

# GUI Testing

→ All the webpages in the Online Course Management System.

1. Login
2. Sign In
3. dashboard(student , instructor) -
  - a. Announcement
    - i. new announcement
  - b. Material
    - i. new material
  - c. Assignment
    - i. new assignment
  - d. chat
  - e. Feedback
4. Profile
  - a. edit Profile
5. Create course(instructor)
6. Join Course(student)

## 1. Log IN

Functionality	Workability
1. Text	UI done
2. Text-field	Password- no special character is needed. Email should be asked from the user instead of Username as username is not asked while registering.
3. Dropdown	not present
4. Buttons	good UI. Login button not working.
5. Check-box	not present
6. Radio- Button	not present
7. Hyperlink	No issue

8. Image	not present
9. Num-input fields	Not present
10. Scroll bar	Not present(not needed)
11. Message	password invalid or account not exist message present
12. Logo	does not direct to home page(not needed)
13. Menus its operation	not present
14. Zoom	zoom function is fine
15. General	all good

## 2. Sign In

Functionality	Workability
16. Text	UI done but can be better(less margin present)
17. Text-field	Password field needs to be written instead of Confirm Password. Necessary fields not specified.
18. Dropdown	Present but options are not proper in 'Batch' field. In OTP, dropdown is not needed.
19. Buttons	good UI. Register button gives error when clicked.
20. Check-box	Not present
21. Radio- Button	not present
22. Hyperlink	no issue
23. Image	not present
24. Num-input fields	all right
25. Scroll bar	present

26. Message	all message related to text field present
27. Logo	does not direct to home page(not needed)
28. Menus its operation	not present
29. Zoom	zoom function is fine
30. General	all good

### 3. Dashboard (instructor)

Functionality	Workability
31. Text	Warning text visible on top
32. Text-field	done
33. Dropdown	not present
34. Buttons	Search option displays page not found. Rest buttons are working. In Edit course, remove button is not working.
35. Check-box	not present
36. Radio- Button	not present
37. Hyperlink	done
38. Image	no image
39. Num-input fields	no input fields present
40. Scroll bar	all done
41. Message	no message needed

Functionality	Workability
---------------	-------------

42. Logo	not present
43. Menus its operation	done
44. Maps	zoom done
45. General	all set

#### 4. Dashboard (student)

46. Text	All done
47. Text-field	done
48. Dropdown	not present
49. Buttons	Search option displays page not found. Rest buttons are working. No option for un-enrol from course.
50. Check-box	not present
51. Radio- Button	not present
52. Hyperlink	done
53. Image	no image
54. Num-input fields	no input fields present
55. Scroll bar	Working fine
56. Message	no message needed
57. Logo	not present
58. Menus its operation	done
59. Zoom	zoom done

60. General	all set
-------------	---------

## 5. Announcement

Functionality	Workability
61. Text	all done
62. Text-field	All okay
63. Dropdown	not present
64. Buttons	Add announcement button is working and everything fine, remove button on every announcement is working. No option for editing announcement is available(for faculty)
65. Check-box	not present
66. Radio- Button	not present
67. Hyperlink	hyperlink not present but link working fine
68. Image	not present
69. Num-input fields	no input field
70. Scroll bar	working fine and responsive
71. Message	No message
72. Logo	not present
73. Menus its operation	Hovering menu option does not change color or add other effects.
74. Maps	zoom in + out working

75. General	all well
-------------	----------

## 6. New announcement

Functionality	Workability
76. Text	all up to date and clear what to do
77. Text-field	clear information available. Mandatory Field is not with Red*. Long description can be written.
78. Dropdown	not present
79. Buttons	All working
80. Check-box	not present
81. Radio- Button	not present
82. Hyperlink	hyperlink not present but link working fine
83. Image	not present
84. Num-input fields	Not present
85. Scroll bar	Not present(not needed)
86. Message	no message during student login
87. Logo	not present
88. Menus its operation	Not applicable
89. Maps	zoom in + out working
90. General	All well

## 7. Material

Functionality	Workability
1. Text	Testing Done
2. Text-field	Mandatory Field is not with Red*. No option for uploading pdf or other type of document.
3. Dropdown	Not present
4. Buttons	<ol style="list-style-type: none"><li>1. Edit button in material page is not present</li><li>2. No <b>OK</b> button provided while error occur</li><li>3. Esc doesn't respond when alerts pop up.</li><li>4. Add button is adding material even though nothing has been specified in fields.</li></ol>
5. Check-box	Not present
6. Radio- Button	Not present
7. Hyperlink	All working.
8. Image	Not present
9. Num-input fields	Not present
10. Scroll bar	Testing Done
11. Message	NA
12. Logo	No logo
13. Menus its operation	Hovering menu option does not change color or add other effects.
14. Maps	Testing Done
15. General	Overall good

## 8. Assignment

Functionality	Workability
1. Text	Testing Done
2. Text-field	Mandatory Field is not with Red*. No option for uploading pdf or other type of document.
3. Dropdown	Present and working
4. Buttons	<ol style="list-style-type: none"><li>1. Save Changes after editing Button not available(for student)</li><li>2. No <b>OK</b> button provided while error occur</li><li>3. Esc doesn't response when alert pop up.</li><li>4. Submission gets edited without edit submission button being clicked.</li></ol>
5. Check-box	NA
6. Radio- Button	NA
7. Hyperlink	Not able to open when hovering on HyperLink in Assignment.
8. Image	NA
9. Num-input fields	NA
10. Scroll bar	Testing Done
11. Message	Error is not getting close
12. Logo	No logo
13. Menus its operation	Hovering menu option does not change color or add other effects.
14. Maps	Testing Done
15. General	Overall good

## 9. Chat



Functionality	Workability
16. Text	All good
17. Text-field	Chat is sending messages even though no text has been written in the chat field.
18. Dropdown	Not present
19. Buttons	All good
20. Check-Box	Not present
21. Radio- Button	Not present
22. Hyperlink	Not present
23. Image	Not present
24. Num-input fields	no separate num-input field.
25. Scroll bar	Working
26. Message	No Success, Warning or Error message is shown.
27. Logo	Not present
28. Menus its operation	Hovering menu option does not change color or add other effects.
29. Maps	Zoom in-out works fine.
30. General	Attractive UI, Responsive. New queries/replies can be displayed on top instead of bottom to improve user experience.

#### 10. Feedback\*

Functionality	Workability
31. Text	In consistent size of text.

32. Text-field	<p>Inconsistent color of the text fields, Text fields are going outside the desired place, and No red * symbol for mandatory fields, however has a function to check that the mandatory field has been filled.</p> <p>Grammarly like extensions can be used to check the input text.</p>
33. Dropdown	Calendar is all good.
34. Buttons	Submit button is not properly aligned to the feedback forum. Also, the submit button can not submit the feedback form.
35. Check-Box	Not present
36. Radio- Button	Not present
37. Hyperlink	Not present
38. Image	Not present
39. Num-input fields	Num-input fields are present in the number field but cannot verify that only the number is being typed.
40. Scroll bar	Works fine
41. Message	Warning message appears if feedback is already given once.
42. Logo	Not present
43. Menus its operation	Hovering menu option does not change color or add other effects.
44. Maps	Zoom in-out works fine.
45. General	All good

## 11. Profile

Functionality	Workability
46. Text	Username not displayed.
47. Text-field	Not present
48. Dropdown	Not present
49. Buttons	All good
50. Check-box	Not present
51. Radio- Button	Not present
52. Hyperlink	Not present
53. Image	Not present
54. Num-input fields	Not present
55. Scroll bar	scrollbar is not present.
56. Message	Not applicable
57. Logo	Not present
58. Menus its operation	Working
59. Maps	Zoom in-out works fine.
60. General	All good

## 12. Edit profile

Functionality	Workability
61. Text	All good

62. Text-field	Neither red * symbol for mandatory fields nor functionality to check that the mandatory field has been filled. Moreover, placeholder is not present for input fields.
63. Dropdown	Not present
64. Buttons	Has a save button but it is harder to spot due to the same color as the background.
65. Check-box	Not present
66. Radio- Button	Not present
67. Hyperlink	Not present
68. Image	Not present
69. Num-input fields	Num-input fields are present in the number field.
70. Scroll bar	Both the vertical and Horizontal scrollbar are not present.
71. Message	No message appears when num-input field has alphanumeric data.
72. Logo	Not present
73. Menus its operation	Not present
74. Maps	Zoom in-out works fine.
75. General	Overall good

### 13. Create course(instructor)

Functionality	Workability
---------------	-------------

76. Text	Text is going outside the text box
77. Text-field	Mandatory Field is not with Red*
78. Dropdown	Not present
79. Buttons	Present and working
80. Check-box	Not present
81. Radio- Button	Not present
82. Hyperlink	Not present
83. Image	Not applicable
84. Num-input fields	Not present
85. Scroll bar	Not applicable
86. Message	No message appears if fields are left blank
87. Logo	NA
88. Menus its operation	All done
89. Maps	All done
90. General	Blinking cursor symbol to enable text fields. Overall good.

#### 14. Join course(student)

Functionality	Workability
91. Text	Text is readable.
92. Text-field	Correct labels are displayed for search field.

93. Dropdown	Not present
94. Buttons	Working
95. Check-box	Not present
96. Radio- Button	Not present
97. Hyperlink	Not present and link found
98. Image	Not found
99. Num-input fields	Not applicable
100. Scroll bar	Not present
101. Message	Enroll course button opens another page to display message of success/failure instead of on the same page.
102. Logo	Not present
103. Menus its operation	All done
104. Maps	All done
105. General	All good

# AUTOMATED GUI TESTING :

Tested on Selenium Web IDE version 4.0.0 alpha59 ARM64 Build Chrome Web Extension ,  
Google Chrome version being 119.0.6045.199 (arm64)

## (1) Login and update profile :

The screenshot displays the Selenium IDE interface for a project named 'Academix portal\*'. The test suite 'login and update profile\*' is currently executing at the URL 'http://127.0.0.1:8000'. The test consists of 16 steps, each with a command, target, and value. The commands include 'open', 'set window size', 'click', and 'type'. The targets are various selectors like '/', '1512x944', 'css=#great1\_button-5', 'id=user', 'id=pass3', 'css=btn:nth-child(8)', 'css=nav-item:nth-child(1) > .nav-link', 'css=button-3', 'name=first\_name', and 'name=batch'. The values include '202101049@daict.ac.in', 'Raj@3501', 'Rajesh', and '2021'. A log at the bottom shows the execution progress, with the final status being 'login and update profile' completed successfully.

Step	Command	Target	Value
1	open	/	
2	set window size	1512x944	
3	click	css=#great1_button-5	
4	click	id=user	
5	type	id=user	202101049@daict.ac.in
6	click	id=pass3	
7	type	id=pass3	Raj@3501
8	click	css=btn:nth-child(8)	
9	click	css=nav-item:nth-child(1) > .nav-link	
10	click	css=button-3	
11	click	name=first_name	
12	type	name=first_name	Rajesh
13	click	name=batch	
14	type	name=batch	2021
15	click	css=button-3	
16	click	css=nav-item:nth-child(4) > .nav-link	

Runs: 1 Failures: 0

Log Reference

- 11. click on name=first\_name OK 06:48:31
- 12. type on name=first\_name with value Rajesh OK 06:48:31
- 13. click on name=batch OK 06:48:31
- 14. type on name=batch with value 2021 OK 06:48:31
- 15. click on css=button-3 OK 06:48:31
- 16. click on css=nav-item:nth-child(4) > .nav-link OK 06:48:31
- 'login and update profile' completed successfully 06:48:32

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestLoginandupdateprofile():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()
```

```

def test_loginandupdateprofile(self):
    # Test name: login and update profile
    # Step # | name | target | value
    # 1 | open | / |
    self.driver.get("http://127.0.0.1:8000/")
    # 2 | setWindowSize | 1512x944 |
    self.driver.set_window_size(1512, 944)
    # 3 | click | css=#great1 .button-5 |
    self.driver.find_element(By.CSS_SELECTOR, "#great1 .button-
5").click()
    # 4 | click | id=user |
    self.driver.find_element(By.ID, "user").click()
    # 5 | type | id=user | 202101049@daiict.ac.in
    self.driver.find_element(By.ID,
"user").send_keys("202101049@daiict.ac.in")
    # 6 | click | id=pass3 |
    self.driver.find_element(By.ID, "pass3").click()
    # 7 | type | id=pass3 | Raj@3501
    self.driver.find_element(By.ID, "pass3").send_keys("Raj@3501")
    # 8 | click | css=.btn:nth-child(8) |
    self.driver.find_element(By.CSS_SELECTOR, ".btn:nth-
child(8)").click()
    # 9 | click | css=.nav-item:nth-child(1) > .nav-link |
    self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(1) >
.nav-link").click()
    # 10 | click | css=.button-3 |
    self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
    # 11 | click | name=first_name |
    self.driver.find_element(By.NAME, "first_name").click()
    # 12 | type | name=first_name | Rajesh
    self.driver.find_element(By.NAME, "first_name").send_keys("Rajesh")
    # 13 | click | name=batch |
    self.driver.find_element(By.NAME, "batch").click()
    # 14 | type | name=batch | 2021
    self.driver.find_element(By.NAME, "batch").send_keys("2021")
    # 15 | click | css=.button-3 |
    self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
    # 16 | click | css=.nav-item:nth-child(4) > .nav-link |
    self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(4) >
.nav-link").click()

```



## (2). Upload material :

Project: **Academix portal\***

Tests: **upload a material\***

Command	Target	Value
1. open	/	
2. set window size	1512x944	
3. click	css=#great2 .button-5	
4. click	id=user	
5. type	id=user	faculty2@dalict.ac.in
6. click	id=pass	
7. type	id=pass	instructor@123
8. click	css=.btn	
9. click	css=div:nth-child(4) .button-5	
10. click	linkText=Material	
11. click	css=fa-regular	
12. click	name=title	
13. type	name=title	today's notes
14. click	name=description	
15. type	name=description	lecture notes for tuesday's lecture
16. click	name=myfile	
17. type	name=myfile	C:\fakepath\CAMF04_callouts_25xFreeCamTemplates_Cam9-2018-2019.zip

Log: 17. type on name=myfile with value C:\fakepath\CAMF04\_callouts\_25xFreeCamTemplates\_Cam9-2018-2019.zip Failed: Playback aborted

Reference: 'upload a material\*' was aborted

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities
```

```
class TestUploadamaterial():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_uploadamaterial(self):
        # Test name: upload a material
        # Step # | name | target | value
        # 1 | open | / |
```

```

self.driver.get("http://127.0.0.1:8000/")
# 2 | setWindowSize | 1512x944 |
self.driver.set_window_size(1512, 944)
# 3 | click | css=#great2 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great2 .button-5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | faculty2@daiict.ac.in
self.driver.find_element(By.ID, "user").send_keys("faculty2@daiict.ac.in")
# 6 | click | id=pass |
self.driver.find_element(By.ID, "pass").click()
# 7 | type | id=pass | instructor@123
self.driver.find_element(By.ID, "pass").send_keys("instructor@123")
# 8 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 9 | click | css=div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(4) .button-5").click()
# 10 | click | linkText=Material |
self.driver.find_element(By.LINK_TEXT, "Material").click()
# 11 | click | css=.fa-regular |
self.driver.find_element(By.CSS_SELECTOR, ".fa-regular").click()
# 12 | click | name=title |
self.driver.find_element(By.NAME, "title").click()
# 13 | type | name=title | today's notes
self.driver.find_element(By.NAME, "title").send_keys("today's notes")
# 14 | click | name=description |
self.driver.find_element(By.NAME, "description").click()
# 15 | type | name=description | lecture notes for tuesday's lecture
self.driver.find_element(By.NAME, "description").send_keys("lecture notes for tuesday's lecture")
# 16 | click | name=myfile |
self.driver.find_element(By.NAME, "myfile").click()
# 17 | type | name=myfile |
C:\fakepath\CAMF04_callouts_25xFreeCamTemplates_Cam9-2018-2019.zip
self.driver.find_element(By.NAME, "myfile").send_keys("C:\\fakepath\\CAMF04_callouts_25xFreeCamTemplates_Cam9-2018-2019.zip")
# 18 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 19 | click | linkText=IT313 |
self.driver.find_element(By.LINK_TEXT, "IT313").click()
# 20 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) > .nav-link").click()

```

### (3). Faculty login and add course :

Selenium IDE - Academix portal

Project: Academix portal

Tests +

Search tests...

- ✓ Add and remove announcement
- ✓ Enroll in course and raise a query and give feedback
- ✓ add an assignment
- ✓ add and update/remove a course
- ✓ answer a query
- ✓ django admin login
- ✓ enroll and unenroll in a course
- ✓ grade assignment , edit due date
- ✓ login and update profile
- ✓ search for enrolled course
- ✓ see announcements , submit assignment
- ✓ test faculty login and add course
- ✓ upload a material

http://127.0.0.1:8000

	Command	Target	Value
1	open	/	
2	set window size	1512x944	
3	click	css=#great2 .button-5	
4	click	id=user	
5	type	id=user	faculty1@dalict.ac.in
6	click	id=pass	
7	type	id=pass	instructor@123
8	click	css=.btn	
9	click	css=.nav-item:nth-child(1) > .nav-link	
10	click	id=coursename	
11	type	id=coursename	computer networks
12	click	id=courseid	
13	type	id=courseid	IT304
14	click	id=savebutton	

Command: open

Target: /

Value:

Description:

Log Reference

**open url**  
Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs.  
arguments:  
url - The URL to open (may be relative or absolute).

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestTestfacultyloginandaddcourse():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_testfacultyloginandaddcourse(self):
        # Test name: test faculty login and add course
        # Step # | name | target | value
        # 1 | open | / |
```

```

self.driver.get("http://127.0.0.1:8000/")
# 2 | setWindowSize | 1512x944 |
self.driver.set_window_size(1512, 944)
# 3 | click | css=#great2 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great2 .button-5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | faculty1@daiict.ac.in
self.driver.find_element(By.ID, "user").send_keys("faculty1@daiict.ac.in")
# 6 | click | id=pass |
self.driver.find_element(By.ID, "pass").click()
# 7 | type | id=pass | instructor@123
self.driver.find_element(By.ID, "pass").send_keys("instructor@123")
# 8 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 9 | click | css=.nav-item:nth-child(1) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(1) > .nav-link").click()
# 10 | click | id=coursename |
self.driver.find_element(By.ID, "coursename").click()
# 11 | type | id=coursename | computer networks
self.driver.find_element(By.ID, "coursename").send_keys("computer networks")
# 12 | click | id=courseid |
self.driver.find_element(By.ID, "courseid").click()
# 13 | type | id=courseid | IT304
self.driver.find_element(By.ID, "courseid").send_keys("IT304")
# 14 | click | id=description |
self.driver.find_element(By.ID, "description").click()
# 15 | type | id=description | Welcome to the world of CN
self.driver.find_element(By.ID, "description").send_keys("Welcome to the world of CN ")
# 16 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 17 | click | css=div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(4) .button-5").click()
# 18 | click | linkText=IT304 |
self.driver.find_element(By.LINK_TEXT, "IT304").click()
# 19 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) > .nav-link").click()

```

#### (4). See announcements and submit assignment :

**Project: Academix portal**

Tests: Search tests...

Command Target Value

Command	Target	Value
1 open	/	
2 set window size	1512x944	
3 click	css=#great1.button-5	
4 click	id=user	
5 type	id=user	202101030@dalict.ac.in
6 click	id=pass3	
7 type	id=pass3	chaitanya@123
8 click	css=.btn:nth-child(8)	
9 click	css=.solutions-box:nth-child(1) > div:nth-child(4).button-5	
10 click	linkText=Announcements	
11 click	linkText=Assignments	
12 click	id=work	
13 type	id=work	C:\fakepath\Week 1 - Fundamentals_1.pptx

Command type Target id=work Value C:\fakepath\Week 1 - Fundamentals\_1.pptx Description

Log Reference

**type locator value**  
Sets the value of an input field, as though you typed it in. Can also be used to set the value of combo boxes, check boxes, etc. In these cases, value should be the value of the option selected, not the visible text. Chrome only: if a file path is given it will be uploaded to the input (for type=file). NOTE: XPath locators are not supported.

arguments:  
locator - An element locator.  
value - The value to input.

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestSeeannouncementssubmitassignment():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_seeannouncementssubmitassignment(self):
        # Test name: see announcements , submit assignment
        # Step # | name | target | value
```

```

# 1 | open | / |
self.driver.get("http://127.0.0.1:8000/")
# 2 | setWindowSize | 1512x944 |
self.driver.set_window_size(1512, 944)
# 3 | click | css=#great1 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great1 .button-5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | 202101030@daiict.ac.in
self.driver.find_element(By.ID, "user").send_keys("202101030@daiict.ac.in")
# 6 | click | id=pass3 |
self.driver.find_element(By.ID, "pass3").click()
# 7 | type | id=pass3 | chaitanya@123
self.driver.find_element(By.ID, "pass3").send_keys("chaitanya@123")
# 8 | click | css=.btn:nth-child(8) |
self.driver.find_element(By.CSS_SELECTOR, ".btn:nth-child(8)").click()
# 9 | click | css=.solutions-box:nth-child(1) > div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, ".solutions-box:nth-child(1) > div:nth-child(4) .button-5").click()
# 10 | click | linkText=Announcements |
self.driver.find_element(By.LINK_TEXT, "Announcements").click()
# 11 | click | linkText=Assignments |
self.driver.find_element(By.LINK_TEXT, "Assignments").click()
# 12 | click | id=work |
self.driver.find_element(By.ID, "work").click()
# 13 | type | id=work | C:\fakepath\Week 1 - Fundamentals_1.pptx
self.driver.find_element(By.ID, "work").send_keys("C:\\fakepath\\Week 1 - Fundamentals_1.pptx")
# 14 | click | css=.button-3 |
self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
# 15 | click | css=.button-3 |
self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
# 16 | click | id=work |
self.driver.find_element(By.ID, "work").click()
# 17 | type | id=work | C:\fakepath\Week 1 - Fundamentals_2.pptx
self.driver.find_element(By.ID, "work").send_keys("C:\\fakepath\\Week 1 - Fundamentals_2.pptx")
# 18 | click | css=.button-3 |
self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
# 19 | click | css=.navbar |
self.driver.find_element(By.CSS_SELECTOR, ".navbar").click()
# 20 | click | id=work |
self.driver.find_element(By.ID, "work").click()
# 21 | type | id=work | C:\fakepath\Week 1 -Introduction.pptx
self.driver.find_element(By.ID, "work").send_keys("C:\\fakepath\\Week 1 -Introduction.pptx")
# 22 | click | linkText=IT313 |
self.driver.find_element(By.LINK_TEXT, "IT313").click()
# 23 | click | css=.nav-item:nth-child(4) > .nav-link |

```

```
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(4) >
.nav-link").click()
```

### (5). Search for enrolled course :

The screenshot shows the Selenium IDE interface for a project named 'Academix portal'. The test suite is titled 'Tests' and contains 14 steps. The step 'search for enrolled course' is selected, and its details are shown in the right pane. The command is 'open', the target is '/', and the value is 'sw'. The bottom pane shows the 'open url' command description: 'Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs. arguments: url - The URL to open (may be relative or absolute).'.

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import
DesiredCapabilities

class TestSearchforenrolledcourse():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_searchforenrolledcourse(self):
        # Test name: search for enrolled course
```

```

# Step # | name | target | value
# 1 | open | / |
self.driver.get("http://127.0.0.1:8000/")
# 2 | setWindowSize | 1512x944 |
self.driver.set_window_size(1512, 944)
# 3 | click | css=#great1 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great1 .button-
5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | 202101030@daiict.ac.in
self.driver.find_element(By.ID,
"user").send_keys("202101030@daiict.ac.in")
# 6 | click | id=pass3 |
self.driver.find_element(By.ID, "pass3").click()
# 7 | type | id=pass3 | chaitanya@123
self.driver.find_element(By.ID, "pass3").send_keys("chaitanya@123")
# 8 | click | css=.btn:nth-child(8) |
self.driver.find_element(By.CSS_SELECTOR, ".btn:nth-
child(8)").click()
# 9 | click | name=Search |
self.driver.find_element(By.NAME, "Search").click()
# 10 | type | name=Search | sw
self.driver.find_element(By.NAME, "Search").send_keys("sw")
# 11 | click | css=.button-3 |
self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
# 12 | click | css=div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(4) .button-
5").click()
# 13 | click | linkText=IT313 |
self.driver.find_element(By.LINK_TEXT, "IT313").click()
# 14 | click | css=.nav-item:nth-child(4) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(4) >
.nav-link").click()

```



## (6). Grade assignment and editing due date

Project: Academix portal

Tests +

Search tests...

http://127.0.0.1:8000

	Command	Target	Value
✓ Add and remove announcement	click	linkText=Assignments	
✓ Enroll in course and raise a query and give feedback	click	css=div:nth-child(7) > a:nth-child(1) > .button-3	
✓ add an assignment	click	id=duedate	
✓ add and update/remove a course	type	id=duedate	2023-12-05T07:41
✓ answer a query	click	css=label:nth-child(10)	
✓ django admin login	click	css=.btn	
✓ enroll and unenroll in a course	click	css=a:nth-child(2) > .button-3	
✓ grade assignment , edit due date	click	id=grade	
✓ login and update profile	type	id=grade	10
✓ search for enrolled course	click	css=.fa-plus	
✓ see announcements , submit assignment	click	linkText=IT313	
✓ test faculty login and add course	click	css=.nav-item:nth-child(3) > .nav-link	
✓ upload a material			

Command: open

Target: /

Value:

Description:

Log Reference

**open url**  
Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs.  
arguments:  
url - The URL to open (may be relative or absolute).

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestGradeassignmenteditduedate():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_gradeassignmenteditduedate(self):
        # Test name: grade assignment , edit due date
        # Step # | name | target | value
        # 1 | open | / |
        self.driver.get("http://127.0.0.1:8000/")
```

```

# 2 | setWindowSize | 1512x944 |
self.driver.set_window_size(1512, 944)
# 3 | click | css=#great2 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great2 .button-
5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | faculty2@daiict.ac.in
self.driver.find_element(By.ID,
"user").send_keys("faculty2@daiict.ac.in")
# 6 | click | id=pass |
self.driver.find_element(By.ID, "pass").click()
# 7 | type | id=pass | instructor@123
self.driver.find_element(By.ID, "pass").send_keys("instructor@123")
# 8 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 9 | click | css=div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(4) .button-
5").click()
# 10 | click | linkText=Assignments |
self.driver.find_element(By.LINK_TEXT, "Assignments").click()
# 11 | click | css=div:nth-child(7) > a:nth-child(1) > .button-3 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(7) > a:nth-
child(1) > .button-3").click()
# 12 | click | id=duedate |
self.driver.find_element(By.ID, "duedate").click()
# 13 | type | id=duedate | 2023-12-05T07:41
self.driver.find_element(By.ID, "duedate").send_keys("2023-12-
05T07:41")
# 14 | click | css=label:nth-child(10) |
self.driver.find_element(By.CSS_SELECTOR, "label:nth-
child(10)").click()
# 15 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 16 | click | css=a:nth-child(2) > .button-3 |
self.driver.find_element(By.CSS_SELECTOR, "a:nth-child(2) > .button-
3").click()
# 17 | click | id=grade |
self.driver.find_element(By.ID, "grade").click()
# 18 | type | id=grade | 10
self.driver.find_element(By.ID, "grade").send_keys("10")
# 19 | click | css=.fa-plus |
self.driver.find_element(By.CSS_SELECTOR, ".fa-plus").click()
# 20 | click | linkText=IT313 |
self.driver.find_element(By.LINK_TEXT, "IT313").click()
# 21 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) >
.nav-link").click()

```

## (7). Enroll and unenroll course :

**Project: Academix portal**

Search tests...

http://127.0.0.1:8000

	Command	Target	Value
1	open	/	
2	set window size	1512x944	
3	click	css=#great1.button-5	
4	click	id=user	
5	type	id=user	202101030@dalict.ac.in
6	click	id=pass3	
7	type	id=pass3	chaitanya@123
8	click	css=.btn:nth-child(8)	
9	click	css=.nav-item:nth-child(3) > .nav-link	
10	click	css=.solutions-box:nth-child(1) .button-5	
11	click	css=div:nth-child(5) .button-5	
12	click	css=.nav-item:nth-child(2) > .nav-link	

Command: open

Target: /

Value:

Description:

Log Reference

**open url**  
Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs.

arguments:  
url - The URL to open (may be relative or absolute).

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestEnrollandunenrollinacourse():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

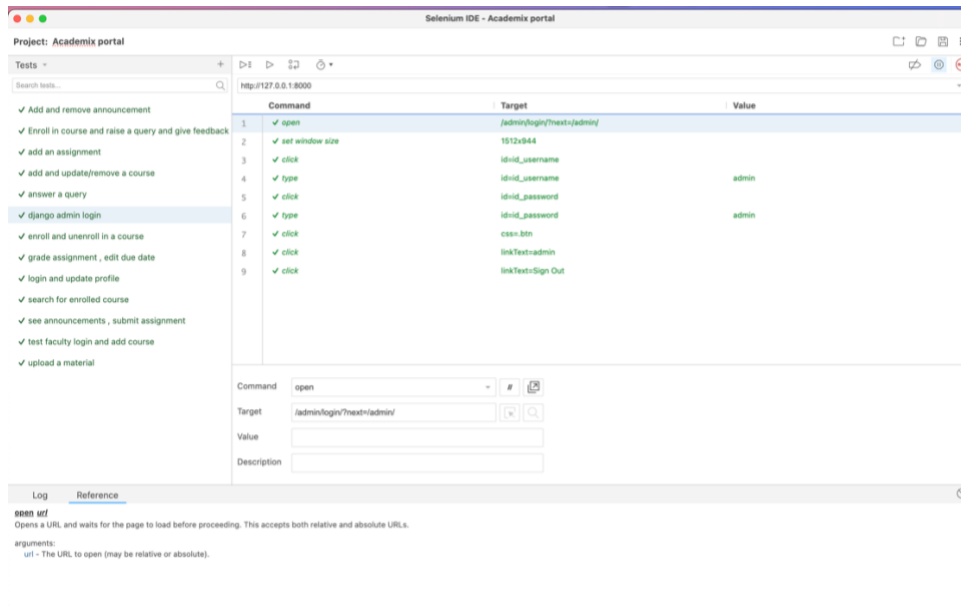
    def test_enrollandunenrollinacourse(self):
        # Test name: enroll and unenroll in a course
        # Step # | name | target | value
        # 1 | open | / |
        self.driver.get("http://127.0.0.1:8000/")
```

```

# 2 | setWindowSize | 1512x944 |
self.driver.set_window_size(1512, 944)
# 3 | click | css=#great1 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great1 .button-
5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | 202101030@daiict.ac.in
self.driver.find_element(By.ID,
"user").send_keys("202101030@daiict.ac.in")
# 6 | click | id=pass3 |
self.driver.find_element(By.ID, "pass3").click()
# 7 | type | id=pass3 | chaitanya@123
self.driver.find_element(By.ID, "pass3").send_keys("chaitanya@123")
# 8 | click | css=.btn:nth-child(8) |
self.driver.find_element(By.CSS_SELECTOR, ".btn:nth-
child(8)").click()
# 9 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) >
.nav-link").click()
# 10 | click | css=.solutions-box:nth-child(1) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, ".solutions-box:nth-
child(1) .button-5").click()
# 11 | click | css=div:nth-child(5) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(5) .button-
5").click()
# 12 | click | css=.nav-item:nth-child(2) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(2) >
.nav-link").click()

```

## (8). Django admin login :



The screenshot shows the Selenium IDE interface for a project named 'Academix portal'. The 'Tests' pane on the left lists 12 test steps, with 'django admin login' selected. The main pane displays the details for step 12: 'click' with the target '.nav-item:nth-child(2) > .nav-link' and no value. Below the table, there are input fields for Command, Target, Value, and Description. The bottom pane shows the 'Log' tab with a message indicating that the 'open' command was executed successfully.

Command	Target	Value
open	/admin/login/?next=/admin/	
set window size	1512x944	
click	id=id_username	admin
type	id=id_username	admin
click	id=id_password	admin
type	id=id_password	admin
click	css=.btn	
click	linkText=admin	
click	linkText=Sign Out	

Log  
 open url  
 Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs.  
 arguments:  
 url - The URL to open (may be relative or absolute).

```

# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import
DesiredCapabilities

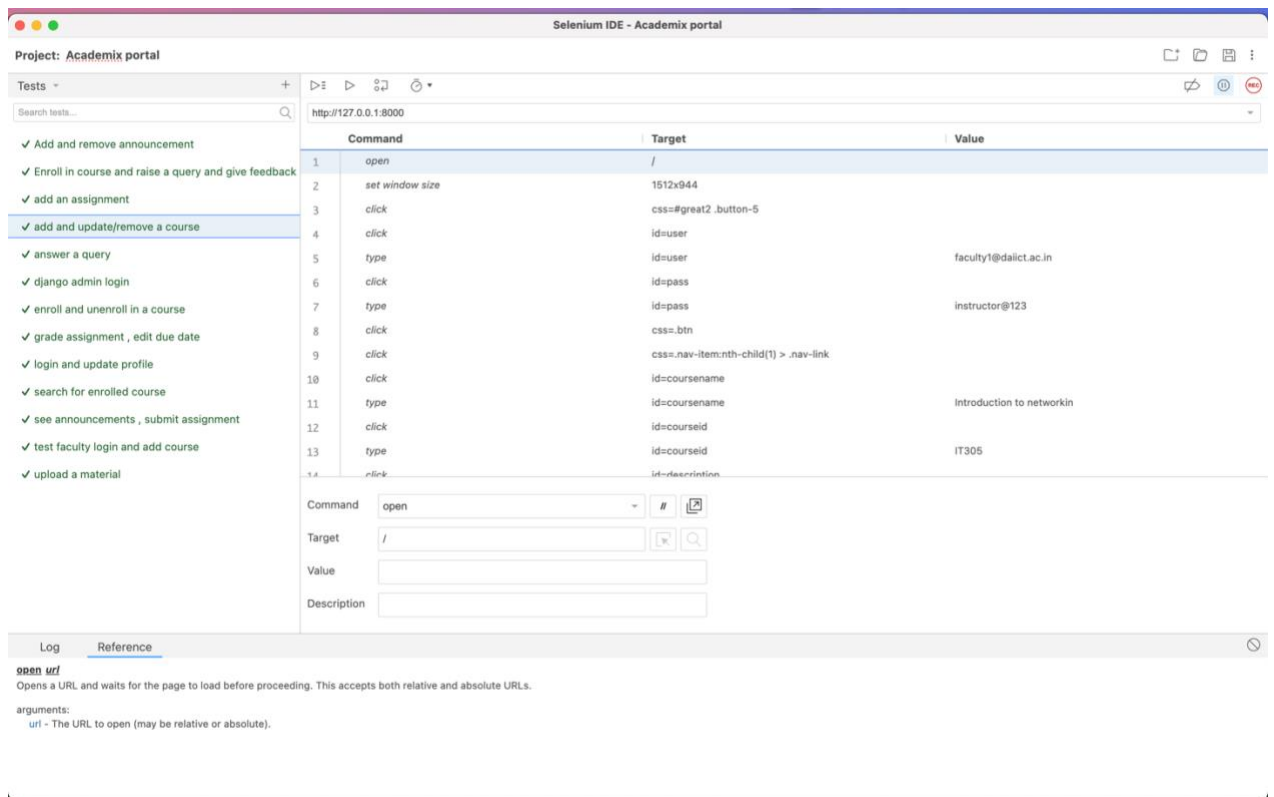
class TestDjangoadminlogin():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_djangoadminlogin(self):
        # Test name: django admin login
        # Step # | name | target | value
        # 1 | open | /admin/login/?next=/admin/ |
        self.driver.get("http://127.0.0.1:8000/admin/login/?next=/admin/")
        # 2 | setWindowSize | 1512x944 |
        self.driver.set_window_size(1512, 944)
        # 3 | click | id=id_username |
        self.driver.find_element(By.ID, "id_username").click()
        # 4 | type | id=id_username | admin
        self.driver.find_element(By.ID, "id_username").send_keys("admin")
        # 5 | click | id=id_password |
        self.driver.find_element(By.ID, "id_password").click()
        # 6 | type | id=id_password | admin
        self.driver.find_element(By.ID, "id_password").send_keys("admin")
        # 7 | click | css=.btn |
        self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
        # 8 | click | linkText=admin |
        self.driver.find_element(By.LINK_TEXT, "admin").click()
        # 9 | click | linkText=Sign Out |
        self.driver.find_element(By.LINK_TEXT, "Sign Out").click()

```

**(9). Add , update and remove course :**



```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestAddandupdateremoveacourse():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

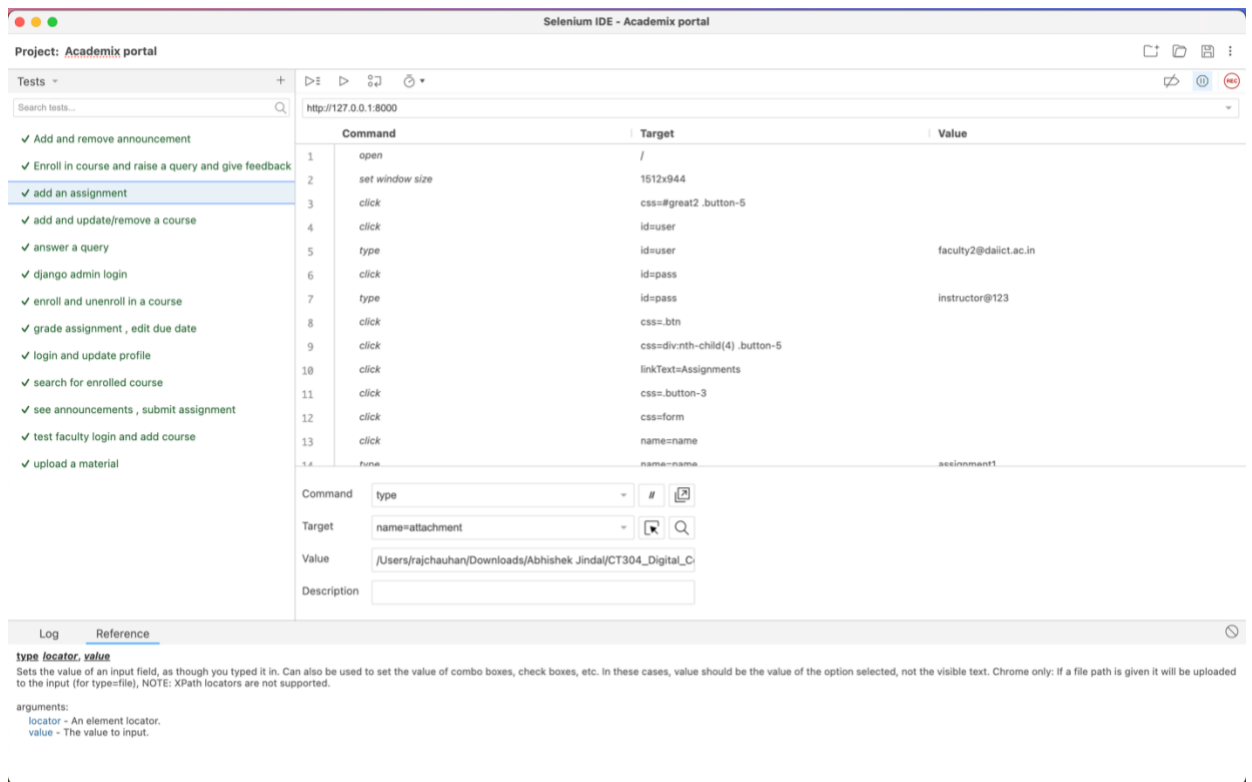
    def test_addandupdateremoveacourse(self):
        # Test name: add and update/remove a course
        # Step # | name | target | value
        # 1 | open | / |
        self.driver.get("http://127.0.0.1:8000/")
        # 2 | setWindowSize | 1512x944 |
        self.driver.set_window_size(1512, 944)
```

```

# 3 | click | css=#great2 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great2 .button-
5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | faculty1@daiict.ac.in
self.driver.find_element(By.ID,
"user").send_keys("faculty1@daiict.ac.in")
# 6 | click | id=pass |
self.driver.find_element(By.ID, "pass").click()
# 7 | type | id=pass | instructor@123
self.driver.find_element(By.ID, "pass").send_keys("instructor@123")
# 8 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 9 | click | css=.nav-item:nth-child(1) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(1) >
.nav-link").click()
# 10 | click | id=coursename |
self.driver.find_element(By.ID, "coursename").click()
# 11 | type | id=coursename | Introduction to networkin
self.driver.find_element(By.ID, "coursename").send_keys("Introduction
to networkin")
# 12 | click | id=courseid |
self.driver.find_element(By.ID, "courseid").click()
# 13 | type | id=courseid | IT305
self.driver.find_element(By.ID, "courseid").send_keys("IT305")
# 14 | click | id=description |
self.driver.find_element(By.ID, "description").click()
# 15 | type | id=description | welcome to my course
self.driver.find_element(By.ID, "description").send_keys("welcome to
my course")
# 16 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 17 | click | css=div:nth-child(5) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(5) .button-
5").click()
# 18 | click | id=description |
self.driver.find_element(By.ID, "description").click()
# 19 | type | id=description | hello all , this is my course
self.driver.find_element(By.ID, "description").send_keys("hello all ,
this is my course")
# 20 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 21 | click | css=.nav-item:nth-child(2) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(2) >
.nav-link").click()
# 22 | click | css=path |
self.driver.find_element(By.CSS_SELECTOR, "path").click()
# 23 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) >
.nav-link").click()

```

**(10). Add an assignment :**



```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestAddanassignment():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_addanassignment(self):
        # Test name: add an assignment
        # Step # | name | target | value
        # 1 | open | / |
        self.driver.get("http://127.0.0.1:8000/")
        # 2 | setWindowSize | 1512x944 |
        self.driver.set_window_size(1512, 944)
```



```

# 3 | click | css=#great2 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great2 .button-
5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | faculty2@daiict.ac.in
self.driver.find_element(By.ID,
"user").send_keys("faculty2@daiict.ac.in")
# 6 | click | id=pass |
self.driver.find_element(By.ID, "pass").click()
# 7 | type | id=pass | instructor@123
self.driver.find_element(By.ID, "pass").send_keys("instructor@123")
# 8 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 9 | click | css=div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(4) .button-
5").click()
# 10 | click | linkText=Assignments |
self.driver.find_element(By.LINK_TEXT, "Assignments").click()
# 11 | click | css=.button-3 |
self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
# 12 | click | css=form |
self.driver.find_element(By.CSS_SELECTOR, "form").click()
# 13 | click | name=name |
self.driver.find_element(By.NAME, "name").click()
# 14 | type | name=name | assignment1
self.driver.find_element(By.NAME, "name").send_keys("assignment1")
# 15 | click | name=description |
self.driver.find_element(By.NAME, "description").click()
# 16 | type | name=description | here is your lab1 work!
self.driver.find_element(By.NAME, "description").send_keys("here is
your lab1 work!")
# 17 | click | id=duedate |
self.driver.find_element(By.ID, "duedate").click()
# 18 | type | id=duedate | 2023-12-04T07:34
self.driver.find_element(By.ID, "duedate").send_keys("2023-12-
04T07:34")
# 19 | click | id=duedate |
self.driver.find_element(By.ID, "duedate").click()
# 20 | click | id=duedate |
self.driver.find_element(By.ID, "duedate").click()
# 21 | click | css=label:nth-child(10) |
self.driver.find_element(By.CSS_SELECTOR, "label:nth-
child(10)").click()
# 22 | type | name=max_grade | 1
self.driver.find_element(By.NAME, "max_grade").send_keys("1")
# 23 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 24 | type | name=max_grade | 2
self.driver.find_element(By.NAME, "max_grade").send_keys("2")
# 25 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 26 | doubleClick | name=max_grade |
element = self.driver.find_element(By.NAME, "max_grade")

```

```

actions = ActionChains(self.driver)
actions.double_click(element).perform()
# 27 | type | name=max_grade | 3
self.driver.find_element(By.NAME, "max_grade").send_keys("3")
# 28 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 29 | type | name=max_grade | 4
self.driver.find_element(By.NAME, "max_grade").send_keys("4")
# 30 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 31 | type | name=max_grade | 5
self.driver.find_element(By.NAME, "max_grade").send_keys("5")
# 32 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 33 | type | name=max_grade | 6
self.driver.find_element(By.NAME, "max_grade").send_keys("6")
# 34 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 35 | type | name=max_grade | 7
self.driver.find_element(By.NAME, "max_grade").send_keys("7")
# 36 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 37 | type | name=max_grade | 8
self.driver.find_element(By.NAME, "max_grade").send_keys("8")
# 38 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 39 | doubleClick | name=max_grade |
element = self.driver.find_element(By.NAME, "max_grade")
actions = ActionChains(self.driver)
actions.double_click(element).perform()
# 40 | type | name=max_grade | 9
self.driver.find_element(By.NAME, "max_grade").send_keys("9")
# 41 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 42 | type | name=max_grade | 10
self.driver.find_element(By.NAME, "max_grade").send_keys("10")
# 43 | click | name=max_grade |
self.driver.find_element(By.NAME, "max_grade").click()
# 44 | click | name=attachment |
self.driver.find_element(By.NAME, "attachment").click()
# 45 | type | name=attachment | /Users/rajchauhan/Downloads/Abhishek
Jindal/CT304_Digital_Communication_23_24/Course_Flyer_CT304_23_24.pdf
self.driver.find_element(By.NAME,
"attachment").send_keys("/Users/rajchauhan/Downloads/Abhishek
Jindal/CT304_Digital_Communication_23_24/Course_Flyer_CT304_23_24.pdf")
# 46 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 47 | click | linkText=IT313 |
self.driver.find_element(By.LINK_TEXT, "IT313").click()
# 48 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) >
.nav-link").click()

```

**(11). Enroll course , raise a query and give course feedback**

Selenium IDE - Academix portal

Project: Academix portal

Tests +

Search tests...

http://127.0.0.1:8000

Command	Target	Value
1 open	/	
2 set window size	1512x944	
3 click	css=#great1 .button-5	
4 click	id=log	
5 click	id=user	
6 type	id=user	202101030@dailict.ac.in
7 click	id=pass3	
8 type	id=pass3	chaitanya@123
9 send keys	id=pass3	\$(KEY_ENTER)
10 click	css=.solutions-box:nth-child(1) > div:nth-child(4) .button-5	
11 click	linkText=Query	
12 click	css=.button-3	
13 click	name=qry	

Command open #

Target /

Value

Description

Log Reference

**open url**  
Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs.  
arguments:  
url - The URL to open (may be relative or absolute).

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestEnrollincourseandraiseaqueryandgivefeedback():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_enrollincourseandraiseaqueryandgivefeedback(self):
        # Test name: Enroll in course and raise a query and give feedback
        # Step # | name | target | value
        # 1 | open | / |
        self.driver.get("http://127.0.0.1:8000/")
        # 2 | setWindowSize | 1512x944 |
        self.driver.set_window_size(1512, 944)
        # 3 | click | css=#great1 .button-5 |
```

```

        self.driver.find_element(By.CSS_SELECTOR, "#great1 .button-
5").click()
        # 4 | click | id=log |
        self.driver.find_element(By.ID, "log").click()
        # 5 | click | id=user |
        self.driver.find_element(By.ID, "user").click()
        # 6 | type | id=user | 202101030@daiict.ac.in
        self.driver.find_element(By.ID,
"user").send_keys("202101030@daiict.ac.in")
        # 7 | click | id=pass3 |
        self.driver.find_element(By.ID, "pass3").click()
        # 8 | type | id=pass3 | chaitanya@123
        self.driver.find_element(By.ID, "pass3").send_keys("chaitanya@123")
        # 9 | sendKeys | id=pass3 | ${KEY_ENTER}
        self.driver.find_element(By.ID, "pass3").send_keys(Keys.ENTER)
        # 10 | click | css=.solutions-box:nth-child(1) > div:nth-child(4)
.button-5 |
        self.driver.find_element(By.CSS_SELECTOR, ".solutions-box:nth-
child(1) > div:nth-child(4) .button-5").click()
        # 11 | click | linkText=Query |
        self.driver.find_element(By.LINK_TEXT, "Query").click()
        # 12 | click | css=.button-3 |
        self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
        # 13 | click | name=qry |
        self.driver.find_element(By.NAME, "qry").click()
        # 14 | type | name=qry | hello sir , i had an query regarding the
SDLC , i need your help in that.
        self.driver.find_element(By.NAME, "qry").send_keys("hello sir , i had
an query regarding the SDLC , i need your help in that.")
        # 15 | click | css=.btn |
        self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
        # 16 | click | linkText=Feedback |
        self.driver.find_element(By.LINK_TEXT, "Feedback").click()
        # 17 | click | name=feedback |
        self.driver.find_element(By.NAME, "feedback").click()
        # 18 | type | name=feedback | excellent course , enjoyed it !
        self.driver.find_element(By.NAME, "feedback").send_keys("excellent
course , enjoyed it !")
        # 19 | click | css=.btn |
        self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
        # 20 | click | linkText=IT313 |
        self.driver.find_element(By.LINK_TEXT, "IT313").click()
        # 21 | click | css=.nav-item:nth-child(4) > .nav-link |
        self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(4) >
.nav-link").click()

```

## (12). Add and remove announcement :

The screenshot shows the Selenium IDE interface for a project named 'Academix portal'. The test suite 'Add and remove announcement' is selected, showing 14 steps. The first step is 'open' with target '/' and value 'http://127.0.0.1:8000'. The second step is 'set window size' with target '1512x944'. The third step is 'click' with target 'css=#great2.button-5'. The fourth step is 'click' with target 'id=user' and value 'faculty2@dalict.ac.in'. The fifth step is 'type' with target 'id=pass' and value 'instructor@123'. The sixth step is 'click' with target 'css=.btn'. The seventh step is 'click' with target 'css=div:nth-child(4).button-5'. The eighth step is 'click' with target 'css=.button-3'. The ninth step is 'click' with target 'name=title' and value 'Extra Lecture I'. The tenth step is 'type' with target 'name=title' and value 'Extra Lecture I'. The eleventh step is 'click' with target 'name=desc' and value 'we have an extra lecture scheduled for tomorrow. section A'. The twelfth step is 'type' with target 'name=desc' and value 'we have an extra lecture scheduled for tomorrow. section A'. The thirteenth step is 'type' with target 'name=desc' and value 'we have an extra lecture scheduled for tomorrow. section A'. The fourteenth step is 'type' with target 'name=desc' and value 'we have an extra lecture scheduled for tomorrow. section A'.

Step	Command	Target	Value
1	open	/	http://127.0.0.1:8000
2	set window size	1512x944	
3	click	css=#great2.button-5	
4	click	id=user	faculty2@dalict.ac.in
5	type	id=pass	instructor@123
6	click	css=.btn	
7	click	css=div:nth-child(4).button-5	
8	click	css=.button-3	
9	click	name=title	Extra Lecture I
10	type	name=title	Extra Lecture I
11	click	name=desc	we have an extra lecture scheduled for tomorrow. section A
12	type	name=desc	we have an extra lecture scheduled for tomorrow. section A
13	type	name=desc	we have an extra lecture scheduled for tomorrow. section A
14	type	name=desc	we have an extra lecture scheduled for tomorrow. section A

Log Reference

**open url**  
Opens a URL and waits for the page to load before proceeding. This accepts both relative and absolute URLs.  
arguments:  
url - The URL to open (may be relative or absolute).

```
# Generated by Selenium IDE
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestAddandremoveannouncement():
    def setup_method(self, method):
        self.driver = webdriver.Chrome()
        self.vars = {}

    def teardown_method(self, method):
        self.driver.quit()

    def test_addandremoveannouncement(self):
        # Test name: Add and remove announcement
        # Step # | name | target | value
        # 1 | open | / |
        self.driver.get("http://127.0.0.1:8000/")
        # 2 | setWindowSize | 1512x944 |
        self.driver.set_window_size(1512, 944)
```

```

# 3 | click | css=#great2 .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "#great2 .button-
5").click()
# 4 | click | id=user |
self.driver.find_element(By.ID, "user").click()
# 5 | type | id=user | faculty2@daiict.ac.in
self.driver.find_element(By.ID,
"user").send_keys("faculty2@daiict.ac.in")
# 6 | click | id=pass |
self.driver.find_element(By.ID, "pass").click()
# 7 | type | id=pass | instructor@123
self.driver.find_element(By.ID, "pass").send_keys("instructor@123")
# 8 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 9 | click | css=div:nth-child(4) .button-5 |
self.driver.find_element(By.CSS_SELECTOR, "div:nth-child(4) .button-
5").click()
# 10 | click | css=.button-3 |
self.driver.find_element(By.CSS_SELECTOR, ".button-3").click()
# 11 | click | name=title |
self.driver.find_element(By.NAME, "title").click()
# 12 | type | name=title | Extra Lecture !
self.driver.find_element(By.NAME, "title").send_keys("Extra Lecture
!")
# 13 | click | name=desc |
self.driver.find_element(By.NAME, "desc").click()
# 14 | type | name=desc | we have an extra lecture scheduled for
tomorrow , section A needs to come on time.
self.driver.find_element(By.NAME, "desc").send_keys("we have an extra
lecture scheduled for tomorrow , section A needs to come on time.")
# 15 | click | css=.btn |
self.driver.find_element(By.CSS_SELECTOR, ".btn").click()
# 16 | click | css=a:nth-child(3) > .button-3 |
self.driver.find_element(By.CSS_SELECTOR, "a:nth-child(3) > .button-
3").click()
# 17 | click | linkText=IT313 |
self.driver.find_element(By.LINK_TEXT, "IT313").click()
# 18 | click | css=.nav-item:nth-child(3) > .nav-link |
self.driver.find_element(By.CSS_SELECTOR, ".nav-item:nth-child(3) >
.nav-link").click()

```