34     private Environment environment;
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:33 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - com.infy.api
 - SearchAPI.java
 - UserAPI.java
 - UserAPI
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - com.infy.utility
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
- mvnw
- mvnw.cmd
- pom.xml
- RuleSet2015.xml

UserAPI.java

```
27 @Validated
28 public class UserAPI {
29
30     @Autowired
31     private UserService userService;
32
33     @Autowired
34     private Environment environment;
35
36     @PostMapping(value = "/login")
37     public ResponseEntity<Users> loginUser(@Valid @RequestBody Login login) throws FoodAddaException {
38         Users user = null;
39         try {
40             user = userService.loginUser(login.getContactNumber(), HashingUtility.getHashValue(login.getPassword()));
41         } catch (NoSuchAlgorithmException e) {
42             throw new FoodAddaException(e.getMessage());
43         }
44
45         return new ResponseEntity<>(user, HttpStatus.OK);
46     }
47
48     @PostMapping(value = "/register")
49     public ResponseEntity<SuccessInfo> registerUser(@Valid @RequestBody Users user) throws FoodAddaException {
50         String successMessage = null;
51         SuccessInfo response = new SuccessInfo();
52         try {
53             user.setPassword(HashingUtility.getHashValue(user.getPassword()));
54             userService.registerUser(user);
55             successMessage = environment.getProperty("UserAPI.REGISTER_USER_SUCCESS1") +
56                             environment.getProperty("UserAPI.REGISTER_USER_SUCCESS2");
57             response = new SuccessInfo();
58             response.setSuccessCode(HttpStatus.CREATED.value());
59             response.setSuccessMessage(successMessage);
60         } catch (NoSuchAlgorithmException e) {
```

com.infy.api.UserAPI.java - Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_Group-23_Server_side/src/main/java

Type here to search

22°C AQI 62 ENG 22:33 24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - com.infy
 - com.infy.api
 - SearchAPI.java
 - UserAPI.java
 - UserAPI
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - com.infy.utility
 - src/main/resources
 - src/test/java
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

UserAPI.java

```
35  
36 @PostMapping(value = "/login")  
37 public ResponseEntity<Users> loginUser(@Valid @RequestBody Login login) throws FoodAddaException {  
38     Users user = null;  
39     try {  
40         user = userService.loginUser(login.getContactNumber(), HashingUtility.getHashValue(login.getPassword()));  
41     } catch (NoSuchAlgorithmException e) {  
42         throw new FoodAddaException(e.getMessage());  
43     }  
44  
45     return new ResponseEntity<>(user, HttpStatus.OK);  
46 }  
47  
48 @PostMapping(value = "/register")  
49 public ResponseEntity<SuccessInfo> registerUser(@Valid @RequestBody Users user) throws FoodAddaException {  
50     String successMessage = null;  
51     SuccessInfo response = new SuccessInfo();  
52     try {  
53         user.setPassword(HashingUtility.getHashValue(user.getPassword()));  
54         userService.registerUser(user);  
55         successMessage = environment.getProperty("UserAPI.REGISTER_USER_SUCCESS1") +  
56             environment.getProperty("UserAPI.REGISTER_USER_SUCCESS2");  
57         response = new SuccessInfo();  
58         response.setSuccessCode(HttpStatus.CREATED.value());  
59         response.setSuccessMessage(successMessage);  
60     } catch (NoSuchAlgorithmException e) {  
61         throw new FoodAddaException(e.getMessage());  
62     }  
63     return new ResponseEntity<>(response,HttpStatus.CREATED);  
64 }  
65  
66 }  
67 }  
68 }
```

Writable

Smart Insert

40:84:1351



Type here to search



22°C AQI 62

ENG 22:33
24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - SearchDAO.java
 - SearchDAOImpl.java
 - UserDAO.java
 - UserDAO
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - com.infy.utility
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

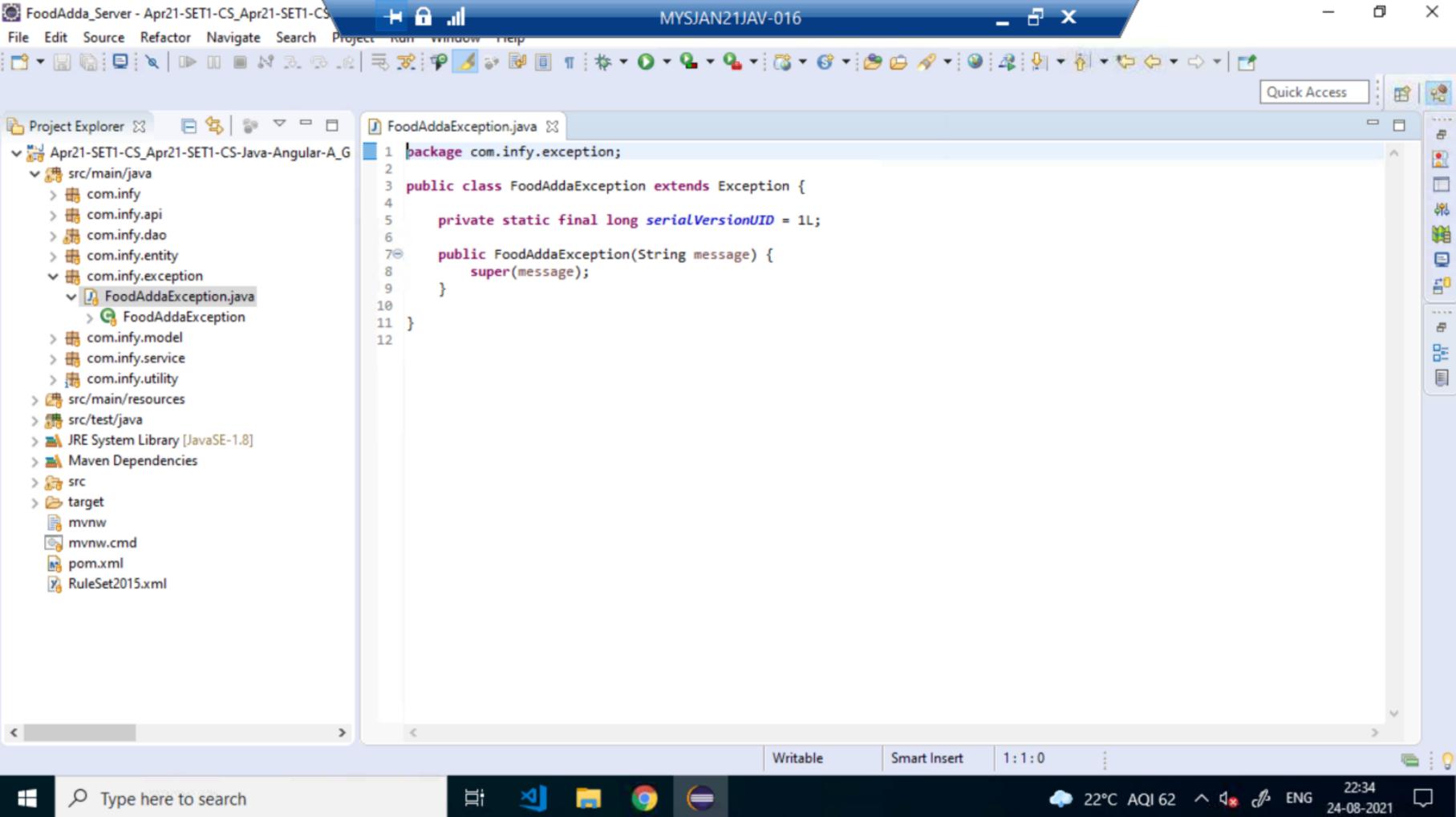
UserDAO.java

```
1 package com.infy.dao;
2
3 import java.util.Optional;
4
5 import org.springframework.data.repository.CrudRepository;
6
7 import com.infy.entity.UsersEntity;
8
9 public interface UserDAO extends CrudRepository<UsersEntity, Integer>{
10
11     Optional<UsersEntity> findByContactNumber(String contactNumber);
12
13 }
14
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:34 24-08-2021



The screenshot shows a Java code editor with the file `Login.java` open. The code defines a class `Login` with attributes `contactNumber` and `password`, both annotated with `@NotNull` and `@Pattern`. The `contactNumber` field has a regular expression constraint of `[6-9][0-9]{9}`. The code includes getters and setters for both fields.

```
1 package com.infy.model;
2
3 import javax.validation.constraints.NotNull;
4 import javax.validation.constraints.Pattern;
5
6 public class Login {
7
8     @NotNull(message = "UserValidator.INVALID_CONTACT_NUMBER_FORMAT")
9     @Pattern(regexp = "[6-9][0-9]{9}", message = "UserValidator.INVALID_CONTACT_NUMBER_FORMAT")
10    private String contactNumber;
11
12    @NotNull(message = "UserValidator.INVALID_PASSWORD_FORMAT")
13    private String password;
14
15    public String getContactNumber() {
16        return contactNumber;
17    }
18    public void setContactNumber(String contactNumber) {
19        this.contactNumber = contactNumber;
20    }
21    public String getPassword() {
22        return password;
23    }
24    public void setPassword(String password) {
25        this.password = password;
26    }
27
28
29
30 }
```



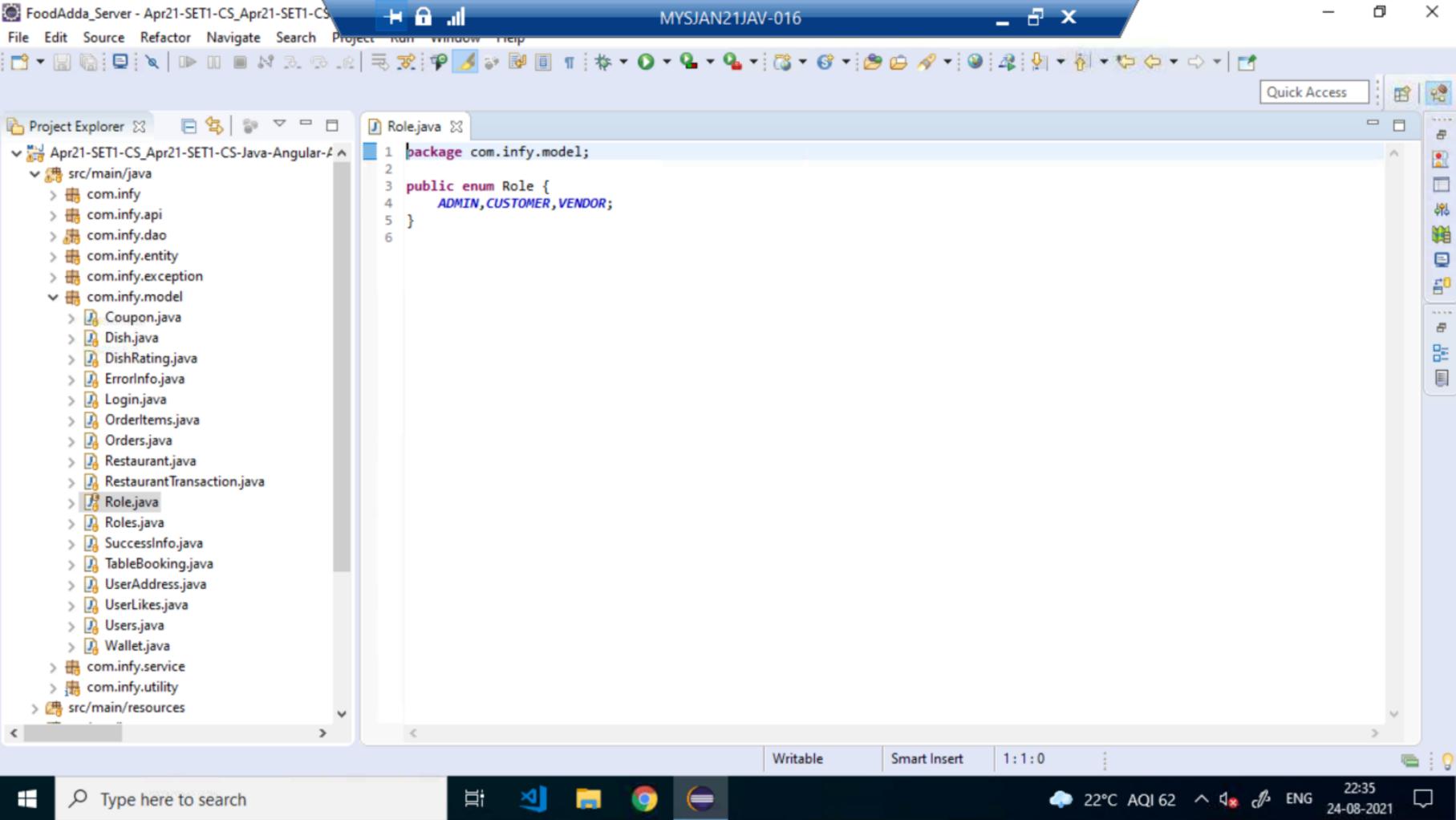
Type here to search



22°C

AQI 62

22:35
24-08-2021



FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - Coupon.java
 - Dish.java
 - DishRating.java
 - ErrorInfo.java
 - Login.java
 - OrderItems.java
 - Orders.java
 - Restaurant.java
 - RestaurantTransaction.java
 - Role.java
 - Roles.java
 - SuccessInfo.java
 - TableBooking.java
 - UserAddress.java
 - UserLikes.java
 - Users.java
 - Wallet.java
 - com.infy.service
 - com.infy.utility
- src/main/resources

Roles.java

```
1 package com.infy.model;
2
3 public class Roles {
4     private Integer roleId;
5     private Role roleType;
6     public Integer getRoleId() {
7         return roleId;
8     }
9     public void setRoleId(Integer roleId) {
10        this.roleId = roleId;
11    }
12    public Role getRoleType() {
13        return roleType;
14    }
15    public void setRoleType(Role role) {
16        this.roleType = role;
17    }
18
19
20
21
22 }
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:35 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - Coupon.java
 - Dish.java
 - DishRating.java
 - ErrorInfo.java
 - Login.java
 - OrderItems.java
 - Orders.java
 - Restaurant.java
 - RestaurantTransaction.java
 - Role.java
 - Roles.java
 - SuccessInfo.java
 - TableBooking.java
 - UserAddress.java
 - UserLikes.java
 - Users.java
 - Wallet.java
 - com.infy.service
 - com.infy.utility
- src/main/resources

SuccessInfo.java

```
1 package com.infy.model;
2
3 public class SuccessInfo {
4     private String successMessage;
5     private Integer successCode;
6     public String getSuccessMessage() {
7         return successMessage;
8     }
9     public void setSuccessMessage(String successMessage) {
10        this.successMessage = successMessage;
11    }
12    public Integer getSuccessCode() {
13        return successCode;
14    }
15    public void setSuccessCode(Integer successCode) {
16        this.successCode = successCode;
17    }
18
19 }
20
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:35 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

UserAddress.java

```
1 package com.infy.model;
2
3
4 import javax.validation.constraints.Pattern;
5
6 public class UserAddress {
7     private Integer userAddressId;
8     @Pattern(regexp = "[a-zA-Z0-9\\s]+$", message = "UserAddress.NULL_ADDRESS_NAME")
9     private String userAddressName;
10    @Pattern(regexp = "[a-zA-Z0-9\\s]+$", message = "UserAddress.NULL_ADDRESS_LINE_1")
11    private String addressLine1;
12    @Pattern(regexp = "[a-zA-Z0-9\\s]+$", message = "UserAddress.NULL_ADDRESS_LINE_2")
13    private String addressLine2;
14    @Pattern(regexp = "[A-Za-z]+(\\s[A-Za-z ]+)?$", message = "UserAddress.INVALID_AREA_FORMAT")
15    private String area;
16    @Pattern(regexp = "[A-Za-z]+$", message = "UserAddress.INVALID_CITY_FORMAT")
17    private String city;
18    private String userState;
19    @Pattern(regexp = "^[1-9]{1}[0-9]{2}\\s{0,1}[0-9]{3}$", message = "UserAddress.INVALID_PIN_CODE")
20    private String pincode;
21    public Integer getUserAddressId() {
22        return userAddressId;
23    }
24    public void setUserAddressId(Integer userAddressId) {
25        this.userAddressId = userAddressId;
26    }
27    public String getUserAddressName() {
28        return userAddressName;
29    }
30    public void setUserAddressName(String userAddressName) {
31        this.userAddressName = userAddressName;
32    }
33    public String getAddressLine1() {
34        return addressLine1;
35    }
36}
```

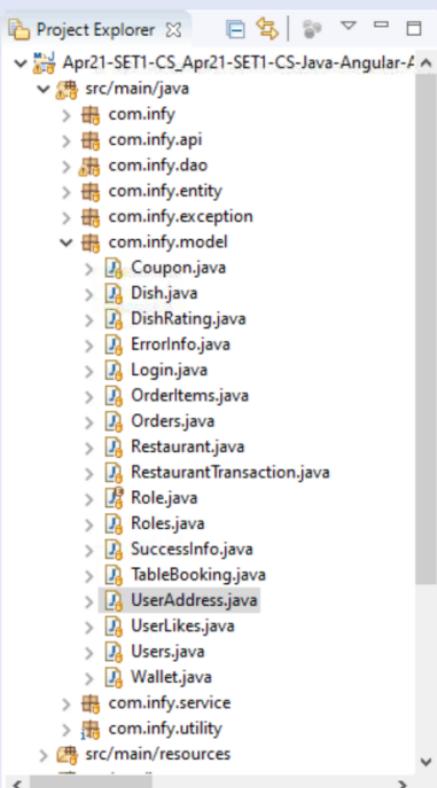
Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:36 24-08-2021



Quick Access



UserAddress.java

```
20
21
22
23
24
25
26
27
28
29
30 public void setUserAddressName(String userAddressName) {
31     this.userAddressName = userAddressName;
32 }
33 public String getAddressLine1() {
34     return addressLine1;
35 }
36 public void setAddressLine1(String addressLine1) {
37     this.addressLine1 = addressLine1;
38 }
39 public String getAddressLine2() {
40     return addressLine2;
41 }
42 public void setAddressLine2(String addressLine2) {
43     this.addressLine2 = addressLine2;
44 }
45 public String getArea() {
46     return area;
47 }
48 public void setArea(String area) {
49     this.area = area;
50 }
51 public String getCity() {
52     return city;
53 }
54 public void setCity(String city) {
55     this.city = city;
56 }
57 public String getUserState() {
58     return userState;
59 }
60 public void setUserState(String userState) {
61     this.userState = userState;
62 }
```

Writable

Smart Insert

1:1:0



Type here to search



22°C AQI 62

22:36
24-08-2021

The screenshot shows a Java code editor with the file `UserAddress.java` open. The code defines a class with various getters and setters for address-related properties. The editor interface includes a Project Explorer on the left, a toolbar at the top, and a status bar at the bottom. A line number 69 is highlighted in the code.

```
40         return addressLine2;
41     }
42     public void setAddressLine2(String addressLine2) {
43         this.addressLine2 = addressLine2;
44     }
45     public String getArea() {
46         return area;
47     }
48     public void setArea(String area) {
49         this.area = area;
50     }
51     public String getCity() {
52         return city;
53     }
54     public void setCity(String city) {
55         this.city = city;
56     }
57     public String getUserState() {
58         return userState;
59     }
60     public void setUserState(String userState) {
61         this.userState = userState;
62     }
63     public String getPincode() {
64         return pincode;
65     }
66     public void setPincode(String pincode) {
67         this.pincode = pincode;
68     }
69
70
71
72 }
```

Line: 69

Writable

Smart Insert

1:1:0



Type here to search



22°C AQI 62

22:36
24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A
- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - Coupon.java
 - Dish.java
 - DishRating.java
 - ErrorInfo.java
 - Login.java
 - OrderItems.java
 - Orders.java
 - Restaurant.java
 - RestaurantTransaction.java
 - Role.java
 - Roles.java
 - SuccessInfo.java
 - TableBooking.java
 - UserAddress.java
 - UserLikes.java
 - Users.java
 - Wallet.java
 - com.infy.service
 - com.infy.utility
- src/main/resources

*Users.java

```
1 package com.infy.model;
2
3 import java.util.List;
4
5 import javax.validation.constraints.Pattern;
6
7 //user model class
8 public class Users {
9
10     private Integer userId;
11     @Pattern(regexp = "^( [A-Za-z]{2,})( [A-Za-z]*){0,2}$", message = "UserValidator.INVALID_USERNAME_FORMAT")
12     private String userName;
13     @Pattern(regexp = "[6789][0-9]{9}", message = "UserValidator.INVALID_CONTACT_NUMBER_FORMAT")
14     private String contactNumber;
15     @Pattern(regexp = "(.+@(.+)$", message = "UserValidator.INVALID_EMAIL_ID_FORMAT")
16     private String emailId;
17     @Pattern(regexp = "^(?=.*[0-9])(?=.*[a-z])(?=.*[A-Z])(?=.*[@#$%^&-+=()])(?=\s+$).{7,20}$"
18             , message = "UserValidator.INVALID_PASSWORD_FORMAT")
19     private String password;
20     private List<Roles> roles;
21     private List<Restaurant> restaurants;
22     private List<UserAddress> addressList;
23     private Wallet wallet;
24     private List<UserLikes> userLikesList;
25     private List<Orders> ordersList;
26
27     public Integer getUserId() {
28         return userId;
29     }
30     public void setUserId(Integer userId) {
31         this.userId = userId;
32     }
33     public String getUserName() {
34         return userName;
35     }
```

Writable

Smart Insert

18 : 13 : 642



Type here to search



22°C AQI 62

ENG 22:36
24-08-2021

The screenshot shows an IDE interface with the following details:

- Title Bar:** FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS, MYSJAN21JAV-016
- Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help
- Toolbar:** Standard icons for file operations, search, and navigation.
- Quick Access:** A small window in the top right corner.
- Project Explorer:** Shows the project structure under "Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A".
 - src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - Coupon.java
 - Dish.java
 - DishRating.java
 - ErrorInfo.java
 - Login.java
 - OrderItems.java
 - Orders.java
 - Restaurant.java
 - RestaurantTransaction.java
 - Role.java
 - Roles.java
 - SuccessInfo.java
 - TableBooking.java
 - UserAddress.java
 - UserLikes.java
 - Users.java
 - Wallet.java
 - com.infy.service
 - com.infy.utility
 - src/main/resources
- Code Editor:** The main window displays the code for the `Users.java` class.

```
28     return userId;
29 }
30 public void setUserId(Integer userId) {
31     this.userId = userId;
32 }
33 public String getUserName() {
34     return userName;
35 }
36 public void setUserName(String userName) {
37     this.userName = userName;
38 }
39 public String getContactNumber() {
40     return contactNumber;
41 }
42 public void setContactNumber(String contactNumber) {
43     this.contactNumber = contactNumber;
44 }
45 public String getEmailId() {
46     return emailId;
47 }
48 public void setEmailId(String emailId) {
49     this.emailId = emailId;
50 }
51 public String getPassword() {
52     return password;
53 }
54 public void setPassword(String password) {
55     this.password = password;
56 }
57
58 public List<Roles> getRoles() {
59     return roles;
60 }
public void setRoles(List<Roles> roles) {
```
- Status Bar:** Writable, Smart Insert, 18:13:642
- Taskbar:** Icons for File, Home, Recent, Task View, Taskbar View, Taskbar Settings, Taskbar Help, Taskbar Status, Taskbar Control Panel, Taskbar Properties, Taskbar Settings, Taskbar Help, Taskbar Status, Taskbar Control Panel, Taskbar Properties.
- System Tray:** Icons for Power, Volume, Network, Battery, Weather (22°C), AQI 62, ENG, 22:37, 24-08-2021.



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A[▲]
- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - Coupon.java
 - Dish.java
 - DishRating.java
 - ErrorInfo.java
 - Login.java
 - OrderItems.java
 - Orders.java
 - Restaurant.java
 - RestaurantTransaction.java
 - Role.java
 - Roles.java
 - SuccessInfo.java
 - TableBooking.java
 - UserAddress.java
 - UserLikes.java
 - Users.java
 - Wallet.java
 - com.infy.service
 - com.infy.utility
- src/main/resources

*Users.java

```
59     return roles;
60   }
61   public void setRoles(List<Roles> roles) {
62     this.roles = roles;
63   }
64   public List<Restaurant> getRestaurants() {
65     return restaurants;
66   }
67   public void setRestaurants(List<Restaurant> restaurants) {
68     this.restaurants = restaurants;
69   }
70   public List<UserAddress> getAddressList() {
71     return addressList;
72   }
73   public void setAddressList(List<UserAddress> addressList) {
74     this.addressList = addressList;
75   }
76   public Wallet getWallet() {
77     return wallet;
78   }
79   public void setWallet(Wallet wallet) {
80     this.wallet = wallet;
81   }
82   public List<UserLikes> getUserLikesList() {
83     return userLikesList;
84   }
85   public void setUserLikesList(List<UserLikes> userLikesList) {
86     this.userLikesList = userLikesList;
87   }
88   public List<Orders> getOrdersList() {
89     return ordersList;
90   }
91   public void setOrdersList(List<Orders> ordersList) {
92     this.ordersList = ordersList;
```

Writable

Smart Insert

18:13:642



Type here to search

22°C AQI 62 ENG 22:37
24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A
- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - Coupon.java
 - Dish.java
 - DishRating.java
 - ErrorInfo.java
 - Login.java
 - OrderItems.java
 - Orders.java
 - Restaurant.java
 - RestaurantTransaction.java
 - Role.java
 - Roles.java
 - SuccessInfo.java
 - TableBooking.java
 - UserAddress.java
 - UserLikes.java
 - Users.java
 - Wallet.java
 - com.infy.service
 - com.infy.utility
- src/main/resources

*Users.java

```
63     }
64     public List<Restaurant> getRestaurants() {
65         return restaurants;
66     }
67     public void setRestaurants(List<Restaurant> restaurants) {
68         this.restaurants = restaurants;
69     }
70     public List<UserAddress> getAddressList() {
71         return addressList;
72     }
73     public void setAddressList(List<UserAddress> addressList) {
74         this.addressList = addressList;
75     }
76     public Wallet getWallet() {
77         return wallet;
78     }
79     public void setWallet(Wallet wallet) {
80         this.wallet = wallet;
81     }
82     public List<UserLikes> getUserLikesList() {
83         return userLikesList;
84     }
85     public void setUserLikesList(List<UserLikes> userLikesList) {
86         this.userLikesList = userLikesList;
87     }
88     public List<Orders> getOrdersList() {
89         return ordersList;
90     }
91     public void setOrdersList(List<Orders> ordersList) {
92         this.ordersList = ordersList;
93     }
94 }
95 }
```

Writable

Smart Insert

18:13:642

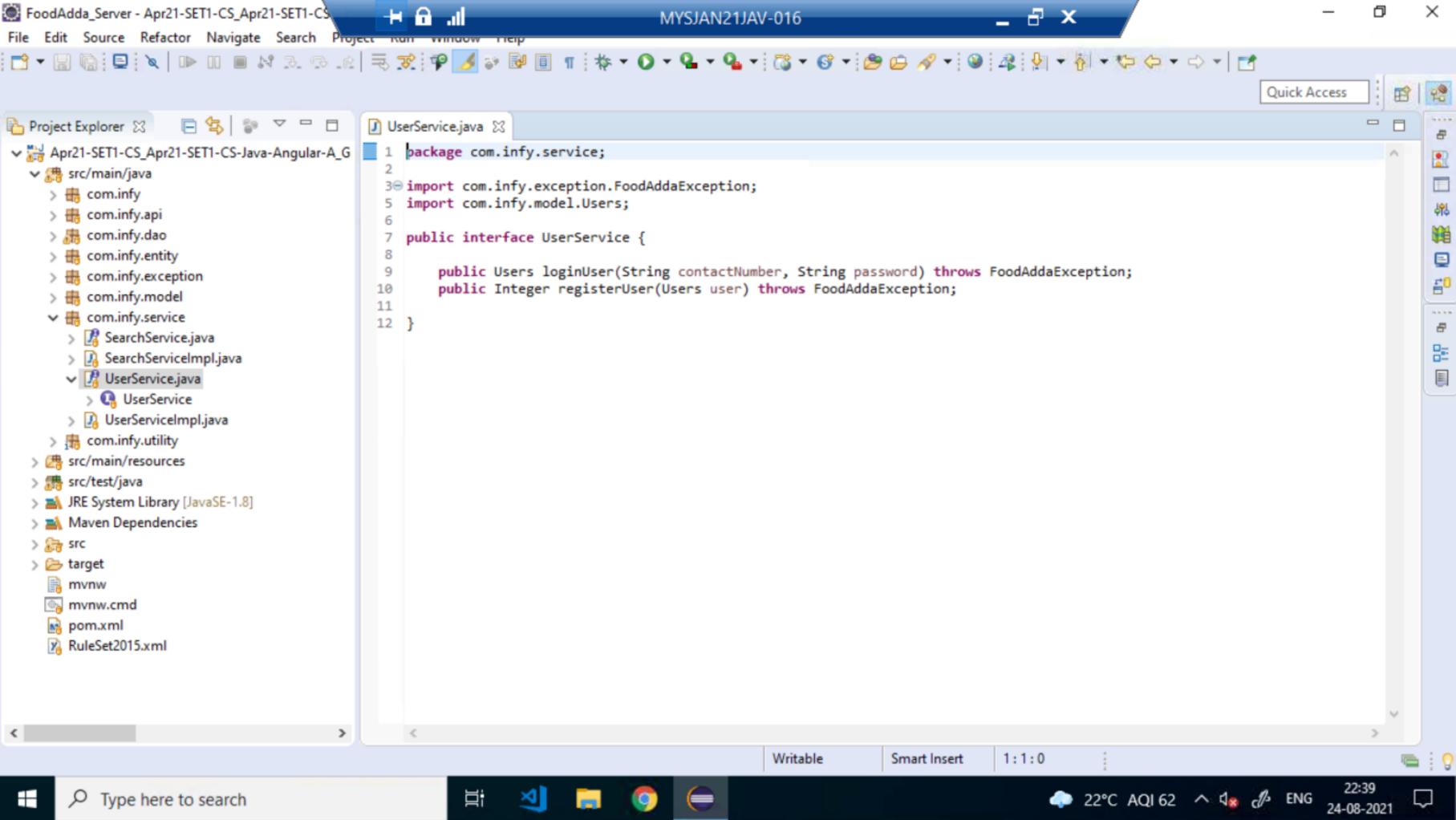


22°C AQI 62

22:37
24-08-2021

Type here to search





FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - SearchService.java
 - SearchServiceImpl.java
 - UserService.java
 - UserService
 - UserServiceImpl.java
 - com.infy.utility
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

UserServiceImpl.java

```
1 package com.infy.service;
2
3 import java.util.ArrayList;
4 import java.util.List;
5 import java.util.Optional;
6
7 import javax.transaction.Transactional;
8
9 import org.springframework.beans.factory.annotation.Autowired;
10 import org.springframework.stereotype.Service;
11
12 import com.infy.dao.UserDAO;
13 import com.infy.entity.RolesEntity;
14 import com.infy.entity.UserAddressEntity;
15 import com.infy.entity.UsersEntity;
16 import com.infy.exception.FoodAddaException;
17 import com.infy.model.Role;
18 import com.infy.model.Roles;
19 import com.infy.model.Users;
20
21 @Service(value="userService")
22 @Transactional
23 public class UserServiceImpl implements UserService {
24
25     @Autowired
26     UserDAO userDAO;
27
28
29     @Override
30     public Users loginUser(String contactNumber, String password) throws FoodAddaException {
31         Optional<UsersEntity> userOptional = userDAO.findByContactNumber(contactNumber);
32         UsersEntity userEntity = userOptional.orElseThrow(() -> new FoodAddaException("UserService.USER_NOT_EXISTS"));
33         if(!userEntity.getPassword().equals(password)) {
34             throw new FoodAddaException("UserService.INVALID CREDENTIALS");
35         }
36     }
37 }
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:39 24-08-2021



Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - SearchService.java
 - SearchServiceImpl.java
 - UserService.java
 - UserServiceImpl.java
 - com.infy.utility
- src/main/resources
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

```
UserServiceImpl.java
```

```
28
29
30     @Override
31     public Users loginUser(String contactNumber, String password) throws FoodAddaException {
32         Optional<UsersEntity> userOptional = userDao.findByContactNumber(contactNumber);
33         UsersEntity userEntity = userOptional.orElseThrow(() -> new FoodAddaException("UserService.USER_NOT_EXISTS"));
34         if(!userEntity.getPassword().equals(password)) {
35             throw new FoodAddaException("UserService.INVALID_CREDENTIALS");
36         }
37         Users users = new Users();
38         users.setUserId(userEntity.getUserId());
39         users.setUserName(userEntity.getUserName());
40         users.setContactNumber(userEntity.getContactNumber());
41         users.setEmailId(userEntity.getEmailId());
42         users.setPassword(userEntity.getPassword());
43         List<Roles> roles = new ArrayList<>();
44         for(RolesEntity rolesEntity : userEntity.getRoles()) {
45             Roles role = new Roles();
46             role.setRoleId(rolesEntity.getRoleId());
47             role.setRoleType(rolesEntity.getRoleType());
48             roles.add(role);
49         }
50         users.setRoles(roles);
51         return users;
52     }
53
54     @Override
55     public Integer registerUser(Users users) throws FoodAddaException {
56         Optional<UsersEntity> optionalUser = userDao.findByContactNumber(users.getContactNumber());
57         if(!optionalUser.isEmpty()) {
58             Boolean checkRole = false;
59             for(RolesEntity role : optionalUser.get().getRoles()) {
60                 if(role.getRoleType() == users.getRoles().get(0).getRoleType()) {
```

Writable

Smart Insert

1:1:0



Type here to search



22°C AQI 62

22:39
24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS

MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

UserServiceImpl.java

```
56     if(!optionalUser.isEmpty()) {
57         Boolean checkRole = false;
58         for(RolesEntity role : optionalUser.get().getRoles()) {
59             if(role.getRoleType() == users.getRoles().get(0).getRoleType()) {
60                 checkRole = true;
61                 break;
62             }
63         }
64         if(Boolean.TRUE.equals(checkRole)) {
65             throw new FoodAddaException("UserService.USER_ALREADY_EXISTS");
66         } else {
67             List<RolesEntity> rolesList=new ArrayList<>();
68             users.getRoles().forEach(role->{
69                 RolesEntity roless=new RolesEntity();
70                 roless.setRoleId(role.getRoleId());
71                 roless.setRoleType(role.getRoleType());
72                 rolesList.add(roless);
73             });
74
75             if(users.getRoles().get(0).getRoleType() == Role.CUSTOMER) {
76                 List<UserAddressEntity> userAddressList=new ArrayList<>();
77                 users.getAddressList().forEach(address->{
78                     UserAddressEntity userAddress=new UserAddressEntity();
79                     userAddress.setUserAddressId(address.getUserAddressId());
80                     userAddress.setUserAddressName(address.getUserAddressName());
81                     userAddress.setAddressLine1(address.getAddressLine1());
82                     userAddress.setAddressLine2(address.getAddressLine2());
83                     userAddress.setArea(address.getArea());
84                     userAddress.setCity(address.getCity());
85                     userAddress.setPincode(address.getPincode());
86                     userAddress.setUserState(address.getUserState());
87                     userAddressList.add(userAddress);
88                 });
89             });
90         }
91     }
92 }
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:40 24-08-2021

The screenshot shows an IDE interface with the following details:

- Title Bar:** FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS, MYSJAN21JAV-016
- Toolbar:** Standard Windows-style toolbar with icons for file operations, search, and navigation.
- Quick Access:** A small window in the top right corner.
- Project Explorer:** Shows the project structure:
 - Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - SearchService.java
 - SearchServiceImpl.java
 - UserService.java
 - UserService
 - UserServiceImpl.java
 - com.infy.utility
 - src/main/resources
 - src/test/java
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml
- Code Editor:** The main window displays the Java code for `UserServiceImpl.java`. The code handles user roles and addresses, specifically for a customer role. It includes logic to set PIN code, user state, and address lists, and to map user roles and addresses to their respective entities.
- Status Bar:** Shows "Writable", "Smart Insert", and a ratio of "1:1:0".
- Taskbar:** Shows the Start button, task switcher, and pinned application icons for File Explorer, File Manager, Task View, Edge, Google Chrome, and File History.
- System Tray:** Shows the date (24-08-2021), time (22:40), battery level, and network status.

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - SearchService.java
 - SearchServiceImpl.java
 - UserService.java
 - UserService - UserServiceImpl.java
 - com.infy.utility
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

UserServiceImpl.java

```
108     rolesList.add(roless);
109 }
110 usersEntity.setRoles(rolesList);
111 //List of type UserAddressEntity created
112 if(users.getRoles().get(0).getRoleType() == Role.CUSTOMER) {
113     List<UserAddressEntity> userAddressList=new ArrayList<>();
114     users.getAddressList().forEach(address->{
115         UserAddressEntity userAddress=new UserAddressEntity();
116         userAddress.setUserAddressId(address.getUserAddressId());
117         userAddress.setUserAddressName(address.getUserAddressName());
118         userAddress.setAddressLine1(address.getAddressLine1());
119         userAddress.setAddressLine2(address.getAddressLine2());
120         userAddress.setArea(address.getArea());
121         userAddress.setCity(address.getCity());
122         userAddress.setPincode(address.getPincode());
123         userAddress.setUserState(address.getUserState());
124         userAddressList.add(userAddress);
125     });
126     usersEntity setAddressList(userAddressList);
127 } else {
128     usersEntity setAddressList(null);
129 }
130 usersEntity.setOrdersList(null);
131 usersEntity.setRestaurants(null);
132 usersEntity.setWallet(null);
133 usersEntity.setUserLikesList(null);
134 return userDao.save(usersEntity).getUserId();
135 }
136 }
137 }
138 }
139 }
140 }
141 }
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:40 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
 - com.infy
 - com.infy.api
 - com.infy.dao
 - com.infy.entity
 - com.infy.exception
 - com.infy.model
 - com.infy.service
 - com.infy.utility
 - ExceptionControllerAdvice.java
 - ExceptionControllerAdvice
 - HashingUtility.java
 - LoggingAspect.java
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

ExceptionControllerAdvice.java

```
1 package com.infy.utility;
2
3 import java.time.LocalDateTime;
4 import java.util.stream.Collectors;
5
6 import javax.validation.ConstraintViolationException;
7
8 import org.springframework.beans.factory.annotation.Autowired;
9 import org.springframework.context.annotation.PropertySource;
10 import org.springframework.core.env.Environment;
11 import org.springframework.http.HttpStatus;
12 import org.springframework.http.ResponseEntity;
13 import org.springframework.web.bind.MethodArgumentNotValidException;
14 import org.springframework.web.bind.annotation.ExceptionHandler;
15 import org.springframework.web.bind.annotation.RestControllerAdvice;
16
17 import com.infy.exception.FoodAddaException;
18 import com.infy.model.ErrorInfo;
19
20 @RestControllerAdvice
21 @PropertySource("classpath:messages.properties")
22 public class ExceptionControllerAdvice {
23
24     @Autowired
25     private Environment environment;
26
27     @ExceptionHandler(Exception.class)
28     public ResponseEntity<ErrorInfo> exceptionHandler(Exception exception) {
29         ErrorInfo error = new ErrorInfo();
30         error.setErrorMessage(exception.getMessage());
31         error.setErrorCode(HttpStatus.INTERNAL_SERVER_ERROR.value());
32         error.setTimestamp(LocalDateTime.now());
33         return new ResponseEntity<>(error, HttpStatus.INTERNAL_SERVER_ERROR);
34     }
}
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:40 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

src/main/java

- com.infy
- com.infy.api
- com.infy.dao
- com.infy.entity
- com.infy.exception
- com.infy.model
- com.infy.service
- com.infy.utility
- ExceptionControllerAdvice.java
- HashingUtility.java
- LoggingAspect.java

src/main/resources

src/test/java

JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

ExceptionControllerAdvice.java

```
27@ExceptionHandler(Exception.class)
28    public ResponseEntity<ErrorInfo> exceptionHandler(Exception exception) {
29        ErrorInfo error = new ErrorInfo();
30        error.setErrorMessage(exception.getMessage());
31        error.setErrorCode(HttpStatus.INTERNAL_SERVER_ERROR.value());
32        error.setTimestamp(LocalDateTime.now());
33        return new ResponseEntity<>(error, HttpStatus.INTERNAL_SERVER_ERROR);
34    }
35
36@ExceptionHandler(FoodAddaException.class)
37    public ResponseEntity<ErrorInfo> foodAddaExceptionHandler(FoodAddaException exception) {
38        ErrorInfo error = new ErrorInfo();
39        error.setErrorMessage(environment.getProperty(exception.getMessage()));
40        error.setTimestamp(LocalDateTime.now());
41        error.setErrorCode(HttpStatus.NOT_FOUND.value());
42        return new ResponseEntity<>(error, HttpStatus.NOT_FOUND);
43    }
44
45
46@ExceptionHandler(MethodArgumentNotValidException.class)
47    public ResponseEntity<ErrorInfo> exceptionHandler(MethodArgumentNotValidException exception) {
48        ErrorInfo errorInfo = new ErrorInfo();
49        errorInfo.setErrorCode(HttpStatus.BAD_REQUEST.value());
50        String errorMsg = exception.getBindingResult().getAllErrors().stream().map(x -> environment.getProperty(x.getDefaultMessage()))
51            .collect(Collectors.joining(", "));
52        errorInfo.setErrorMessage(errorMsg);
53        errorInfo.setTimestamp(LocalDateTime.now());
54        return new ResponseEntity<>(errorInfo, HttpStatus.BAD_REQUEST);
55    }
56
57@ExceptionHandler(ConstraintViolationException.class)
58    public ResponseEntity<ErrorInfo> exceptionHandler(ConstraintViolationException exception) {
59        ErrorInfo errorInfo = new ErrorInfo();
60        errorInfo.setErrorCode(HttpStatus.BAD_REQUEST.value());
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:41 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

src/main/java

- com.infy
- com.infy.api
- com.infy.dao
- com.infy.entity
- com.infy.exception
- com.infy.model
- com.infy.service
- com.infy.utility
- ExceptionControllerAdvice.java
- HashingUtility.java
- LoggingAspect.java

src/main/resources

src/test/java

JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*ExceptionControllerAdvice.java

```
38     ErrorInfo error = new ErrorInfo();
39     error.setErrorMessage(environment.getProperty(exception.getMessage()));
40     error.setTimestamp(LocalDateTime.now());
41     error.setErrorCode(HttpStatus.NOT_FOUND.value());
42     return new ResponseEntity<>(error, HttpStatus.NOT_FOUND);
43 }
44
45
46@ExceptionHandler(MethodArgumentNotValidException.class)
47 public ResponseEntity<ErrorInfo> exceptionHandler(MethodArgumentNotValidException exception) {
48     ErrorInfo errorInfo = new ErrorInfo();
49     errorInfo.setErrorCode(HttpStatus.BAD_REQUEST.value());
50     String errorMsg = exception.getBindingResult().getAllErrors().stream().map(x ->
51         environment.getProperty(x.getDefaultMessage()))
52             .collect(Collectors.joining(", "));
53     errorInfo.setErrorMessage(errorMsg);
54     errorInfo.setTimestamp(LocalDateTime.now());
55     return new ResponseEntity<>(errorInfo, HttpStatus.BAD_REQUEST);
56 }
57
58@ExceptionHandler(ConstraintViolationException.class)
59 public ResponseEntity<ErrorInfo> exceptionHandler(ConstraintViolationException exception) {
60     ErrorInfo errorInfo = new ErrorInfo();
61     errorInfo.setErrorCode(HttpStatus.BAD_REQUEST.value());
62     String errorMsg = exception.getConstraintViolations().stream().map(x -> environment.getProperty(x.getMessage()))
63         .collect(Collectors.joining(", "));
64     errorInfo.setErrorMessage(errorMsg);
65     errorInfo.setTimestamp(LocalDateTime.now());
66     return new ResponseEntity<>(errorInfo, HttpStatus.BAD_REQUEST);
67 }
68
69 }
70
71 }
```

Writable Smart Insert 51 : 10 : 2039

Type here to search

22°C AQI 62 ENG 22:41 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Navigate Search Project Run Window Help

Quick Access

*messages.properties

```
1 #SearchService messages
2 SearchService.NO_RESTAURANTS_FOUND=Sorry! No restaurants found.
3 SearchService.RESTAURANT_DOES_NOT_EXIST=Restaurant does not exist.
4
5
6 #RestaurantValidator messages
7 RestaurantValidator.INVALID_RESTAURANT_TYPE=Please enter valid restaurant type.
8 RestaurantValidator.INVALID_RESTAURANT_CONTACT=The contact number should contain 10 digits.
9 RestaurantValidator.INVALID_RESTAURANT_NAME=Please enter a valid restaurant name.
10 RestaurantValidator.INVALID_RESTAURANT_STREET=Please enter a valid address street.
11 RestaurantValidator.INVALID_RESTAURANT_PLOT_NUMBER=Please enter a valid plot no for the address.
12 RestaurantValidator.INVALID_RESTAURANT_AREA=Please enter valid area.
13 RestaurantValidator.INVALID_RESTAURANT_STATE=Please enter valid state.
14 RestaurantValidator.INVALID_RESTAURANT_PIN=Please enter valid PIN code.
15 RestaurantValidator.INVALID_RESTAURANT_RATING=The rating is invalid.
16 RestaurantValidator.INVALID_RESTAURANT_STATUS>Status is invalid.
17 #DishValidator messages
18 DishValidator.INVALID_DISH_NAME=Please enter valid dish name.
19 DishValidator.INVALID_DISH_TYPE=Please enter valid dish type.
20 DishValidator.INVALID_DISH_DESCRIPTION=The dish description should contain only alphabets.
21 DishValidator.INVALID_DISH_SPECIALTY=The dish specialty should contain only alphabets.
22 DishValidator.INVALID_DISH_PRICE=The price should be more than 0.
23 DishValidator.INVALID_DISH_RATING=The dish rating should be more than 0 and less than 5.
24 DishValidator.INVALID_DISH_CUISINE=Please enter a valid dish cuisine.
25
26 RestaurantRegisterService.NO_DISH_FOUND= Sorry! No dish is available.
27 RestaurantRegisterService.DISH_CANNOT_BE_ADDED=Sorry! Dish cannot be added with the given details.
28 RestaurantRegisterService.DISH_CANNOT_BE_UPDATED= Dish can't be updated.
29 RestaurantRegisterService.RESTAURANT_CANNOT_BE_REGISTERED= Sorry! Restaurant cannot be registered with the given details.
30 #RegisterAPI messages
31 RestaurantRegisterAPI.RESTAURANT_REGISTRATION_SUCCESS=Congratulations! Restaurant
32 registered successfully with Restaurant ID:
33 RestaurantRegisterAPI.DISH_DETAILS_UPDATED=Dish details updated successfully!
34 RestaurantRegisterAPI.DISH_DETAILS_DELETED=Dish deleted successfully!
35 RestaurantRegisterAPI.DISH_DETAILS_ADDED=Dish details added successfully!
```



Type here to search



22°C AQI 62 ENG 22:42
24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Navigate Search Project Run Window Help

Quick Access

Project Explorer

src/main/java

- com.infy
- com.infy.api
- com.infy.dao
- com.infy.entity
- com.infy.exception
- com.infy.model
- com.infy.service
- com.infy.utility

src/main/resources

- application.properties
- messages.properties
- Table Scripts_Cibo.sql

src/test/java

JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

- mvnw
- mvnw.cmd
- pom.xml
- RuleSet2015.xml

*messages.properties

```
20 RestaurantRegisterService.DISH_CANNOT_BE_UPDATED= Dish can't be updated.
21 RestaurantRegisterService.RESTAURANT_CANNOT_BE_REGISTERED= Sorry! Restaurant cannot be registered with the given details.
22 #RegisterAPI messages
23 RestaurantRegisterAPI.RESTAURANT_REGISTRATION_SUCCESS=Congratulations! Restaurant
24 registered successfully with Restaurant ID:
25 RestaurantRegisterAPI.DISH_DETAILS_UPDATED=Dish details updated successfully!
26 RestaurantRegisterAPI.DISH_DETAILS_DELETED=Dish deleted successfully!
27 RestaurantRegisterAPI.DISH_DETAILS_ADDED=Dish details added successfully!
28 #RecommendationService messages
29 RecommendationService.NO_RESTAURANT_FOUND= Sorry! No restaurants are available in your area.
30
31 #UserService messages
32 UserService.INVALID_CREDENTIALS=Invalid user name or password.
33 UserService.USER_ALREADY_EXISTS=User already exists.
34 UserService.USER_NOT_EXISTS=User not exists.
35 UserService.ADDRESS_NAME_ALREADY_EXISTS= Address name already exist.
36 UserService.ADDRESS_NOT_FOUND= Address not found.
37
38 #OrderService.messages
39 orderService.NO_USER_FOUND= Sorry! No user found.
40 orderService.NO_ORDER_FOUND= No orders found.
41
42 #UserValidator messages
43 UserValidator.INVALID_CONTACT_NUMBER_FORMAT=The contact number should contain 10 digits.
44 UserValidator.INVALID_PASSWORD_FORMAT= Password should contain one upper case, one lower case,
45 one number and one character at least.
46 UserValidator.INVALID_USERNAME_FORMAT=User name should contain only letters of the English
47 alphabet or one space between words.
48 UserValidator.INVALID_EMAIL_ID_FORMAT=Invalid email id, make sure it is in example@example.com format.
49 UserValidator.INVALID_ROLE_TYPE = Role type should either be customer or vendor.
50
51
52
53
54
55
56
57
58
59
60 #UserAPI messages
61 UserAPI.REGISTER_USER_SUCCESS1=Congratulations
62 UserAPI.REGISTER_USER_SUCCESS2=! You have successfully registered to Cibo
63 UserAPI.ADDRESS_DETAILS_DELETED= address deleted successfully.
```

Writable Insert 55:1:3199

Type here to search

22°C AQI 62 ENG 22:42 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Navigate Search Project Run Window Help

Quick Access

*messages.properties

```
49
50 #UserValidator messages
51 UserValidator.INVALID_CONTACT_NUMBER_FORMAT=The contact number should contain 10 digits.
52 UserValidator.INVALID_PASSWORD_FORMAT= Password should contain one upper case, one lower case,
53 one number and one character at least.
54 UserValidator.INVALID_USERNAME_FORMAT=User name should contain only letters of the English
55 alphabet or one space between words.
56 UserValidator.INVALID_EMAIL_ID_FORMAT=Invalid email id, make sure it is in example@exam.com format.
57 UserValidator.INVALID_ROLE_TYPE = Role type should either be customer or vendor.
58
59
60 #UserAPI messages
61 UserAPI.REGISTER_USER_SUCCESS1=Congratulations
62 UserAPI.REGISTER_USER_SUCCESS2=! You have successfully registered to Cibo
63 UserAPI.ADDRESS_DETAILS_DELETED= address deleted successfully.
64 # AdminServices Messages
65
66 AdminServices.RESTAURANT_DOES_NOT_EXIST=Restaurant you are searching for does not exist.
67 AdminService.NO_NEW_RESTAURANT=No new Restaurant Registered.
68 AdminService.NO_RESTAURANT_FOUND_UNDER_RATINGS =No restaurant below average rating.
69
70 #BookTableService messages
71 BookTableService.TABLE_NOT_BOOKED=Sorry! The table could not be booked, please try again.
72
73 #UserDAO messages
74 UserDAO.UPDATE_ADDRESS=Address updated successfully.
75
76 #UserAddress messages
77 UserAddress.NULL_ADDRESS_NAME=Address name field should not be empty.
78 UserAddress.NULL_ADDRESS_LINE_1=Address line 1 should have plot no./street name/ building name, etc.
79 UserAddress.NULL_ADDRESS_LINE_2=Address line 2 should have location/landmark.
80 UserAddress.INVALID_AREA_FORMAT=Area should contain proper area name.
81 UserAddress.INVALID_CITY_FORMAT=City should contain only alphabets.
82 UserAddress.INVALID_PIN_CODE=Pin code should be of 6-digit.
83
```

Line: 65
Press 'Tab' from proposal table or click for focus

Writable Insert 55:1:3199

Type here to search

22°C AQI 62 ENG 22:42 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

UserServiceTest.java

```
1 package com.infy.service.test;
2
3 import java.util.ArrayList;
4 import java.util.List;
5 import java.util.Optional;
6
7 import org.junit.Assert;
8 import org.junit.Rule;
9 import org.junit.Test;
10 import org.junit.jupiter.api.Assertions;
11 import org.junit.rules.ExpectedException;
12 import org.junit.runner.RunWith;
13 import org.mockito.InjectMocks;
14 import org.mockito.Mock;
15 import org.mockito.Mockito;
16 import org.springframework.boot.test.context.SpringBootTest;
17 import org.springframework.test.context.junit4.SpringRunner;
18
19 import com.infy.dao.UserDAO;
20 import com.infy.entity.RolesEntity;
21 import com.infy.entity.UserAddressEntity;
22 import com.infy.entity.UsersEntity;
23 import com.infy.exception.FoodAddaException;
24 import com.infy.model.Login;
25 import com.infy.model.Role;
26 import com.infy.model.Roles;
27 import com.infy.model.UserAddress;
28 import com.infy.model.Users;
29 import com.infy.service.UserService;
30 import com.infy.service.UserServiceImpl;
31 import com.infy.utility.HashingUtility;
32
33 @RunWith(SpringRunner.class)
34 @SpringBootTest
```

Writable Smart Insert 1:1:0

Type here to search

22°C AQI 62 ENG 22:43 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
32
33     @RunWith(SpringRunner.class)
34     @SpringBootTest
35     public class UserServiceTest {
36
37         @Mock
38         private UserDAO userDAO;
39
40         @InjectMocks
41         private UserService userService = new UserServiceImpl();
42
43
44
45         @Test
46         public void authenticateUserNotExist() throws Exception {
47             Login login = new Login();
48             login.setContactNumber("9876543210");
49             login.setPassword(HashingUtility.getHashValue("Scott@1234"));
50             Optional<UsersEntity> user = Optional.empty();
51             Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(user);
52             FoodAddaException exception = Assertions.assertThrows(FoodAddaException.class, () ->
53                 userService.loginUser(login.getContactNumber(), login.getPassword()));
54             Assertions.assertEquals("UserService.USER_NOT_EXISTS", exception.getMessage());
55         }
56
57         @Test
58         public void authenticateUserInvalidCredential() throws Exception {
59             Login login = new Login();
60             login.setContactNumber("8884967823");
61             login.setPassword(HashingUtility.getHashValue("Scott@1234"));
62
63             UsersEntity users = new UsersEntity();
64             users.setUserId(101);
65             users.setUserName("SCOTT");
66         }
67     }
68 }
```

Writable Smart Insert 53 : 9 : 1626

Type here to search

22°C AQI 62 ENG 22:43 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
62
63     UsersEntity users = new UsersEntity();
64     users.setUserId(101);
65     users.setUserName("SCOTT");
66     users.setContactNumber("8884967823");
67     users.setEmailId("scott@stark.com");
68     users.setPassword("3284cbd43ac6fc733d7c3d2176e7a52baeba81471702ec332a0a65689cf16e3");
69     List<RolesEntity> roles = new ArrayList<>();
70     RolesEntity role = new RolesEntity();
71     role.setRoleId(3);
72     role.setRoleType(Role.CUSTOMER);
73     roles.add(role);
74     users.setRoles(roles);
75
76     Optional<UsersEntity> user = Optional.of(users);
77     Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(user);
78     FoodAddaException exception = Assertions.assertThrows(FoodAddaException.class, () ->
79         userService.loginUser(login.getLoginNumber(), login.getPassword()));
80     Assertions.assertEquals("UserService.INVALID_CREDENTIALS", exception.getMessage());
81 }
82
83 @Test
84 public void authenticateUserValidCredential() throws Exception{
85     Login login = new Login();
86     login.setContactNumber("8884967823");
87     login.setPassword(HashingUtility.getHashValue("Scott@123"));
88
89     UsersEntity users = new UsersEntity();
90     users.setUserId(101);
91     users.setUserName("SCOTT");
92     users.setContactNumber("8884967823");
93     users.setEmailId("scott@stark.com");
94     users.setPassword("3284cbd43ac6fc733d7c3d2176e7a52baeba81471702ec332a0a65689cf16e3");
95     List<RolesEntity> roles = new ArrayList<>();
```

Writable Smart Insert 79: 9: 2698

Type here to search

22°C AQI 62 ENG 22:43 24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - src/main/resources
 - src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

```
*UserServiceTest.java
89     UsersEntity users = new UsersEntity();
90     users.setUserId(101);
91     users.setUserName("SCOTT");
92     users.setContactNumber("8884967823");
93     users.setEmailId("scott@stark.com");
94     users.setPassword("3284cbd43ac6fc733d7c3d2176e7a52baeba81471702ec332a0a65689cf16e3");
95     List<RolesEntity> roles = new ArrayList<>();
96     RolesEntity role = new RolesEntity();
97     role.setRoleId(3);
98     role.setRoleType(Role.CUSTOMER);
99     roles.add(role);
100    users.setRoles(roles);
101
102    Optional<UsersEntity> user = Optional.of(users);
103    Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(user);
104    Users userss = userService.loginUser(login.getContactNumber(), login.getPassword());
105    Assertions.assertEquals(users.getUserId(), userss.getUserId());
106}
107
108@Test
109 public void registerUserIfNotExistAndAddRoleCustomer() throws Exception{
110     UsersEntity usersEntity = new UsersEntity();
111     usersEntity.setUserId(5000);
112     usersEntity.setUserName("Rishabh");
113     usersEntity.setEmailId("rishabh@gmail.com");
114     usersEntity.setContactNumber("9876543211");
115     usersEntity.setPassword(HashingUtility.getHashValue("Rishabh@123"));
116     //RolesEntity (Entity) List is created
117     List<RolesEntity> rolesListEntities=new ArrayList<>();
118     RolesEntity rolesEntity = new RolesEntity();
119     rolesEntity.setRoleType(Role.CUSTOMER);
120     rolesListEntities.add(rolesEntity);
121     usersEntity.setRoles(rolesListEntities);
122     //List of two UserAddressEntity created
```

Writable

Smart Insert

79: 9: 2698



Type here to search



22°C AQI 62

22:44
24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - src/main/resources
 - src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

UserServiceTest.java

```
118     RolesEntity rolesEntity = new RolesEntity();
119     rolesEntity.setRoleType(Role.CUSTOMER);
120     rolesListEntities.add(rolesEntity);
121     usersEntity.setRoles(rolesListEntities);
122     //List of type UserAddressEntity created
123     List<UserAddressEntity> userAddressListEntities=new ArrayList<>();
124     UserAddressEntity userAddressEntity = new UserAddressEntity();
125     userAddressEntity.setUserAddressName("HOME");
126     userAddressEntity.setAddressLine1("Address Line");
127     userAddressEntity.setAddressLine2("Address Line 2");
128     userAddressEntity.setArea("Area");
129     userAddressEntity.setCity("City");
130     userAddressEntity.setPincode("111111");
131     userAddressEntity.setUserState("State");
132     userAddressListEntities.add(userAddressEntity);
133     usersEntity.setAddressList(userAddressListEntities);
134     usersEntity.setOrdersList(null);
135     usersEntity.setRestaurants(null);
136     usersEntity.setWallet(null);
137     usersEntity.setUserLikesList(null);
138
139     Users userData = new Users();
140     userData.setUserId(5000);
141     userData.setUserName("Rishabh");
142     userData.setEmailId("rishabh@gmail.com");
143     userData.setContactNumber("9876543211");
144     userData.setPassword(HashingUtility.getHashCode("Rishabh@123"));
145     //RolesEntity (Entity) List is created
146     List<Roles> rolesList=new ArrayList<>();
147     Roles roles = new Roles();
148     roles.setRoleType(Role.CUSTOMER);
149     rolesList.add(roles);
150     userData.setRoles(rolesList);
151     //List of type UserAddressEntity created
```

Writable

Smart Insert

122:49:4529



Type here to search



22°C AQI 62



ENG

22:44
24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - src/main/resources
 - src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

UserServiceTest.java

```
149     rolesList.add(roles);
150     userData.setRoles(rolesList);
151     //List of type UserAddressEntity created
152     List<UserAddress> userAddressList=new ArrayList<>();
153     UserAddress userAddress = new UserAddress();
154     userAddress.setUserAddressName("HOME");
155     userAddress.setAddressLine1("Address Line");
156     userAddress.setAddressLine2("Address Line 2");
157     userAddress.setArea("Area");
158     userAddress.setCity("City");
159     userAddress.setPincode("111111");
160     userAddress.setUserState("State");
161     userAddressList.add(userAddress);
162     userData.setAddressList(userAddressList);
163     userData.setOrdersList(null);
164     userData.setRestaurants(null);
165     userData.setWallet(null);
166     userData.setUserLikesList(null);
167     Mockito.when(userDAO.save(Mockito.any(UserEntity.class))).thenReturn(usersEntity);
168     Assertions.assertEquals(userData.getUserId() , userService.registerUser(userData));
169 }
170
171 @Test
172 public void registerUserIfNotExistAndRoleAdmin() throws Exception{
173     UsersEntity usersEntity = new UsersEntity();
174     usersEntity.setUserId(5000);
175     usersEntity.setUserName("Rishabh");
176     usersEntity.setEmailId("rishabh@gmail.com");
177     usersEntity.setContactNumber("9876543211");
178     usersEntity.setPassword(HashingUtility.getHashValue("Rishabh@123"));
179     //RolesEntity (Entity) List is created
180     List<RolesEntity> rolesListEntities=new ArrayList<>();
181     RolesEntity rolesEntity = new RolesEntity();
```

Writable

Smart Insert

122:49:4529



Type here to search



22°C AQI 62

22:44
24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
178     usersEntity.setContactNumber("9876543211");
179     usersEntity.setPassword(HashingUtility.getHashValue("Rishabh@123"));
180     //RolesEntity (Entity) List is created
181     List<RolesEntity> rolesListEntities=new ArrayList<>();
182     RolesEntity rolesEntity = new RolesEntity();
183     rolesEntity.setRoleType(Role.ADMIN);
184     rolesListEntities.add(rolesEntity);
185     usersEntity.setRoles(rolesListEntities);
186     usersEntity.setAddressList(null);
187     usersEntity.setOrdersList(null);
188     usersEntity.setRestaurants(null);
189     usersEntity.setWallet(null);
190     usersEntity.setUserLikesList(null);
191
192     Users userData = new Users();
193     userData.setUserId(5000);
194     userData.setUserName("Rishabh");
195     userData.setEmailId("rishabh@gmail.com");
196     userData.setContactNumber("9876543211");
197     userData.setPassword(HashingUtility.getHashValue("Rishabh@123"));
198     //RolesEntity (Entity) List is created
199     List<Roles> rolesList=new ArrayList<>();
200     Roles roles = new Roles();
201     roles.setRoleType(Role.ADMIN);
202     rolesList.add(roles);
203     userData.setRoles(rolesList);
204     userData.setAddressList(null);
205     userData.setOrdersList(null);
206     userData.setRestaurants(null);
207     userData.setWallet(null);
208     userData.setUserLikesList(null);
209     Mockito.when(userDAO.save(Mockito.any(UsersEntity.class))).thenReturn(usersEntity);
210     Assertions.assertEquals(userData.getUserId() , userService.registerUser(userData));
211 
```

Writable Smart Insert 122:49:4529

Type here to search

22°C AQI 62 ENG 22:44 24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - src/main/resources
 - src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

```
*UserServiceTest.java
209     Mockito.when(userDAO.save(Mockito.any(UserEntity.class))).thenReturn(usersEntity);
210     Assertions.assertEquals(userData.getUserId(), userService.registerUser(userData));
211 }
214@Test
215 public void registerUserIfExistAndAddRoleCustomer() throws Exception{
216     UserEntity usersEntity = new UserEntity();
217     usersEntity.setId(101);
218     usersEntity.setName("SCOTT");
219     usersEntity.setEmailId("scott@stark.com");
220     usersEntity.setContactNumber("8884967823");
221     usersEntity.setPassword(HashingUtility.getHashCode("Scott@123"));
222     //RolesEntity (Entity) List is created
223     List<RolesEntity> rolesListEntities=new ArrayList<>();
224     RolesEntity rolesEntity = new RolesEntity();
225     rolesEntity.setRoleType(Role.ADMIN);
226     rolesListEntities.add(rolesEntity);
227     usersEntity.setRoles(rolesListEntities);
228     //List of type UserAddressEntity created
229     List<UserAddressEntity> userAddressListEntities=new ArrayList<>();
230     new UserAddressEntity().setUserAddressName("Home");
231     new UserAddressEntity().setAddressLine1("407, Rajhans Apartments");
232     new UserAddressEntity().setAddressLine2("Lashkar Mohalla");
233     new UserAddressEntity().setArea("Vasant Vihar");
234     new UserAddressEntity().setCity("Delhi");
235     new UserAddressEntity().setPincode("110029");
236     new UserAddressEntity().setUserState("Delhi");
237     userAddressListEntities.add(new UserAddressEntity());
238     usersEntity.setAddressList(userAddressListEntities);
239     usersEntity.setOrdersList(null);
240     usersEntity.setRestaurants(null);
241     usersEntity.setWallet(null);
242     usersEntity.setUserLikesList(null);
```

Writable

Smart Insert

122:49:4529



Type here to search



22°C AQI 62



ENG

22:44
24-08-2021



Quick Access

Project Explorer

- Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G
 - src/main/java
 - src/main/resources
 - src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - src
 - target
 - mvnw
 - mvnw.cmd
 - pom.xml
 - RuleSet2015.xml

```
*UserServiceTest.java
237     userAddressListEntities.add(new UserAddressEntity());
238     usersEntity.setAddressList(userAddressListEntities);
239     usersEntity.setOrdersList(null);
240     usersEntity.setRestaurants(null);
241     usersEntity.setWallet(null);
242     usersEntity.setUserLikesList(null);
243
244     Users userData = new Users();
245     userData.setUserId(101);
246     userData.setUserName("SCOTT");
247     userData.setEmailId("scott@stark.com");
248     userData.setContactNumber("8884967823");
249     userData.setPassword(HashingUtility.getHashValue("Scott@123"));
250     //RolesEntity (Entity) List is created
251     List<Roles> rolesList=new ArrayList<>();
252     Roles roles = new Roles();
253     roles.setRoleType(Role.CUSTOMER);
254     rolesList.add(roles);
255     userData.setRoles(rolesList);
256     //List of type UserAddressEntity created
257     List<UserAddress> userAddressList=new ArrayList<>();
258     UserAddress userAddress = new UserAddress();
259     userAddress.setUserAddressName("Home");
260     userAddress.setAddressLine1("407, Rajhans Apartments");
261     userAddress.setAddressLine2("Lashkar Mohalla");
262     userAddress.setArea("Vasant Vihar");
263     userAddress.setCity("Delhi");
264     userAddress.setPincode("110029");
265     userAddress.setUserState("Delhi");
266     userAddressList.add(userAddress);
267     userData.setAddressList(userAddressList);
268     userData.setOrdersList(null);
269     userData.setRestaurants(null);
270     userData.setWallet(null);
```

Writable

Smart Insert

122 : 49 : 4529



Type here to search



22°C AQI 62

ENG 22:45
24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS

MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
userData.setContactNumber(null);
userData.setRestaurants(null);
userData.setWallet(null);
userData.setUserLikesList(null);

Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(Optional.of(usersEntity));
Assertions.assertEquals(userData.getUserId(), userService.registerUser(userData));

}

@Test
public void registerUserIfExistAndAddRoleAdmin() throws Exception{
    UsersEntity usersEntity = new UsersEntity();
    usersEntity.setId(101);
    usersEntity.setName("SCOTT");
    usersEntity.setEmailId("scott@stark.com");
    usersEntity.setContactNumber("8884967823");
    usersEntity.setPassword(HashingUtility.getHashValue("Scott@123"));
    //RolesEntity (Entity) List is created
    List<RolesEntity> rolesListEntities=new ArrayList<>();
    RolesEntity rolesEntity = new RolesEntity();
    rolesEntity.setRoleType(Role.CUSTOMER);
    rolesListEntities.add(rolesEntity);
    usersEntity.setRoles(rolesListEntities);
    //List of type UserAddressEntity created
    List<UserAddressEntity> userAddressListEntities=new ArrayList<>();
    new UserAddressEntity().setUserAddressName("Home");
    new UserAddressEntity().setAddressLine1("407, Rajhans Apartments");
    new UserAddressEntity().setAddressLine2("Lashkar Mohalla");
    new UserAddressEntity().setArea("Vasant Vihar");
    new UserAddressEntity().setCity("Delhi");
    new UserAddressEntity().setPincode("110029");
    new UserAddressEntity().setLongitude("77.123456");
    new UserAddressEntity().setLatitude("28.567890");
}
```

Writable Smart Insert 122:49:4529

Type here to search

22°C AQI 62 ENG 22:46 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
308     usersEntity.setUserLikesList(null);
309
310     Users userData = new Users();
311     userData.setUserId(101);
312     userData.setUserName("SCOTT");
313     userData.setEmailId("scott@stark.com");
314     userData.setContactNumber("8884967823");
315     userData.setPassword(HashingUtility.getHashCode("Scott@123"));
316     //RolesEntity (Entity) List is created
317     List<Roles> rolesList=new ArrayList<>();
318     Roles roles = new Roles();
319     roles.setRoleType(Role.ADMIN);
320     rolesList.add(roles);
321     userData.setRoles(rolesList);
322     //List of type UserAddressEntity created
323     List<UserAddress> userAddressList=new ArrayList<>();
324     UserAddress userAddress = new UserAddress();
325     userAddress.setUserAddressName("Home");
326     userAddress.setAddressLine1("407, Rajhans Apartments");
327     userAddress.setAddressLine2("Lashkar Mohalla");
328     userAddress.setArea("Vasant Vihar");
329     userAddress.setCity("Delhi");
330     userAddress.setPincode("110029");
331     userAddress.setUserState("Delhi");
332     userAddressList.add(userAddress);
333     userData.setAddressList(userAddressList);
334     userData.setOrdersList(null);
335     userData.setRestaurants(null);
336     userData.setWallet(null);
337     userData.setUserLikesList(null);
338
339
340     Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(Optional.of(usersEntity));
341     Assertions.assertEquals(userData.getUserId() , userService.registerUser(userData));
```

Writable Smart Insert 122:49:4529

Type here to search

22°C AQI 62 ENG 22:46 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
340     Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(Optional.of(usersEntity));
341     Assertions.assertEquals(userData.getUserId(), userService.registerUser(userData));
342 }
343
344
345 @Test
346 public void registerUserIfExistRole() throws Exception{
347     UsersEntity usersEntity = new UsersEntity();
348     usersEntity.setUserId(101);
349     usersEntity.setUserName("SCOTT");
350     usersEntity.setEmailId("scott@stark.com");
351     usersEntity.setContactNumber("8884967823");
352     usersEntity.setPassword(HashingUtility.getHashCode("Scott@123"));
353     //RolesEntity (Entity) List is created
354     List<RolesEntity> rolesListEntities=new ArrayList<>();
355     RolesEntity rolesEntity = new RolesEntity();
356     rolesEntity.setRoleType(Role.CUSTOMER);
357     rolesListEntities.add(rolesEntity);
358     usersEntity.setRoles(rolesListEntities);
359     //List of type UserAddressEntity created
360     List<UserAddressEntity> userAddressListEntities=new ArrayList<>();
361     new UserAddressEntity().setUserAddressName("Home");
362     new UserAddressEntity().setAddressLine1("407, Rajhans Apartments");
363     new UserAddressEntity().setAddressLine2("Lashkar Mohalla");
364     new UserAddressEntity().setArea("Vasant Vihar");
365     new UserAddressEntity().setCity("Delhi");
366     new UserAddressEntity().setPincode("110029");
367     new UserAddressEntity().setUserState("Delhi");
368     userAddressListEntities.add(new UserAddressEntity());
369     usersEntity.setAddressList(userAddressListEntities);
370     usersEntity.setOrdersList(null);
371     usersEntity.setRestaurants(null);
372     usersEntity.setWallet(null);
373 }
```

Writable Smart Insert 122:49:4529

Type here to search

22°C AQI 62 ENG 22:46 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
366     new UserAddressEntity().setPincode("110029");
367     new UserAddressEntity().setUserState("Delhi");
368     userAddressListEntities.add(new UserAddressEntity());
369     usersEntity.setAddressList(userAddressListEntities);
370     usersEntity.setOrdersList(null);
371     usersEntity.setRestaurants(null);
372     usersEntity.setWallet(null);
373     usersEntity.setUserLikesList(null);
374
375     Users userData = new Users();
376     userData.setUserId(101);
377     userData.setUserName("SCOTT");
378     userData.setEmailId("scott@stark.com");
379     userData.setContactNumber("8884967823");
380     userData.setPassword(HashingUtility.getHashCode("Scott@123"));
381     //RolesEntity (Entity) List is created
382     List<Roles> rolesList=new ArrayList<>();
383     Roles roles = new Roles();
384     roles.setRoleType(Role.CUSTOMER);
385     rolesList.add(roles);
386     userData.setRoles(rolesList);
387     //List of type UserAddressEntity created
388     List<UserAddress> userAddressList=new ArrayList<>();
389     UserAddress userAddress = new UserAddress();
390     userAddress.setUserAddressName("Home");
391     userAddress.setAddressLine1("407, Rajhans Apartments");
392     userAddress.setAddressLine2("Lashkar Mohalla");
393     userAddress.setArea("Vasant Vihar");
394     userAddress.setCity("Delhi");
395     userAddress.setPincode("110029");
396     userAddress.setUserState("Delhi");
397     userAddressList.add(userAddress);
398     userData.setAddressList(userAddressList);
399     userData.setOrdersList(null);
```

Writable Smart Insert 122:49:4529

Type here to search

22°C AQI 62 ENG 22:46 24-08-2021

FoodAdda_Server - Apr21-SET1-CS_Apr21-SET1-CS MYSJAN21JAV-016

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

Apr21-SET1-CS_Apr21-SET1-CS-Java-Angular-A_G

- src/main/java
- src/main/resources
- src/test/java
 - com.infy
 - com.infy.dao.test
 - com.infy.service.test
 - SearchServiceTest.java
 - UserServiceTest.java
 - UserServiceTest
JRE System Library [JavaSE-1.8]

Maven Dependencies

src

target

mvnw

mvnw.cmd

pom.xml

RuleSet2015.xml

*UserServiceTest.java

```
379     userData.setContactNumber("8884967823");
380     userData.setPassword(HashingUtility.getHashValue("Scott@123"));
381     //RolesEntity (Entity) List is created
382     List<Roles> rolesList=new ArrayList<>();
383     Roles roles = new Roles();
384     roles.setRoleType(Role.CUSTOMER);
385     rolesList.add(roles);
386     userData.setRoles(rolesList);
387     //List of type UserAddressEntity created
388     List<UserAddress> userAddressList=new ArrayList<>();
389     UserAddress userAddress = new UserAddress();
390     userAddress.setUserAddressName("Home");
391     userAddress.setAddressLine1("407, Rajhans Apartments");
392     userAddress.setAddressLine2("Lashkar Mohalla");
393     userAddress.setArea("Vasant Vihar");
394     userAddress.setCity("Delhi");
395     userAddress.setPincode("110029");
396     userAddress.setUserState("Delhi");
397     userAddressList.add(userAddress);
398     userData.setAddressList(userAddressList);
399     userData.setOrdersList(null);
400     userData.setRestaurants(null);
401     userData.setWallet(null);
402     userData.setUserLikesList(null);

403
404
405     Mockito.when(userDAO.findByContactNumber(Mockito.anyString())).thenReturn(Optional.of(usersEntity));
406     FoodAddaException foodAddaException = Assertions.assertThrows(FoodAddaException.class, () ->
407         userService.registerUser(userData));
408     Assertions.assertEquals("UserService.USER_ALREADY_EXISTS" , foodAddaException.getMessage());
409
410 }
411
412 }
```

Writable Smart Insert 407: 9: 16336

Type here to search

23°C Light rain ENG 22:47 24-08-2021