



Web Automation **Pytest Playwright** with **Allure Report**



# Jaman sebelum Pytest Playwright Allure

1. Test dilakukan secara manual termasuk test regresi
2. Menjalankan test pada satu browser pada suatu waktu
3. Laporan dibuat secara manual





## PYTEST

Framework pengujian di Python untuk menulis dan menjalankan tes otomatis dengan sintaks sederhana dan fleksibel.

Mampu menangani pengujian dari unit testing sederhana hingga pengujian skala besar yang lebih kompleks.



## PLAYWRIGHT

Framework browser automation berbagai bahasa pemrograman, termasuk Python, dan memungkinkan pengujian di berbagai browser (Chromium, Firefox, WebKit).

Kecepatan, dukungan multi-browser, testing parallel



## ALLURE

Alat reporting yang menampilkan hasil pengujian dalam format yang mudah dipahami dengan tampilan visual

Komprehensif dengan diagram, logs

# Report

Test Case ID	Features	User Story	Test Case	Pre-condition and Setup	Test Data	Test Steps	Expected Result	Actual Result	Test Result	Screenshot
S-001	QSD-1	QSD-6 As a buyer, I want to be able to login to the website	Open the login page	Test Environment: - Test Device: Huawei Mate Book D14 - OS: Win11 Home - Browser: Chrome and Mozilla	<a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. Open the following URL: - <a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. The login page is loaded successfully 2. Username and Password field is visible and editable 3. Login button is Clickable	1. User cannot see the login page	Pass	<a href="#">Login page</a>
S-002	QSD-1	QSD-6 As a buyer, I want to be able to login to the website	Login as Standard User	Test Environment: - Test Device: Huawei Mate Book D14 - OS: Win11 Home - Browser: Chrome and Mozilla	<a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. Open website 2. Enter "standard_user" into "Username" 3. Enter "secret_sauce" into "Password" 4. Click on "login-button"	1. The login page is loaded successfully 2. Username and Password field is visible and editable 3. Login button is Clickable 4. User login successfully without issue	1. User logged in successfully	Pass	<a href="#">Login as standard user</a>
S-003	QSD-1	QSD-6 As a buyer, I want to be able to login to the website	Login as Locked Out User	Test Environment: - Test Device: Huawei Mate Book D14 - OS: Win11 Home - Browser: Chrome and Mozilla	<a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. Open website 2. Enter "locked_out_user" into "Username" 3. Enter "secret_sauce" into "Password" 4. Click on "login-button"	1. The login page is loaded successfully 2. Username and Password field is visible and editable 3. Login button is Clickable 4. User login successfully without issue	1. User cannot login	Fail	<a href="#">Login as locked out user</a>
S-004	QSD-1	QSD-6 As a buyer, I want to be able to login to the website	Login as Problem User	Test Environment: - Test Device: Huawei Mate Book D14 - OS: Win11 Home - Browser: Chrome and Mozilla	<a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. Open website 2. Enter "problem_user" into "Username" 3. Enter "secret_sauce" into "Password" 4. Click on "login-button"	1. The login page is loaded successfully 2. Username and Password field is visible and editable 3. Login button is Clickable 4. User login successfully without issue	1. User logged in successfully with issue afterward	Pass	<a href="#">Login as problem user</a>
S-005	QSD-1	QSD-6 As a buyer, I want to be able to login to the website	Login as Performance Glitch User	Test Environment: - Test Device: Huawei Mate Book D14 - OS: Win11 Home - Browser: Chrome and Mozilla	<a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. Open website 2. Enter "performance_glitch_user" into "Username" 3. Enter "secret_sauce" into "Password" 4. Click on "login-button"	1. The login page is loaded successfully 2. Username and Password field is visible and editable 3. Login button is Clickable 4. User login successfully without issue	1. User logged in successfully but took more than 4 seconds or a glitch	Pass	<a href="#">Login as performance glitch user</a>
S-006	QSD-1	QSD-6 As a buyer, I want to be able to login to the website	Login as Error User	Test Environment: - Test Device: Huawei Mate Book D14 - OS: Win11 Home - Browser: Chrome and Mozilla	<a href="https://www.xayce.com/">https://www.xayce.com/</a>	1. Open website 2. Enter "error_user" into "Username" 3. Enter "secret_sauce" into "Password" 5. Click on "login-button"	1. The login page is loaded successfully 2. Username and Password field is visible and editable 3. Login button is Clickable 4. User login successfully without issue	1. User logged in successfully	Pass	<a href="#">Login as error user</a>

# Report

t

## laporan.html

Report generated on 13-Oct-2024 at 11:02:02 by [pytest-html](#) v4.1.1

### Environment

Python	3.12.3
Platform	Windows-10-10.0.19045-SP0
Packages	<ul style="list-style-type: none"><li>• pytest: 8.3.3</li><li>• pluggy: 1.5.0</li></ul>
Plugins	<ul style="list-style-type: none"><li>• allure-pytest: 2.13.5</li><li>• base-url: 2.1.0</li><li>• html: 4.1.1</li><li>• metadata: 3.1.1</li><li>• playwright: 0.5.2</li></ul>
JAVA_HOME	C:\Program Files\Java\jdk-22
Base URL	

### Summary

6 tests took 00:02:25.

(Un)check the boxes to filter the results.

2 Failed, 3 Passed, 0 Skipped, 0 Expected failures, 0 Unexpected passes, 0 Errors, 0 Runurs

[Show all details](#) / [Hide all details](#)

Result	Test	Duration	Links
Passed	test_performa.py::test_login_sd(chromium-visual_user-secret_sauce)	00:00:19	
Passed	test_performa.py::test_login_sd(chromium-performance_gitch_user-secret_sauce)	00:00:34	
Passed	test_performa.py::test_login_sd(chromium-standard_user-secret_sauce)	00:00:41	
Failed	test_performa.py::test_login_sd(chromium-error_user-secret_sauce)	00:00:20	

```
page = <Page url='https://www.saucedemo.com/checkout-step-two.html', Username = 'error_user', Password = 'secret_sauce'
```

[expand \[-\]](#)

```
@pytest.mark.parametrize('Username,Password', credentials)
def test_login_sd(page:Page, Username,Password):
    # browser membuka page.
    with allure.step("membuka website"):
        page.goto('https://saucedemo.com')
        sleep(2)

    with allure.step("Login sesuai credential"):
        # # page.get_by_placeholder('Username').fill('standard_user')
        # # page.get_by_placeholder('Password').fill('secret_sauce')
        # page.locator("//input[id='user-name']").fill('standard_user')
        # page.locator("//input[id='password']").fill('secret_sauce')
        # page.locator("//input[id='login-button']").click()
```

Failed test\_performa.py::test\_login\_sd(chromium-problem\_user-secret\_sauce)

00:00:17

```
page = <Page url='https://www.saucedemo.com/checkout-step-one.html', Username = 'problem_user', Password = 'secret_sauce'
```

[expand \[-\]](#)

```
@pytest.mark.parametrize('Username,Password', credentials)
def test_login_sd(page:Page, Username,Password):
    # browser membuka page.
    with allure.step("membuka website"):
        page.goto('https://saucedemo.com')
        sleep(2)

    with allure.step("Login sesuai credential"):
        # # page.get_by_placeholder('Username').fill('standard_user')
        # # page.get_by_placeholder('Password').fill('secret_sauce')
        # page.locator("//input[id='user-name']").fill('standard_user')
        # page.locator("//input[id='password']").fill('secret_sauce')
        # page.locator("//input[id='login-button']").click()
```

Failed test\_performa.py::test\_login\_sd(chromium-locked\_out\_user-secret\_sauce)

00:00:15

```
page = <Page url='https://www.saucedemo.com/', Username = 'locked_out_user', Password = 'secret_sauce'
```

[expand \[-\]](#)

```
@pytest.mark.parametrize('Username,Password', credentials)
def test_login_sd(page:Page, Username,Password):
    # browser membuka page.
    with allure.step("membuka website"):
        page.goto('https://saucedemo.com')
        sleep(2)

    with allure.step("Login sesuai credential"):
        # # page.get_by_placeholder('Username').fill('standard_user')
        # # page.get_by_placeholder('Password').fill('secret_sauce')
        # page.locator("//input[id='user-name']").fill('standard_user')
        # page.locator("//input[id='password']").fill('secret_sauce')
        # page.locator("//input[id='login-button']").click()
        page.get_by_role('textbok', name='Username').fill(Username)
```

# Report



Allure



Overview



Categories



Suites



Graphs



Timeline



Behaviors



Packages

En



Collapse

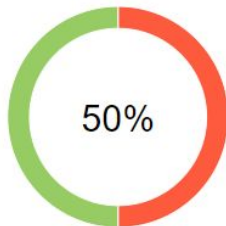
## ALLURE REPORT

10/10/2024

18:58:00 - 19:00:05 (2m 05s)

6

test cases



### SUITES 1 item total

test\_pertama



Show all

### ENVIRONMENT

There are no environment variables

### FEATURES BY STORIES 6 items total

Show all

## TREND

allure

There is nothing to show

### CATEGORIES 1 item total

Product defects



Show all

## EXECUTORS

There is no information about tests executors

# Virtual Environment itu, adalah kebiasaan yang baik :)

```
python -m venv nama_virtual_environment
```

```
nama_virtual_environment/Scripts/activate
```



PYAUTO



# Instalasi Pytest

```
pip install pytest
```

```
pytest --version
```



PYAUTO





# Instalasi Selenium



# Instalasi Playwright

```
pip install pytest-playwright
```

```
playwright --version
```

-Instalasi browser-  
playwright install



PY AUTO ID

# Instalasi Allure

Karena berbasis java, pastikan sistem teman-teman udah terinstal java version 8

<https://github.com/allure-framework/allure2/releases/tag/2.30.0>

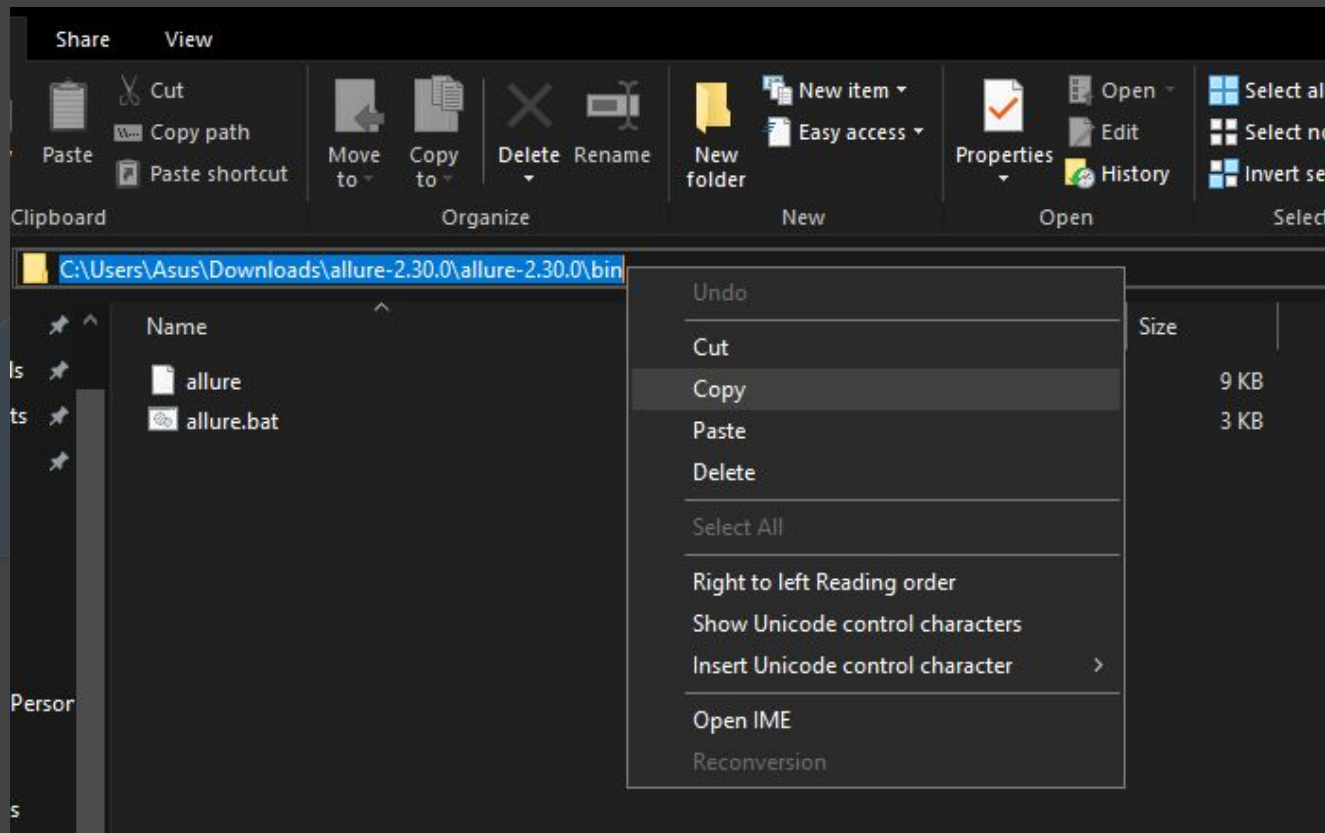
Download file allure2.30.0.tgz atau allure 2.30.0.zip lalu extract

copy patch

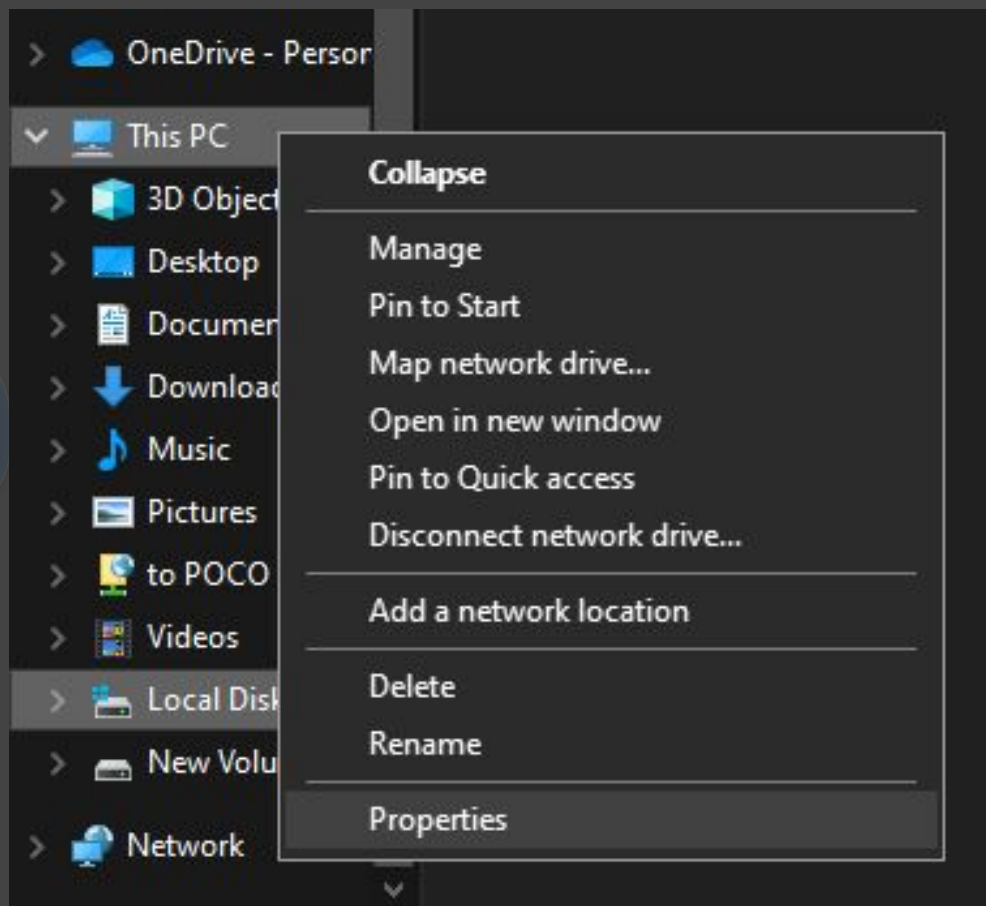
pip install allure-pytest



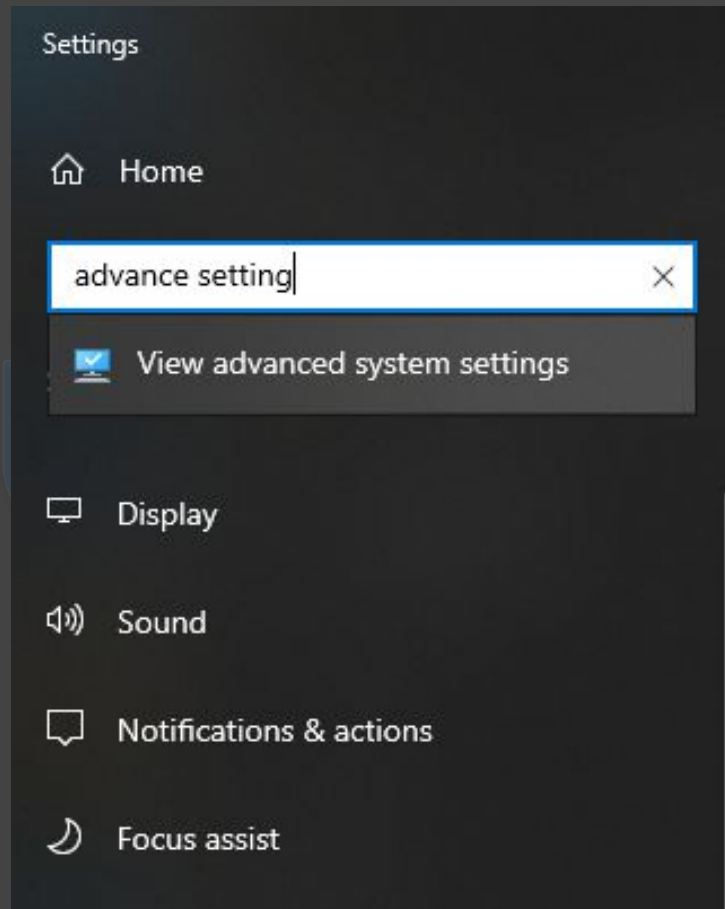
# Copy path



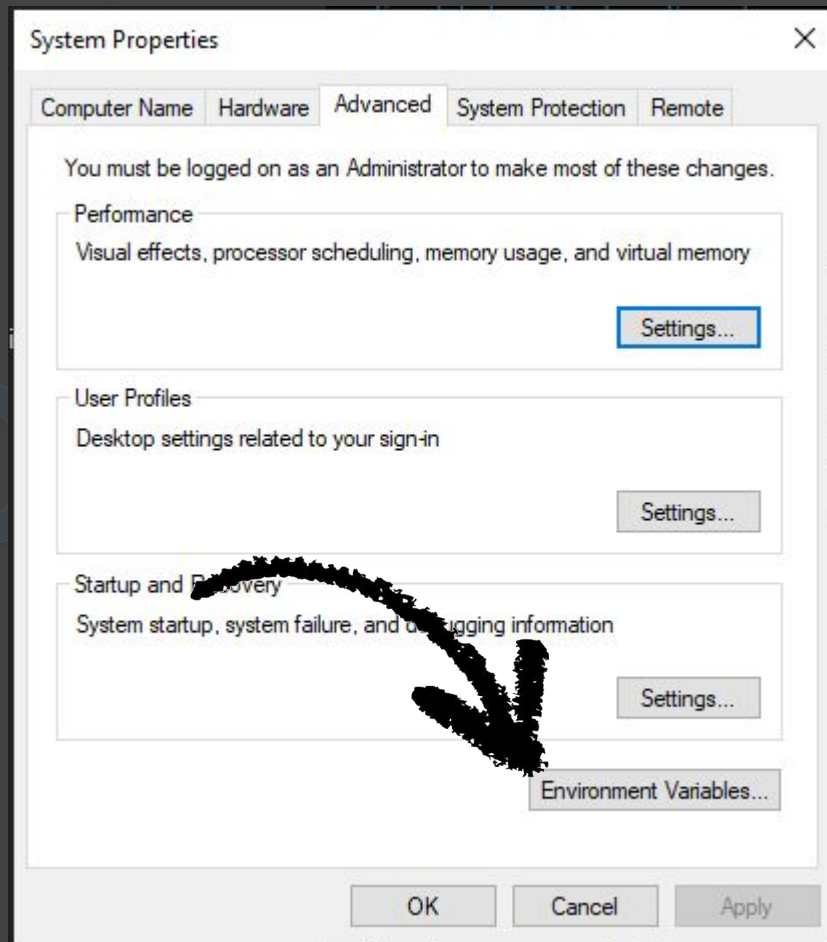
# Copy path



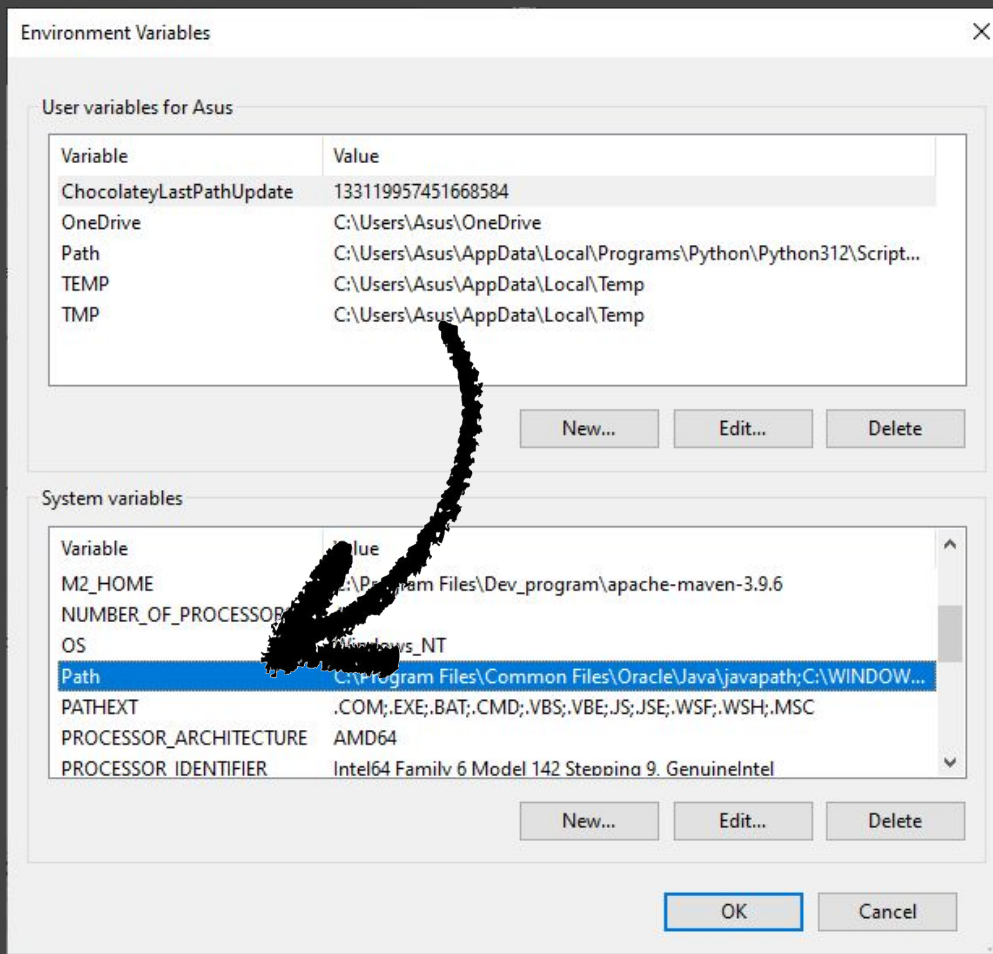
# Copy path



# Copy path

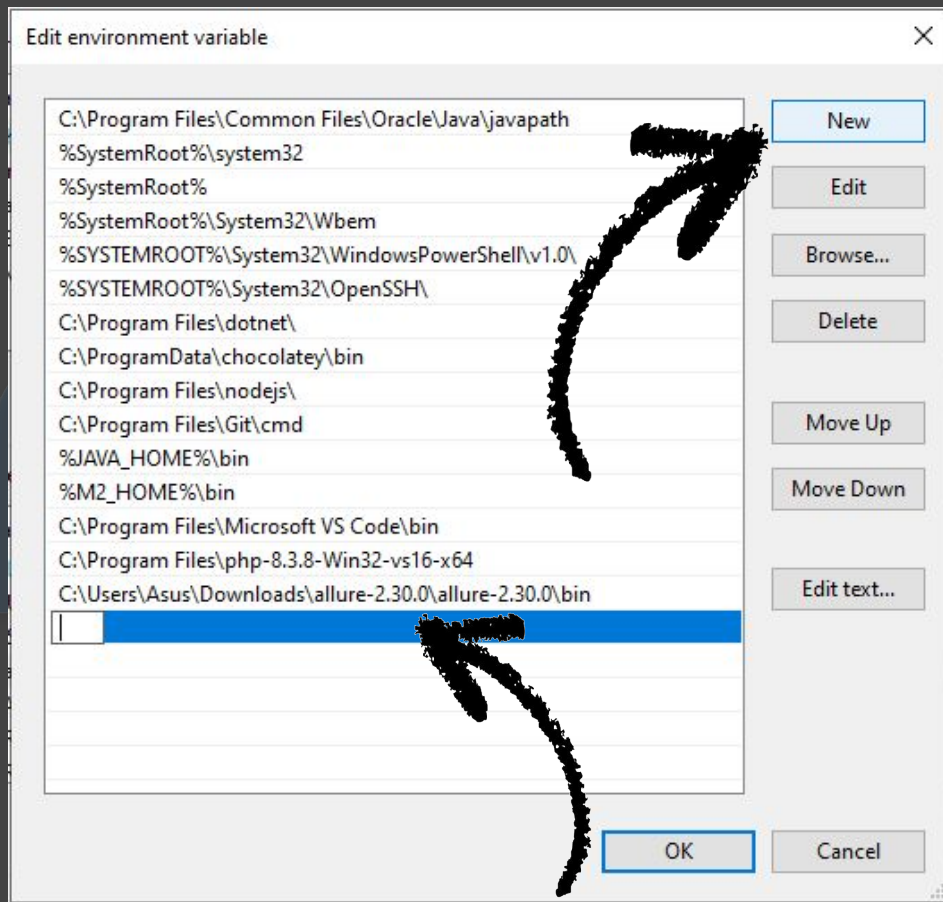


# Copy path





# Copy path



Markinul  
mari -kita-nulis



# Action



Action	Description
<code>locator.check()</code>	Check the input checkbox
<code>locator.click()</code>	Click the element
<code>locator.uncheck()</code>	Uncheck the input checkbox
<code>locator.hover()</code>	Hover mouse over the element
<code>locator.fill()</code>	Fill the form field, input text
<code>locator.focus()</code>	Focus the element
<code>locator.press()</code>	Press single key
<code>locator.set_input_files()</code>	Pick files to upload
<code>locator.select_option()</code>	Select option in the drop down



# Assertion



Assertion	Description
<code>expect(locator).to_be_checked()</code>	Checkbox is checked
<code>expect(locator).to_be_enabled()</code>	Control is enabled
<code>expect(locator).to_be_visible()</code>	Element is visible
<code>expect(locator).to_contain_text()</code>	Element contains text
<code>expect(locator).to_have_attribute()</code>	Element has attribute
<code>expect(locator).to_have_count()</code>	List of elements has given length
<code>expect(locator).to_have_text()</code>	Element matches text
<code>expect(locator).to_have_value()</code>	Input element has value
<code>expect(page).to_have_title()</code>	Page has title
<code>expect(page).to_have_url()</code>	Page has URL



# Kelompok belajar ini di persembahkan oleh



**ARIA UNO SUSENO**  
FOUNDER

**not played for this season.**

PAC	99
SHO	99
PAS	99
DRI	99
DEF	99
PHY	99

in preparation for  
PACONA  
APAC 2024  
APAC 2024

A portrait of Aria Uno Suseno, a man with glasses and a black shirt, pointing upwards. A yellow dashed circle highlights his head.

**YALI YANTO**  
MODERATOR

PAC	99
SHO	99
PAS	99
DRI	99
DEF	99
PHY	99

A portrait of Yali Yanto, a man with a mustache and a blue shirt, smiling. A yellow dashed circle highlights his head.

**ERI PERMADI**  
SPEAKER

PAC	99
SHO	99
PAS	99
DRI	99
DEF	99
PHY	99

A portrait of Eri Permadi, a man wearing a cap and a brown hoodie, looking to the side. A yellow dashed circle highlights his head.