



SAVEETHA SCHOOL OF ENGINEERING
SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



LAB MANUAL

CSA37

SOFTWARE TESTING



SAVEETHA SCHOOL OF ENGINEERING
SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CONTENTS:

| S.No | CONTENT | PAGE NUMBER |
|------|----------------------------------|-------------|
| 1 | CSE DEPARTMENT VISION-MISSION | |
| 2 | CSE- PO-PSO | |
| 3 | COURSE OUTCOMES | |
| 4 | DO'S AND DON'TS | |
| 5 | INSTRUCTIONS FOR THE EXPERIMENTS | |
| 6 | EXPERIMENT LIST | |
| 7 | CO-PO ATTAINMENT | |



CSE DEPARTMENT VISION-MISSION

Vision of the Department

To establish an environment to provide quality education and Inculcate research attributes among computer science engineering graduates through problem solving skills and technological innovations.

Mission of the Department

M1

To create and sustain an academic environment to the highest level in teaching and research by enhancing the knowledge of the faculties and students in technological advancements to solve real time problems.

M2

Providing a suitable environment for the students to develop professionalism with knowledge in Computer Science & Engineering to meet the contemporary industry needs and satisfy global standards.

M3

To facilitate the development of professional behaviour and stronger ethical values so as to work with commitment for the progress of the nation and face challenges with ethical and social responsibility.

INSTRUCTIONS FOR THE EXPERIMENTS



DO'S AND DON'TS

- ✓ Be on time to lab and maintain silence.
- ✓ Inform the instructor /TA in case of any working environment problem.
- ✓ Be aware of all the safety devices. Even through the instructor and TA will take care of emergencies.
- ✓ Bring all the necessary things required to do laboratory experiments like observation notebook, record notebook and any others.
- ✓ Enter the system number, in-time, out-time register number, name and signature in the students log register.
- ✓ Arrange the seating chairs and system accessories in place at the end of the session.
- ✓ Shut down the system properly and switch off the power switch at the end of the session.
- ✓ Keep your bags in front of the labs empty space.
- ✓ Do not eat, drink, chew gum, smoke or apply cosmetics in the lab.
- ✓ Do not unplug/plug any wires of the system connectivity.
- ✓ Do not use or charge the mobile phone or any electronic gadgets inside the lab.
- ✓ Not to troubleshoot by yourself without knowledge of instructor/TA.
- ✓ Do not open any unnecessary applications in system.
- ✓ Mobiles phones are prohibited inside the lab.
- ✓ Do not share a system to do experiments.

Students strictly follow all the above instructions



LIST OF EXPERIMENTS

COURSE CODE : CSA37

COURSE NAME : SOFTWARE TESTING

1. Select any system, such as an ATM system, and thoroughly examine its system specifications to identify and document the different software defects present.
2. To write a case study to identify & analyze the requirements of ARMS Portals and to write a positive Test Scenarios as well as Test cases for the above application.
3. To write a case study to identify & analyze the requirements of ARMS Portals and to write a negative Test Scenarios as well as Test cases for the above application.
4. To write a case study to identify & analyze the requirements of e-Commerce Applications and to write a positive Test Scenarios as well as Test cases for the above application.
5. To write a case study to identify & analyze the requirements of e-Commerce Applications and to write a negative Test Scenarios as well as Test cases for the above application.
6. To write a case study to identify & analyze the requirements of Food Delivery Applications and to write a positive Test Scenarios as well as Test cases for the above application.

7. To write a case study to identify & analyze the requirements of Food Delivery Applications and to write a negative Test Scenarios as well as Test cases for the above application
8. To write a case study to identify & analyze the requirements of Banking Applications and to write a positive Test Scenarios as well as Test cases for the above application.
9. To write a case study to identify & analyze the requirements of Banking Applications and to write a negative Test Scenarios as well as Test cases for the above application
10. Develop a comprehensive test plan document for a chosen application, such as a Library Management System, outlining the various testing strategies, methodologies, and criteria to ensure thorough evaluation and validation of the system's functionality.
11. Demonstrate the working of Junit to reverse a word and using assert statement for Proof of the value.
12. Write a white box testing code (junit) to String comparison of word and using assert statement for Proof the value.
13. Write a junit code for voting system and uses assert statement and verify the white box testing.
14. Write a program using function to calculate the simple interest. Suppose the customer is a senior citizen. He is being offered 12 percent rate of interest; for all other customers, the ROI is 10 percent. The output values should verify using white box testing.
15. Check whether the given number is palindrome or not and verify the output values should verify using white box testing.
16. Write a program to convert Decimal number equivalent to Binary number and octal numbers? The output values should verify using white box testing?

17. Write a Java Program to Convert a Given Number of Days in Terms of Years, Weeks & Days. The output values should verify using white box testing.
18. Find the factorial of n? The output values should verify using white box testing?
19. Find the year of the given date is leap year or not .The output values should verify using white box testing?
20. Write a program to find the square, cube of the given decimal number. The output values should verify using white box testing.
21. Write a Selenium program for automation and open a chrome browser with google.com
22. Write a Selenium program for automation and open a Mozilla browser with google.com
23. Write a Selenium program to Automate login in ARMS Portal with the help of Chrome browser
24. Write a Selenium program Automate login in Github Portal with the help of Chrome browser
25. Write a Selenium program Automate login in Swiggy Portal with the help of Chrome browser.

EXP NO : 1

ATM System

Objective: Select any system, such as an ATM system, and thoroughly examine its system specifications to identify and document the different software defects present.

Common software defects that can be found in ATM systems:

Features to be tested:

1. Validity of the card.
2. Withdraw Transaction flow of ATM.
3. Authentication of the user's.
4. Dispense the cash from the account.

| Bug-Id | Bug Name |
|---------------|---|
| ATM_001 | Invalid Card ATM_002 Invalid PIN |
| ATM_003 | Invalid Account type ATM_004 Insufficient Balance |
| ATM_005 | Transaction Limit |
| ATM_006 | Day limit |
| ATM_007 | Invalid money denominations |
| ATM_008 | Receipt not printed |
| ATM_009 | PIN change mismatch |

BUG REPORT:

| |
|---|
| Bug Id: ATM_001 |
| Bug Description: Invalid card |
| Steps to reproduce: 1. Keep valid card in the ATM. |
| Expected Result: Welcome Screen |
| Actual Result: Invalid card |
| Status : Pass/Fail |

Bug Id: ATM_002

Bug Description: Invalid PIN entered

Steps to reproduce:

1. Keep a valid card in ATM.
2. Enter the authorized PIN.
3. Menu screen should be displayed.

Expected Result: Menu screen displayed

Actual Result: Invalid PIN screen is displayed

Status : Pass/Fail

Bug Id: ATM_003

Bug Description: Invalid Account type selected.

Steps to reproduce:

1. Enter a valid user PIN number.
2. Select the withdraw option on the main menu.
3. Choose the correct type of account (either savings or current account).

Expected Result: Enter the Amount screen displayed

Actual Result: Invalid Account type screen is displayed.

Status : Pass/Fail

Bug Id: ATM_004

Bug Description: Insufficient Balance

Steps to reproduce:

1. Menu screen should be displayed.
2. Select the withdraw option.
3. Select the correct type of account.
4. Enter the sufficient amount to withdraw from the account.
5. Dispense the cash screen & amount to be deducted from account

Expected Result: Collect the amount screen displayed

Actual Result: Insufficient balance in the account

Status : Pass/Fail

Bug Id: ATM_005

Bug Description: Withdraw Limit per transaction.

Steps to reproduce:

1. Menu screen should be displayed.
2. Select the withdraw option.
3. Select the correct type of account.
4. Enter sufficient amount to withdraw from the account Transaction within the limit.
5. Dispense the cash screen & amount to be deducted from account.

Expected Result: Cash is dispensed and collect the receipt

Actual Result: Transaction limit exceeded screen is displayed

Status : Pass/Fail

Bug Id: ATM_006

Bug Description: Withdraw limit per day

Steps to reproduce:

1. Keep a valid card in ATM.
2. Enter the authorized PIN.
3. Enter the amount to withdraw from the account.
4. Amount enter is over the day limit (>40000)
5. Amount enter is over the day limit and display screen is displayed.

Expected Result: Cash is dispensed and collect the receipt.

Actual Result: Day limit exceeded screen is displayed.

Status : Pass/Fail

Bug Id: ATM_007

Bug Description: Amount enter denominations

Steps to reproduce:

1. Keep a valid card in ATM.
2. Enter the authorized PIN.
3. Enter the amount which should be in multiples of 100.
4. Cash Dispenser screen is displayed.

Expected Result: Collect the amount screen is displayed.

Actual Result: Amount enter not in required denominations.

EXP NO : 2

Case Study on ARMS Portals -Positive Test Cases

Objective: Case Study: Requirements Analysis and Testing for ARMS Portal.

| Sl.no | Items to be tested | Features | Approach | Test environment | Training needed | Risk | Test case | Pass/Fail |
|-------|--------------------|---|-----------------|---------------------------|-----------------|------|--|-----------|
| 1 | Home | Holidays Notification Attendance Conformatio n | Manual approach | Operating systems/Browser | Not required | No | -Open the website arms.saveetha.com -Enter the user name and password -click into home button -check whether holiday is there | Pass |
| 2 | My course | *Inprogress course *Graduation status *Completed course *External credits | Manual approach | Operating systems/Browser | Not required | No | -Open the website arms.saveetha.com -Enter the user name and password. -click on “=” button and go to my course | Pass |
| 3 | Enrollment | Select slot A Slot B Slot C Slot D *course details *Note=Approved Pending | Manual approach | Operating systems/Browser | Not required | No | - Open the website arms.saveetha.com -Enter the user name and password - click on “=” button and go to Enrollment select slot A, | Pass |

| | | | | | | | | |
|---|------------------|--|------------------------------------|---------------------------|--------------|----|---|------|
| | | | | | | | slot Bis there are not. | |
| 4 | Attendance | *Attendance report *course code *course name | Manual approach | Operating systems/Browser | Not required | No | - click on “=” button and go to attendance -Click on attendance report whether course code is there are not. | Pass |
| 5 | Assignment | * course *select *Complete/Assignment | Manual approach | Operating systems/Browser | Not required | No | -click on “=” button and go to Assignment. -Click on assignment select coming up | Pass |
| 6 | Examination | *No due request *Please enable pop up option | Manual approach | Operating systems/Browser | Not required | No | -click on “=” button and go to Examination .click list no due payment history is there are not | Pass |
| 7 | Financial record | *Due list *Payment history *Print payment | Manual approach Manual approach | Operating systems/Browser | Not required | No | click on “=” button and go to Financial record click due list payment | Pass |

| | | | | | | | | |
|----|-------------------|--|-----------------|---------------------------|--------------|----|-----------------------------|------|
| | | | | | | | history is there are not | |
| 8 | Disciplinary | Disciplinary record issue details, last action details | Manual approach | Operating systems/Browser | Not required | No | | Pass |
| 9 | Raise Infra issue | | Manual approach | Operating systems/Browser | Not required | No | | Pass |
| 10 | My profile | | Manual approach | Operating systems/Browser | Not required | No | | Pass |

EXP NO : 3

Case Study on ARMS Portals -Negative Test Cases

Objective: Case study to identify & analyze the requirements of ARMS Portals -negative Test Scenarios as well as Test cases for the above application.

| Item to be tested | Features | Approach | Data Test | Test Environment | Training needed | Risk | Test case | Expected output | Pass / Fail |
|----------------------|--|-----------------|----------------------|----------------------------|-----------------|------|--|--|-------------|
| 1) Home | Holidays Notification Attendance Confirmation | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * open website aims.sse. saveetha.com * enter username & password * enter wrong password | Username and password entered is invalid | Pass |
| 2) My course | * In Progress Courses * Graduation status * External credit status | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go to my course * Check whether completed courses is there or not | My course wasn't appeared | Pass |
| 3) Enrollment | Select Slot Select Slot A Slot B Course Details Note: Approved | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go to enrollment * Select Slot A, Slot B, Slot C | My enrollment wasn't appeared | Pass |
| 4) Attendance | S. No Course Code Course Name Class attended view | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go to Attendance * Click on Attendance report | Attendance wasn't appeared | Pass |
| 5) Assignment | Completed Assignment content Exams | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go on Assignment * Select Course opening Assignment | Assignment wasn't appeared | Pass |
| 6) Examination | No due request please enable popup options in web browser to print hall ticket | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Click on examination * Go to no due request whether it is there or not | Request is not accepted | Pass |
| 7) Financial Record | * Due List * Payment history * Print Payment | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go to Financial Record * Check due list, Payment history | Payment history not cleared | Pass |
| 8) Disciplinary | Disciplinary Record Issue details Last action details complaint | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go to Disciplinary issue details * Last action details | Issues were not cleared | Pass |
| 9) Raise Infra Issue | Raise issue content choose file location | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go to infra issues content * Choose file location | Not showing file location | Pass |
| 10) My Profile | Edit profile student record details | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go on my profile * Check student record details | Profile was not updating | Pass |

EXP NO : 4

Case study on e-Commerce Applications- Positive Test Cases

Objective: Case study to identify & analyze the requirements of Flipkart - Positive Test Scenarios as well as Test cases for the above application.

| Item to be tested | Features | Approach | Data Test | Test Environment | Training needed | Risk | Test case | Pass / Fail |
|-------------------|--|-----------------|-----------|------------------|-----------------|------|---|-------------|
| Home | Search button, shops, Notification, Account Cart | Manual approach | No | Laptop, Android | No | No | * Open the Flipkart * Click on the email, password, * Click on the Home page | Pass |
| Search button | Search for product, brands and shirts | Manual approach | No | Laptop, Android | No | No | * Open the Flipkart * Click on the home page the search button appear * Click on the search button and find the product | Pass |
| Shops | Offer Zone, More on Flipkart, Trending stores, recently viewed stores | Manual approach | No | Laptop, Android | No | No | * Open the Flipkart * Click on the home page * Click the shops | Pass |
| Notifications | All Offers | Manual approach | No | Laptop, Android | No | No | * Click on the Home page and click the notifications * We can find the any offers or getting to buy product. | Pass |
| Account | Orders, wishlist, coupons, Help Centre Edit Profile Logout | Manual approach | No | Laptop, Android | No | No | * Click on the Flipkart * Click on the Home page and press the Account and we will find the orders that we placed * Click on the Account and find the username and phone number | Pass |
| Cart | My cart, Address, Product, order place | Manual approach | No | Laptop, Android | No | No | * Click on the Flipkart * Click on the Home page and click the cart * Click the product and place the order in the my cart. | Pass |
| Accounts | Account setting, My activity, Earn with Flipcart, feed back, Information | Manual approach | No | Laptop, Android | No | No | * Click on the Flipkart * Click on the Home page and press the account * Click the account setting we can edit the profile, save the address | Pass |

| Item to be tested | Features | Approach | Data Test | Test Environment | Training needed | Risk | Test case | Pass / Fail |
|-------------------|---|-----------------|-----------|------------------|-----------------|------|---|-------------|
| My cart | Address, Remove, save later, Buy this now | Manual approach | No | Laptop, Android | No | No | * Click on the Flipkart * Click on the Home page and press the product and remove, save * Place the order | Pass |
| Search button | Search the product, Related searches, popular products, Discover more | Manual approach | No | Laptop, Android | No | No | * Click on the Flipkart * Click on the Home page and click the search button and search the product and * Place the order * We can search it any of the product that you wish. | Pass |
| Accounts | Order placed, Cancelled | Manual approach | No | Laptop, Android | No | No | * Click on the Flipkart * Click on the Home page and click the account and press the order * And we can find the product that we have placed the order and cancelled the order | Pass |

EXP NO : 5

Case study on e-Commerce Applications- Negative Test Cases

Objective: To write a case study to identify & analyze the requirements of e-Commerce Applications (Flipkart) and to write a negative Test Scenarios as well as Test cases for the above application.

| Items to be selected | Features | Approach | Data Test | Test Environment | Training Needed | Risk | Test case | Expected Output | Pass |
|----------------------|--|-----------------|-----------|------------------|-----------------|------|--|---|------|
| Home | Search button, shops, notifications, Account etc | Manual Approach | No | Laptop, Android | No | No | Step1:- Open the flipkart Step2:- Click on the Email, password | Other tab Opened => Incorrect Email or password | Pass |
| Search button | Search for products Brands & shirts | Manual Approach | No | Laptop, Android | No | No | Step1:- Search for products Step2:- Incorrect product will appear | Wrong Products Appear | Pass |
| Shops | Search for Mobiles, Fashion, Electronics | Manual Approach | Np | Laptop, Android | No | No | Step1:- Open the shops Step2:- Click on The mobiles & search | Searched Mobiles do not appear | Pass |

| Items to be selected | Features | Approach | Data Test | Test Environment | Training Needed | Risk | Test Case | Expected Output | Pass/Fail |
|----------------------|--------------------------------------|-----------------|-----------|------------------|-----------------|------|--|-----------------------------|-----------|
| Trending Stores | Search for student And wedding Store | Manual approach | No | Laptop, Android | No | No | Step1:- Open The shops Step2:- In that open trending stores Step3:- But it will open another store | Not open Student's Club | Pass |
| Notifications | Order list will appear | Manual approach | No | Laptop, Android | No | No | Step1:- Open the Notifications Step2:- Order placed Product Step3:- Does not appear in notifications | Does not show notifications | Pass |
| Account | Click on orders , Wishlist, Coupons | Manual approach | No | Laptop, Android | No | No | | Help centre Does not appear | Pass |

| Items to be selected | Features | Approach | Data Test | Test Environment | Training needed | Risk | Test Case | Expected output | Pass/Fail |
|----------------------|-----------------------|-----------------|-----------|------------------|-----------------|------|---|------------------------------|-----------|
| Accounts | Click on Logout | Manual Approach | No | Laptop, Android | No | No | Step1:- Open flipcart and click in account Step2:- At last click on logout option Step3:- But flipcart doesn't logout | Flipcart doesn't logout | Pass |
| Accounts | Click on edit profile | Manual Approach | No | Laptop, Android | No | No | Step1:- Open flipcart and click on accounts Step2:- Click on edit profile Step3:- Profile will not be updating | Profile will not be updating | Pass |
| Cart | Click on place order | Manual Approach | No | Laptop, Android | No | No | Step1:- Open accounts & click place order Step2:- Place order | Order not placed | Pass |
| Shops | Search for medicines | Manual Approach | No | Laptop, Android | No | No | Step1:- Open flipcart & click medicines Step2:- medicines will not appear | Medicines do not appear | Pass |

EXP NO : 6

Case study on Food Delivery Applications- Positive Test Cases

To write a case study to identify & analyze the requirements of Food Delivery Applications and to write a positive Test Scenarios as well as Test cases for the above application.

| Item to be tested | Features | Approach | Data Test | Test Environment | Training needed | Risk | Test case | Pass / Fail |
|-----------------------|---|-----------------|----------------------|----------------------------|-----------------|------|---|-------------|
| 1) Home | Offers Delivery Dining Money | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * open the website https://zomato.com * Enter user names & password * Click on the login button * Check whether offers, delivery, dining, money is there or not | Pass |
| 2) Food orders | * Your orders * Favourite orders * Address book | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on food orders * Check whether your orders, Favourite orders address books is there or not | Pass |
| 3) Restaurants Awards | * Winning Restaurants | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on restaurant awards * Check whether winning restaurants is there or not | Pass |
| 4) Zomato UPI | * Activate Zomato UPI * Zomato gold | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on Zomato UPI * Check activate UPI and Zomato gold is there or not | Pass |
| 5) Events | * Your event tickets * Events help | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on events * To check your event tickets * Events help is there or not | Pass |

| | | | | | | | | |
|--------------------------|--|-----------------|-------------------|----------------------------|--------------|----|---|------|
| 6) Dining | * Your transactions * Your dining rewards * Your slot bookings | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on dining * Open and check whether your transactions, your dining rewards, your slot bookings | Pass |
| 7) Money | * Buy gift card * Claim gift card * Zomato credits | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on money * Open and check whether buy gift card, claim gift card is there or not | Pass |
| 8) More | * Choose language * Send feed back * Log out | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on profile view activity * Open and check whether add review, add photo, edit profile, reviews is there or not | Pass |
| 9) Profile view activity | * Add review * Add photo * Edit Profile * Reviews | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on your rating * Open and check whether understand your rating and your calculated, ok is there or not | Pass |
| 10) Your Rating | * Understand your rating * Your rating calculated ok | Manual Approach | Username Password | Operating System / Browser | Not Required | No | * Click on '≡' button * Go on my profile * Check student record details | Pass |

EXP NO : 7

Case study on Food Delivery Applications- Negative Test Cases

To write a case study to identify & analyze the requirements of Food Delivery Applications and to write a negative Test Scenarios as well as Test cases for the above application

| Item to be searched | Features | Approach | Data set | Test environment | Training needed | Risk | Test case | Expected output | Pass/Fail |
|----------------------------|---|-----------------|---------------------|--------------------------|------------------------|-------------|---|--|------------------|
| Home | Offers Delivary Dining Money | Manual approach | User name /Password | Operating system/ Brower | Not Required | No | ->Open website https://zomoto.com ->Enter user name &password opened (or) not | Other tab opened >Incorrect email (or) password | Pass |
| Food order | Your order Favourite-Order Address Book | Manual approach | User name /Password | Operating system/ Brower | Not Required | No | Click on food order Favorite order Address is there or not | Wrong item appear | Pass |
| Restau rants Award s | Wining Restaurant | Manual approach | User name /Password | Operating system/ Brower | Not Required | No | ->Click on restaurant awards ->Check whether wining restaurant is there or not | Fake restaur ant name apper | Pass |
| Zomoto Upi | Activit y Zomot UPI Zomot gold | Manual approach | User name /Password | Operating system/ Brower | Not Required | No | ->Click on zomoto UPI ->Check activity UPI and zomoto gold is there (or) not | ->Wrong accept ->Insufficient balance will appear on wallet or bank | Pass |
| Events | Your events tickets Event help | Manual approach | User name /Password | Operating system/ Brower | Not Required | No | ->Click on the events your event ticket >Event help is veaver not | Wrong tickets display on the screen | Pass |

| | | | | | | | | | |
|-------------------------|---|-----------------|--------------------|-----------------------------|--------------|----|--|--|-------|
| Dining | Your transaction Your dining reward Your Slot booking | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Click on the dining disk on trance,your dining reward slot booking | After debited money from bank ofill foof order is booked | Pas s |
| Money | Buy gift card Clime gift card Card zomoto credit | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Click on money oprn and check buy gift cards | Not access for claming the gift card | Pas s |
| Move | Choose Language Send feedback | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Choose language and send feedback | Unable to send feedba ck layout | Pas s |
| Profile view - activity | Layout Add review Add photo Edit profile Review | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Profile view Add review review | Unable to send the review of users (Proxi review) | Pas s |
| Your rating | Understand Your rating Tour rayting eduoted OK. | Manual Approach | Username /Password | Operating system/ Brower | Not Required | No | Your rating Check whether your rating is ok is there caught | Unable to see your item rating | Pas s |

| Item to be searched | Features | Approach | Data set | Test environment | Training needed | Risk | Test case | Expected output | Pass/Fail |
|----------------------------|---|-----------------|---------------------|--------------------------|------------------------|-------------|---|---|------------------|
| Home | Offers Delivary Dining Money | Manual approach | User name /Password | Operating system/Browser | Not Required | No | ->Open website https://zomoto.com ->Enter user name &password opened (or) not | Other tab opened >Incorrect email (or) password | Pass |
| Food order | Your order Favourite-Order Address Book | Manual approach | User name /Password | Operating system/Browser | Not Required | No | Click on food order Favorite order Address is there or not | Wrong item appear | Pass |
| Restaurant Awards | Wining Restaurant | Manual approach | User name /Password | Operating system/Browser | Not Required | No | ->Click on restaurant awards ->Check whether wining restaurant is there or not | Fake restaurant name appear | Pass |
| Zomoto Upi | Activity Zomot UPI Zomot gold | Manual approach | User name /Password | Operating system/Browser | Not Required | No | ->Click on zomoto UPI ->Check activity UPI and zomoto gold is there (or) not | ->Wrong accept ->Insufficient balance will appear on wallet or bank | Pass |
| Events | Your events tickets Event help | Manual approach | User name /Password | Operating system/Browser | Not Required | No | ->Click on the events your event ticket >Event help is veaver not | Wrong tickets display on the screen | Pass |

| | | | | | | | | | |
|-------------------------|---|-----------------|--------------------|-----------------------------|--------------|----|--|--|-------|
| Dining | Your transaction Your dining reward Your Slot booking | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Click on the dining disk on trance,your dining reward slot booking | After debited money from bank ofill foof order is booked | Pas s |
| Money | Buy gift card Clime gift card Card zomoto credit | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Click on money oprn and check buy gift cards | Not access for claming the gift card | Pas s |
| Move | Choose Language Send feedback | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Choose language and send feedback | Unable to send feedba ck layout | Pas s |
| Profile view - activity | Layout Add review Add photo Edit profile Review | Manual approach | Username /Password | Operating system/ Brower | Not Required | No | Profile view Add review review | Unable to send the review of users (Proxi review) | Pas s |
| Your rating | Understand Your rating Tour rayting eduoted OK. | Manual Approach | Username /Password | Operating system/ Brower | Not Required | No | Your rating Check whether your rating is ok is there caught | Unable to see your item rating | Pas s |

EXP NO : 8

Case study on Banking Applications - Positive Test Cases

To write a case study to identify & analyze the requirements of Banking Applications and to write a positive Test Scenarios as well as Test cases for the above application.

| ITEMS TO BE TESTED | FEATURES | APPROACH | DATA TEST | TEST ENVIRONMENT | TRAINING NEED ED | RI SK | TEST CASE | PASS/ FAIL |
|--------------------|-----------------------|-----------------|--------------------|------------------|------------------|-------|---|------------|
| User Registration | Name, Email, Password | Manual approach | Username, password | Laptop, phone | NO | NO | Step1: Verify that user can successfully register with valid information. Step2: Verify that all mandatory fields are validated properly. Step3: Verify the unique username is required during registration | Pass |
| User Login | Email, Password | Manual approach | Username, password | Laptop, phone | NO | NO | Step1: Verify that a registered user can login with valid credentials. | Pass |

| | | | | | | | | |
|----------------------------|-----------------------------|------------------------|-------------------------------|------------------|----|----|---|------|
| | | | | | | | Step2: Verify that appropri ate error message are displaye d. Step3: Verify the system maintain s user sessions and keeps the user logged | |
| Accoun t manage ment | Accoun t informa tion | Manual approac h | Usern ame, passw ord | Laptop, phone | NO | NO | Step1: Verify that user can create a bank account or not. Step2: Verify user can view account details includin g balance and transacti on history. Step3: Verity that user update informati on. | Pass |

| | | | | | | | | |
|---------------|------------------------------|-----------------|---------------------|---------------|----|----|---|------|
| Fund transfer | Balance history, transaction | Manual approach | User name, password | Laptop, phone | NO | NO | Step1: Verify that user can transfer money or not' Step2: Verify that the transferred amount is accurately reflected in sender or receiver. Step3: Proper validation is performed for account numbers and transfer amounts. | Pass |
| Bill payment | Electricity water, phone | Manual approach | User name, password | Laptop, phone | NO | NO | Step1: Verify that a user can pay bills successfully Step2: Verify that the payment is reflected in the user's transaction history of | Pass |

| | | | | | | | | |
|------------------|--------------------|-----------------|--------------------|---------------|----|----|---|------|
| | | | | | | | biller's record. Step3: Verify that appropriate error messages are displayed for invalid bill payment attempts. | |
| Account security | Length, complexity | Manual approach | Username, password | Laptop, phone | NO | NO | Step1: User can update their password successfully. Step2: Verify that new password meets the security. Step3: It access two-factors authentication for enhanced security. | Pass |
| Customer support | Chat, email, phone | Manual approach | Username, password | Laptop, phone | NO | NO | Step1: User can conduct customer support via various channels. Step2: | Pass |

| | | | | | | | | |
|----------------|----------------------------|-----------------|---------------------|---------------|----|----|---|------|
| | | | | | | | Customer support requests are handled promptly and efficiently. Step3: User receives the appropriate response. | |
| Mobile banking | User friendly , mobile app | Manual approach | User name, password | Laptop, phone | NO | NO | Step1: Mobile banking applications provides a seamless and user-friendly experience. Step2: All functionalities available on the web-platform. Step3: Mobile app supports various mobile devices. | Pass |
| Transaction | Transfer, bill payment | Manual approach | User name, | Laptop, phone | NO | NO | Step1: User received | Pass |

| | | | | | | | | |
|------------------------------------|-------------------------------------|-----------------|---------------------|---------------|----|----|---|------|
| notification | t, account change | | password | | | | real notification for transaction. Step2: Notification contain accurate and relevant information. Step3: Verify the user can customize the notifications | |
| Integration with external services | External services , payment gateway | Manual approach | User name, password | Laptop, phone | NO | NO | Step1: Banking application integrate smoothly with external services. Step2: Data exchange between the applications Step3: Relevant information for external services | Pass |

EXP NO : 9

Case study on Banking Applications - Negative Test Cases

To write a case study to identify & analyze the requirements of Banking Applications and to write a negative Test Scenarios as well as Test cases for the above application

| Items to be tested | Features | Approach | Data Test | Test Environment | Training Need ed | Risk | Testcase 9 | | Pass/Fail |
|---------------------------|-----------------------|-----------------|--------------------|-------------------------|-------------------------|-------------|---|---|------------------|
| User Registration | Name, email, password | Manual Approach | Username, Password | Laptop, Phone | NO | NO | Step1- verify that user can successfully register with valid information Step2- Verify that all mandatory fields are validated Properly Step3- Verify that unique user name is required during registration. | The application should display on appropriate error message indicating that the password entered is invalid | pass |
| User Login | Email, password | Manual Approach | Username, Password | Laptop, Phone | NO | NO | Step1- verify that a registered user can login with valid credentials. Step2- Verify that appropriate error message is | Password is incorrect | Pass |

| | | | | | | | | | |
|--------------------|---------------------|-----------------|---------------------|---------------|----|----|--|-----------------------|------|
| | | | | | | | displayed. Step3- Verify that the system maintains user session and keeps the user logged | | |
| Account management | Account information | Manual Approach | User name, Password | Laptop, Phone | NO | NO | Step1- verify that user can create a bank account or not Step2- Verify that user can view account details including balance and transaction history Step3- Verify that user can update account information | Account no is invalid | Pass |

| | | | | | | | | | |
|------------------|---------------------------|-----------------|--------------------|---------------|----|----|---|--|------|
| Bill payment | Electricity, Water, phone | Manual Approach | Username, Password | Laptop, Phone | NO | NO | <p>Step1- verify that user can pay bill successfully</p> <p>Step2- Verify that payment is reflected in the user's transaction history of biller's records.</p> <p>Step3- Verify that appropriate error message is displayed.</p> | Bill was not found in transaction list | Pass |
| Account Security | Length Complexity | Manual Approach | Username, Password | Laptop, Phone | NO | NO | <p>Step1- verify that user can update their password successfully</p> <p>Step2- Verify that new password meets the security</p> <p>Step3- Verify that it access two factors authentication.</p> | The password entered is incorrect | Pass |

EXP NO : 10

Library Management System-Test Plan

Objective: Develop a comprehensive test plan document for a chosen application, such as a Library Management System, outlining the various testing strategies, methodologies, and criteria to ensure thorough evaluation and validation of the system's functionality.

| Items to be tested | features | Approach | Data test | Test environment | Training needed | Risk | Test cases | Pass/fail |
|------------------------|-----------------------|-----------------|-------------------|------------------|-----------------|------|--|-----------|
| 1.Registration | User's authentication | Manual approach | Username password | Browser | No | No | 1.Open the website 2.enter the username/password 3.login 4.home page will appear | Pass |
| 2.Searching for a book | Book identification | Manual approach | Book categories | Browser | No | No | Step1: In the home page , click the category. Step2: select the required category. Step3: choose the book from that category. | Pass |
| 3. Borrowing a book | Book borrowing | Manual approach | Book selection | Browser | No | No | Step1: in the home page , click the category. Step2: select the required category. Step3: choose the book from the category. Step4: Borrow the book | Pass |
| 4. Returning a book | Book returning | Manual approach | Book selection | Browser | No | No | Step1: in the home page , click the category. | Pass |

| | | | | | | | | |
|---|---------------------------------|-----------------|-------------------|---------|----|----|---|------|
| | | | | | | | Step2: select the return option. Step3: return the book. | |
| 5.Adding a new book the library | New book addition | Manual approach | Book availability | Browser | No | No | Step1: select the category of that book Step2: add the number for that book. Step3: add the book to the library . | Pass |
| 6. No. of book in the library | Number of books | Manual approach | Number Of books | Browser | No | No | Step1: add the category in the library Step2: add all the books in all category. Step3: No. of books will be identified. | Pass |
| 7.Generating a report of borrowed books | Generalized report of borrowing | Manual approach | No .of books | Browser | No | No | Step1: select the borrowed books Step2: select the date ,in which the report need to generate. Step3: report will be generate. | Pass |
| 8. Book-id number | Identification for the book | Manual approach | No. of books | Browser | No | No | Step1: select the required book. Step2: books contain a id number to categorize. Step3: click the Book-id number | Pass |

| | | | | | | | | |
|----------------|---------------------------------------|----------------------------|--------------------------------|---------|----|----|---|------|
| 9. Security | Authentic ation purpose | Manua l approa ch | Access control | Browser | No | No | Step1: click the security, in the home page. Step2: select the security options. Step3: keep the security of user's wish. | Pass |
| 10. Payment | Renewal purpose /members hip | Manua l approa ch | Userna me / passwo rd | Browser | No | No | Step1: click the payment option in the home page. Step2: if renewal-do payment Step3: If membership- fill details and pay. | Pass |

EXP NO : 11

Demonstrate the working of Junit to reverse a word and using assert statement for Proof of the value

AIM : To understand the working of Junit assert statements by comparing the reversed value with expected one.

```
import static org.junit.Assert.assertEquals;
import java.util.Scanner;
class saveethaTest
{
    public static void main(String[] args)
    {
        String str;
        char ch;
        Scanner sc=new Scanner(System.in);
        System.out.print("Enter a string : ");
        str=sc.nextLine();
        System.out.println("Reverse of a String "+str+" is :");
        for(int j=str.length();j>0;--j)
        {
            System.out.print(str.charAt(j-1));
            assertEquals("mani",str);
        }
        assertEquals("mani",str);
    }
}
```

OUTPUT

| Input | Actual output |
|--------------|----------------------|
| mani | inam |

Test cases:

Test case no: 1

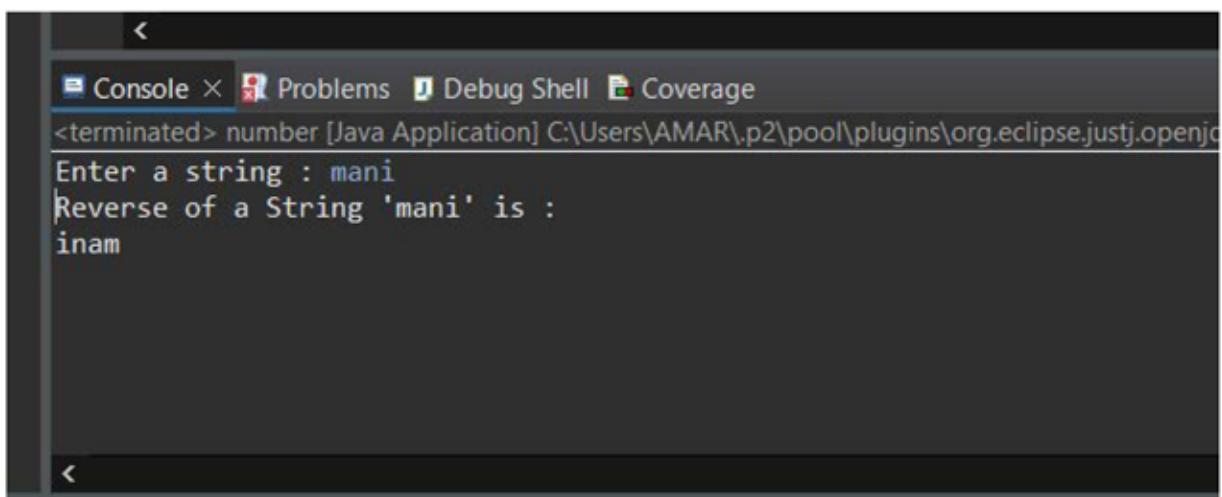
Test case name: Expected one same as actual one

| Input = mani | Expected output | Actual output | Remarks |
|---------------------|------------------------|----------------------|----------------|
| | inam | inam | SUCCESS |

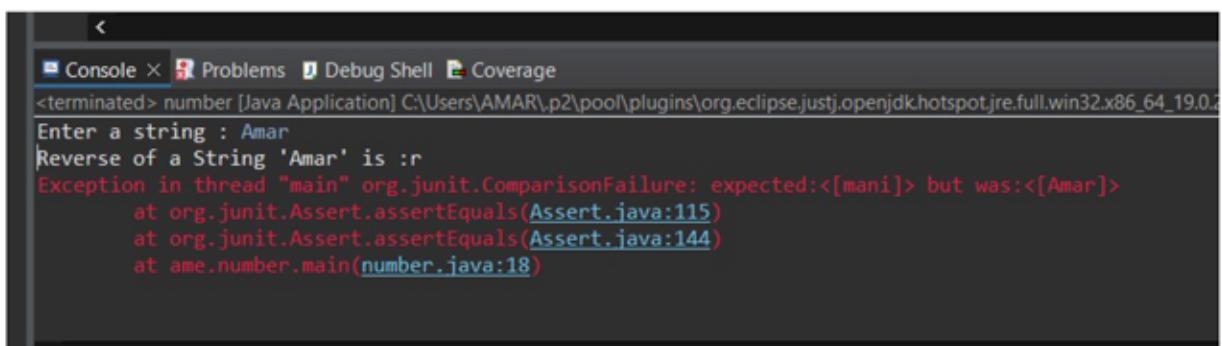
Test case no: 2

Test case name: Expected one same as actual one

| Input = Amar | Expected output | Actual output | Remarks |
|---------------------|------------------------|----------------------|----------------|
| | rama | r | FAILURE |



```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org.eclipse.justj.openj
Enter a string : mani
Reverse of a String 'mani' is :
inam
```



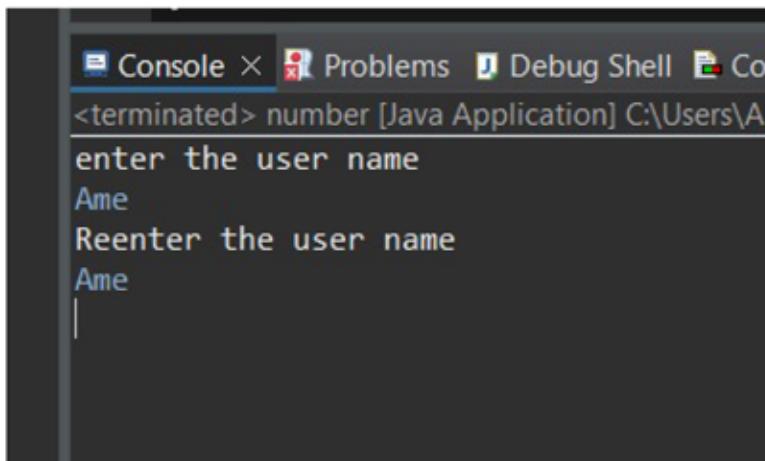
```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_19.0.2
Enter a string : Amar
Reverse of a String 'Amar' is :r
Exception in thread "main" org.junit.ComparisonFailure: expected:<[mani]> but was:<[Amar]>
    at org.junit.Assert.assertEquals(Assert.java:115)
    at org.junit.Assert.assertEquals(Assert.java:144)
    at ame.number.main(number.java:18)
```

EXP NO : 12

Write a white box testing code (junit) to String comparison of word and using assert statement for Proof the value.

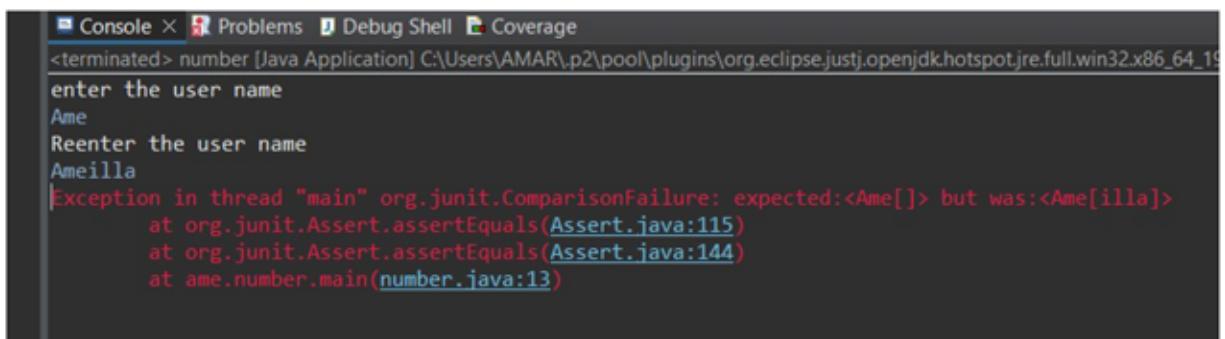
AIM: To understand the working of Junit assert statements by comparing two strings.

```
import static org.junit.Assert.assertEquals;
import java.util.Scanner;
public class third {
    public static void main(String [] args)
    {
        Scanner in=new Scanner(System.in);
        System.out.println("enter the user name");
        String str1=in.nextLine();
        System.out.println("Reenter the user name");
        String str2=in.nextLine();
        assertEquals(str1,str2);
    }
}
```



The screenshot shows the Eclipse IDE's Console view. It displays the output of a Java application named 'number'. The application prompts the user to enter a name, which is typed as 'Ame'. It then asks the user to re-enter the name, also typed as 'Ame'. The console output is as follows:

```
<terminated> number [Java Application] C:\Users\AMAR
enter the user name
Ame
Reenter the user name
Ame
```



The screenshot shows the Eclipse IDE's Console view. It displays the output of the same Java application, but this time the second input ('Reenter the user name') is typed as 'Ameilla'. This causes a failure in the assertEquals() assertion, resulting in a ComparisonFailure exception. The console output is as follows:

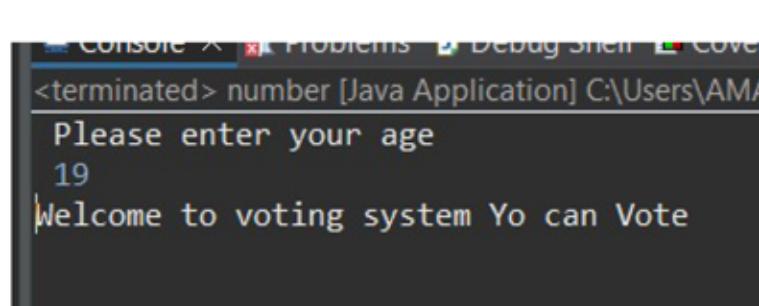
```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_19
enter the user name
Ame
Reenter the user name
Ameilla
Exception in thread "main" org.junit.ComparisonFailure: expected:<Ame[]> but was:<Ame[illa]>
    at org.junit.Assert.assertEquals(Assert.java:115)
    at org.junit.Assert.assertEquals(Assert.java:144)
    at ame.number.main(number.java:13)
```

EXP NO : 13

Write a junit code for voting system and uses assert statement and verify the white box testing.

AIM: To understand the working of Junit True statements by checking the voting age.

```
import static org.junit.Assert.assertTrue;
import java.util.Scanner;
class four
{
public static void main(String[] args)
{
int age,shrt;
Scanner scan = new Scanner(System.in);
System.out.println(" Please enter your age");
age = scan.nextInt();
if(age>=18)
{
System.out.println("Welcome to voting system Yo can Vote");
}
else
{
    shrt= (18 - age);
System.out.println("Sorry,You can vote after :" + shrt + " years");
assertTrue(age==shrt);
}
}
```



```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org  
Please enter your age  
19  
Welcome to voting system Yo can Vote
```



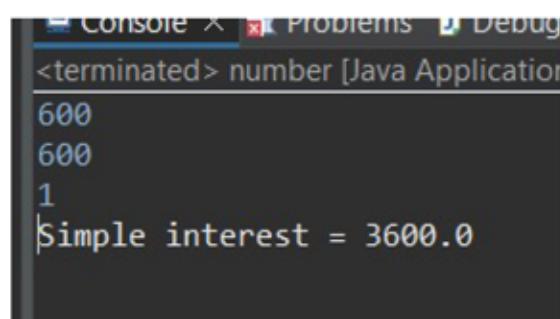
```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org  
Please enter your age  
15  
Sorry,You can vote after :3 years  
Exception in thread "main" java.lang.AssertionError  
at org.junit.Assert.fail(Assert.java:86)  
at org.junit.Assert.assertTrue(Assert.java:41)  
at org.junit.Assert.assertTrue(Assert.java:52)  
at ame.number.main(number.java:21)
```

EXP NO : 14

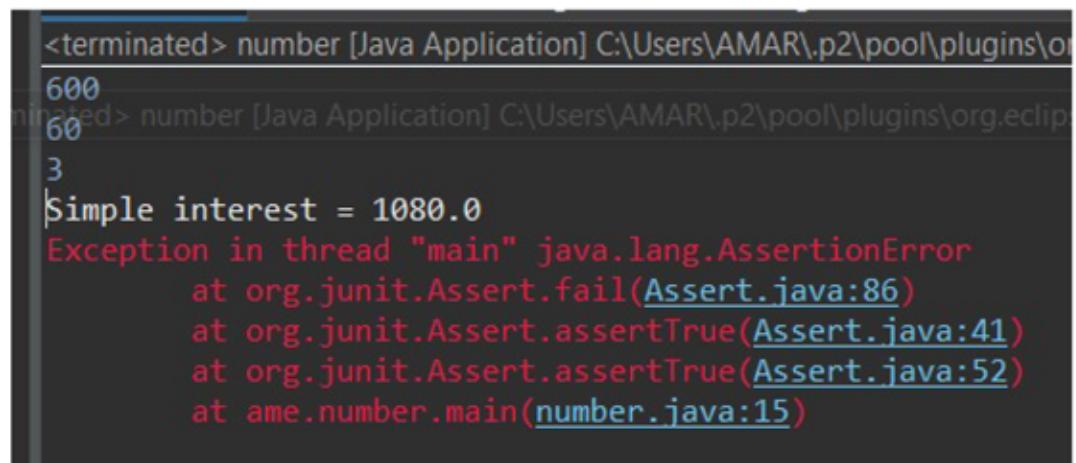
Write a program using function to calculate the simple interest. Suppose the customer is a senior citizen. He is being offered 12 percent rate of interest; for all other customers, the ROI is 10 percent. The output values should verify using white box testing.

AIM: Write a program that calculates the simple interest based on the percentage rate conditions and verify the result using assertTrue code.

```
import static org.junit.Assert.assertTrue;
import java.util.Scanner;
class intrest
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        float P=sc.nextFloat();
        float R=sc.nextFloat();
        float T=sc.nextFloat();
        float SI = (P * T * R) / 100;
        System.out.println("Simple interest = " + SI);
        assertTrue(3600==SI);
    }
}
```



```
<terminated> number [Java Application]
600
600
1
Simple interest = 3600.0
```



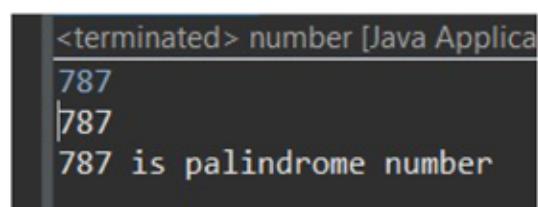
```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org.eclip
600
60
3
Simple interest = 1080.0
Exception in thread "main" java.lang.AssertionError
at org.junit.Assert.fail(Assert.java:86)
at org.junit.Assert.assertTrue(Assert.java:41)
at org.junit.Assert.assertTrue(Assert.java:52)
at ame.number.main(number.java:15)
```

EXP NO : 15

Check whether the given number is palindrome or not and verify the output values should verify using white box testing.

AIM: To check whether the given number is palindrome or not and verify the result using assertTrue code.

```
import java.util.Scanner;
import static org.junit.Assert.assertTrue;
public class palindrome
{
    public static void main(String args[])
    {
        Scanner in = new Scanner(System.in);
        int r, sum = 0, temp; int n = in.nextInt();
        temp = n;
        while (n > 0)
        {
            r = n % 10; n = n / 10;
            sum = (sum * 10)+r;
        }
        System.out.println(sum);
        assertTrue(787==sum);
        if(temp==sum)
            System.out.println(sum+" is palindrome number");
        else
            System.out.println(sum+" is not palindrome number");
    }
}
```



A terminal window showing the execution of a Java application. The output shows the input number 787, the calculated sum 787, and the message "787 is palindrome number".

```
<terminated> number [Java Application]
787
787
787 is palindrome number
```

```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\or  
143  
341  
Exception in thread "main" java.lang.AssertionError  
    at org.junit.Assert.fail(Assert.java:86)  
    at org.junit.Assert.assertTrue(Assert.java:41)  
    at org.junit.Assert.assertTrue(Assert.java:52)  
    at ame.number.main(number.java:18)
```

EXP NO : 16

Write a program to convert Decimal number equivalent to Binary number and octal numbers? The output values should verify using white box testing?

AIM: To convert the decimal number to its equivalent binary number and octal number and the output values verified using Assert code.

```
import static org.junit.Assert.assertTrue;  
  
import java.util.Scanner;  
class binary  
  
{  
    public static void main(String[] args)  
    { Scanner in = new Scanner(System.in);  
        // decimal number  
        int decimal = in.nextInt();  
        // convert decimal to binary  
        String binary = Integer.toBinaryString(decimal);  
  
        System.out.println("BINARY IS " + binary);  
        //convert decimal to octal  
        System.out.print("OCTAL IS ");  
        System.out.println(Integer.toOctalString(decimal));  
        //assertEquals("1100" , binary);  
        assertTrue(14== decimal);  
  
    }  
}
```

```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org  
14  
BINARY IS 1110  
OCTAL IS 16
```

```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org  
15  
BINARY IS 1111  
OCTAL IS 17  
Exception in thread "main" java.lang.AssertionError  
at org.junit.Assert.fail(Assert.java:86)  
at org.junit.Assert.assertTrue(Assert.java:41)  
at org.junit.Assert.assertTrue(Assert.java:52)  
at ame.number.main(number.java:21)
```

EXP NO : 17

Write a Java Program to Convert a Given Number of Days in Terms of Years, Weeks & Days. The output values should verify using white box testing.

AIM:

```
import static org.junit.Assert.assertTrue;
import java.util.Scanner;
public class year
{
    public static void main(String args[])
    {
        int m, year, week, day;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number of days:");
        m = s.nextInt();
        year = m / 365;
        assertTrue(2==year);
        m = m % 365;
        System.out.println("No. of years:"+year);
        week = m / 7;
        m = m % 7;
        System.out.println("No. of weeks:"+week);
        day = m;
        System.out.println("No. of days:"+day); }
```

```
<terminated> number [java Application] c
Enter the number of days:800
No. of years:2
No. of weeks:10
No. of days:0
```

```
<terminated> number [java Application] C:\Users\AMAR\pz\poor\plugins\
Enter the number of days:500
No. of years:1
No. of weeks:19
No. of days:2
Exception in thread "main" java.lang.AssertionError
    at org.junit.Assert.fail(Assert.java:86)
    at org.junit.Assert.assertTrue(Assert.java:41)
    at org.junit.Assert.assertTrue(Assert.java:52)
    at ame.number.main(number.java:22)
```

EXP NO : 18

Find the factorial of n? The output values should verify using white box testing?

AIM:

```
import static org.junit.Assert.assertTrue;
import java.util.Scanner;
class factorial
{
    public static void main(String[] args)
    {
        int i,j,pr=1;
        try{
            Scanner s=new Scanner(System.in);
            System.out.println("Enter the number to find the factorial");
            int n=s.nextInt();
            if(n<0)
            {
                System.out.println("Invalid");
            }
            else if(n==0)
            {
                System.out.println("1");
            }
            else
            {
                for(i=n;i>0;i--)
                {
                    pr=pr*i;
                }
                System.out.println("The answer is:"+pr);
                assertTrue(120==pr);
            }
        }
        catch(Exception e)
        {
            System.out.println("Invalid");
        }
    }
}
```

```
<terminated> number [Java Application] C:\Users\AMAN\OneDrive\Desktop\Java\src\com\javatpoint\factorial.java
Enter the number to find the factorial
5
The answer is:120
```

```
terminated> number [java Application] C:\Users\AMAR\b2\poor\plugins\o  
Enter the number to find the factorial  
6  
The answer is:720  
Exception in thread "main" java.lang.AssertionError  
at org.junit.Assert.fail(Assert.java:86)  
at org.junit.Assert.assertTrue(Assert.java:41)  
at org.junit.Assert.assertTrue(Assert.java:52)  
at ame.number.main(number.java:29)
```

EXP NO : 19

Find the year of the given date is leap year or not .The output values should verify using white box testing?

AIM: To find the year of the given date is leap year or not and the result is verified using whitebox testing.

```
import static org.junit.Assert.assertTrue;
import java.util.Scanner;
class leapyear
{
    public static void main(String[] args)
    {
        int i=0;
        System.out.println("Enter the date/month/year");
        Scanner s=new Scanner(System.in);
        String re=s.nextLine();
        String[] r=re.split("/",3);
        int x=Integer.parseInt(r[2]);
        assertTrue(x==2000);
        if(x%4==0)
        {
            System.out.println("It is a leap year");
        }
        else{
            System.out.println("It is not a leap year:");
        }
    }
}
```

```
<terminated> number [Java Application] C:\Users\AMAN
Enter the number to find the factorial
5
The answer is:120
```

```
<terminated> number [Java Application]
Enter the date/month/year
12/04/2000
It is a leap year
```

EXP NO : 20

Write a program to find the square, cube of the given decimal number. The output values should verify using white box testing

AIM: To write a program to find the square, cube of the given decimal number. The output values should verify using white box testing.

```
import static org.junit.Assert.assertTrue;
import java.util.Scanner;
public class CubeSquare{
    public static void main(String[] args)
    {
        try{
            Scanner s=new Scanner(System.in);
            System.out.println("Enter an number");
            double n=s.nextDouble();
            double a=0,b=0;
            a=n*n;
            b=n*n*n;
            System.out.println("The square of number="+a);
            System.out.println("The square of number="+b);
        }
        catch(Exception e)
        {
            System.out.println("Invalid");
        }
    }
    assertTrue(expected output==a);
    assertTrue(expected output ==b);
}
```

```
<terminated> number [Java Application]
Enter an number
5
The square of number=25.0
The square of number=125.0
```

```
<terminated> number [Java Application] C:\Users\AMAR\p2\pool\plugins\org.
Enter an number
25
The square of number=625.0
The square of number=15625.0
Exception in thread "main" java.lang.AssertionError
    at org.junit.Assert.fail(Assert.java:86)
    at org.junit.Assert.assertTrue(Assert.java:41)
    at org.junit.Assert.assertTrue(Assert.java:52)
    at ame.number.main(number.java:16)
```

EXP NO : 21

Write a Selenium program for automation and open a chrome browser with google.com

AIM: To write a Selenium program to automate the process of opening a Chrome browser and navigating to the website "google.com."

The specific requirements for this program are as follows:

Selenium Automation:

The program should utilize the Selenium framework to automate browser interactions. Selenium provides a set of libraries and tools for browser automation, allowing us to simulate user actions and perform tasks programmatically.

Open Chrome Browser:

The program should initiate and open a Chrome browser instance. It should utilize the WebDriver component of Selenium to establish a connection with the browser. WebDriver acts as a bridge between the automation script and the browser, allowing control and interaction with the browser's functionalities.

Navigate to "google.com":

The program should instruct the Chrome browser to navigate to the website "google.com." It should utilize WebDriver commands to perform this action, such as `driver.get ("https://www.google.com")`, where `driver` is the WebDriver instance.

Verification:

The program should ensure that the browser successfully opens and navigates to "google.com" without any errors. It should validate the expected behavior by checking that the page title or a specific element on the page is present.

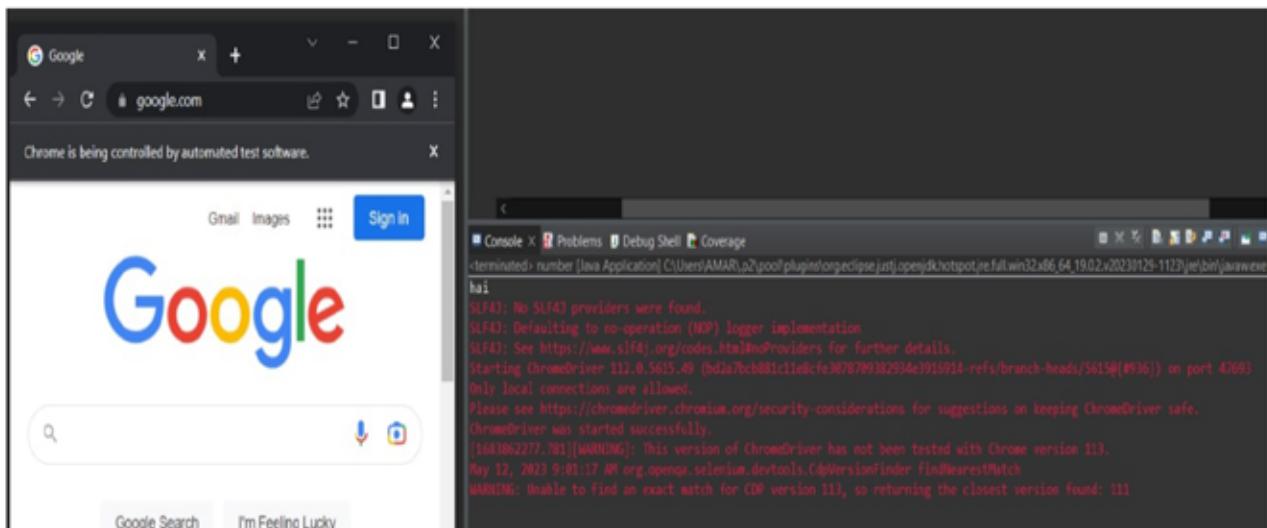
PROGRAM

```
package ame;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.WebElement;  
import org.openqa.selenium.chrome.ChromeDriver;
```

```

public class CHROME{
    public static void main(String[] args) {
        System.out.println("hai");
        System.setProperty("webdriver.chrome.driver","C:\\\\chromedriver_win3
2\\\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.google.com/");
    }
}

```



EXP NO : 22

Write a program for automation and open a Mozilla browser with google.com

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;

public class SeleniumExample {
    public static void main(String[] args) {
        // Set the path to the geckodriver executable
        System.setProperty("webdriver.gecko.driver",
"/path/to/geckodriver");

        // Create a new instance of the Firefox driver
        WebDriver driver = new FirefoxDriver();

        // Navigate to Google
        driver.get("https://www.google.com");

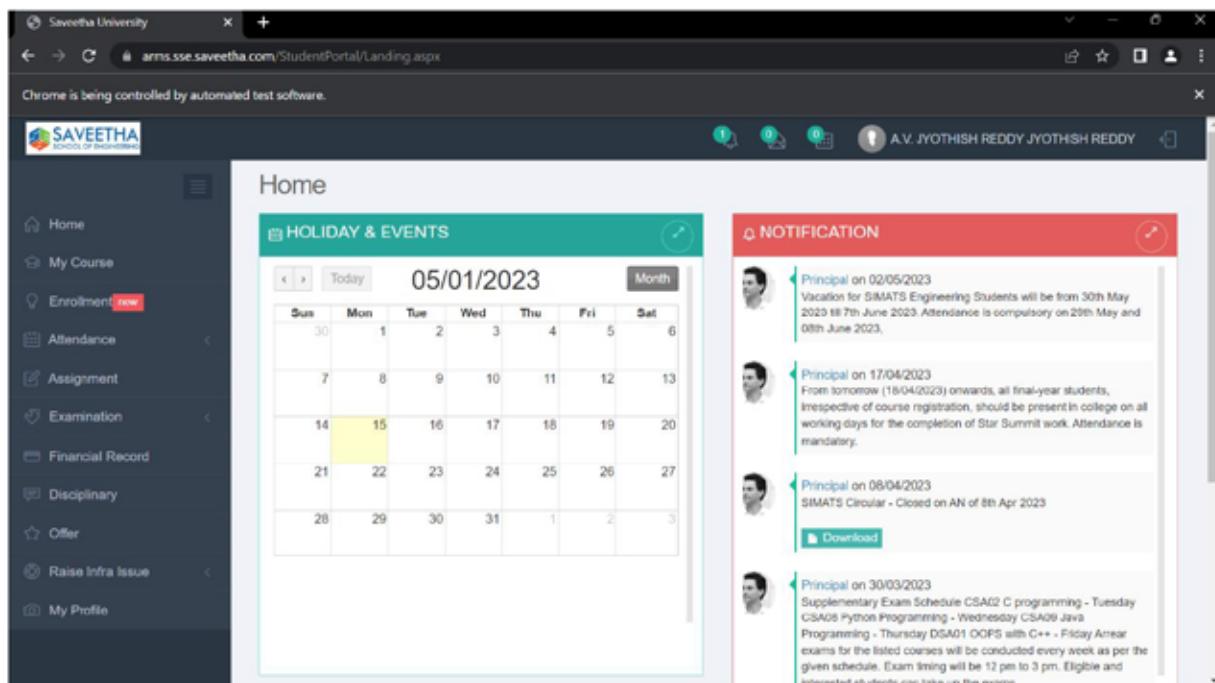
        // Close the browser
        driver.quit();
    }
}
```



EXP NO : 23

Selenium program to Automate login in ARMS Portal with the help of Chrome browser

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class loginpage {
    public static void main(String[] args) {
        System.out.println("hai");
        System.setProperty("webdriver.chrome.driver", "C:\\selenium\\chromedriver_win32\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://arms.sse.saveetha.com");
        WebElement username=driver.findElement(By.id("txtusername"));
        System.out.println(username);
        WebElement password=driver.findElement(By.id("txtpassword"));
        System.out.println(password);
        WebElement login=driver.findElement(By.name("btnlogin"));
        System.out.println(login);
        username.sendKeys(".XXXXX");
        password.sendKeys("XXXXX");
        login.click();
    }
}
```



EXP NO : 24

Selenium program Automate login in Github Portal with the help of Chrome browser

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class SeleniumExample {
    public static void main(String[] args) {
        // Set the path to the chromedriver executable
        System.setProperty("webdriver.chrome.driver", "/path/to/chromedriver");

        // Create a new instance of the Chrome driver
        WebDriver driver = new ChromeDriver();

        // Navigate to GitHub
        driver.get("https://github.com");

        // Close the browser
        driver.quit();
    }
}
```

The screenshot shows the GitHub homepage with a search bar at the top. Below the search bar, there's a navigation bar with links for Product, Solutions, Open Source, and Pricing. On the right side of the navigation bar are 'Search', 'Sign in', and 'Sign up' buttons. Below the navigation bar, there are links for Explore, Topics, Trending, Collections, Events, and GitHub Sponsors. To the right of these links is a 'Get email updates' button.

In the center, there's a search results section for the topic '#home-page'. It shows a single repository result:

- ant-design / ant-design-landing**: A repository with 5.4k stars.
- Description: Landing Pages of Ant Design System.
- Tags: react, editor, template, design, landing-page, templates, homepage, site, landing, landings, home-page, landingpage, landing-pages, page-editor.
- Last updated: 3 weeks ago.
- Language: JavaScript.

To the right of the repository details, there are buttons for 'Star' and 'Improve this page'. Below these buttons, there's a section titled 'Add this topic to your repo' with instructions on how to associate a repository with the topic.

EXP NO : 25

Write a Selenium program Automate login in Swiggy Portal with the help of Chrome browser.

```
package Proj;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.WebElement;  
import org.openqa.selenium.chrome.ChromeDriver;  
  
public class loginpage {  
    public static void main(String s[]) {  
  
        WebDriver driver = new ChromeDriver();  
        System.setProperty("webdriver.chrome.driver", "C://chromedriver.exe");  
        driver.get("https://www.swiggy.com/?utm_source=Google-Sok&utm_medium=CPC&utm_campaign=perf_food_nu_google_search_sok_narm_brand_v1_v2_march23_brand_em&gclid=CjwKCAjwYKjBh5EiwAiFdSfrdLVA-UKWYVK60DxCwxHLKmh3lOfjK9ya_mDxHMfbN5Waws1oQUvh0CNFsQAvD_BwE");  
  
        WebElement textBox = driver.findElement(By.className("x4bK8"));  
  
        driver.manage().window().maximize();  
        textBox.click();  
    }  
}
```

