

Pre-requisites:

platformName	text	Android	🗑
appium:platformVersion	text	14	🗑
appium:deviceName	text	appDevice	🗑
appium:automationName	text	UiAutomator2	🗑
appium:appPackage	text	io.appium.android	🗑
appActivity	text	io.appium.android.apis.ApiDemos	🗑

☒ Automatically add necessary Appium vendor prefixes on start

JSON Representation

```
{
  "platformName": "Android",
  "appium:platformVersion": "14",
  "appium:deviceName": "appDevice",
  "appium:automationName": "UiAutomator2",
  "appium:appPackage": "io.appium.android.apis",
  "appActivity": "io.appium.android.apis.ApiDemos"
}
```

Run the appium server by typing the following commands in command prompt:

```
> nvm use v18
```

```
> appium
```

Run the Emulator.

Then Start the Session in the appium Inspector.

Then in the visual studio terminal run the following command:

```
> npx wdio
```

Create locators.js file in the test\specs

Script in locators.js file:

```
describe('Locators in webDriverIO', () => {
  it('First Locator', async () => {
    await $('~Accessibility').click();
  })
})
```

Then run by following command in terminal of visual studio:

```
> npx wdio
```

```
> npx wdio test/specs/locators.js
```

Error Occurs. Then delete the terminal and create the terminal again git bash.

```
> npx wdio
```

Where it shows the test has been passed.

The screenshot shows the VS Code interface with the Explorer sidebar on the left showing a project structure for 'MOBILETEST'. The main editor displays the 'wdio.conf.js' file. The configuration includes capabilities for an Android emulator. The terminal at the bottom shows the test results: 'this is the first test' and '1 passing (919ms)'. It also shows the completion of spec files and logs from the wdio local-runner and cli-launcher.

```
JS wdio.conf.js X JS firstTest.js
EXPLORER
  MOBILETEST
    node_modules
    test\specs
      JS firstTest.js
      {} package-lock.json
      {} package.json
      JS wdio.conf.js

JS wdio.conf.js
1  exports.config = {
52    // https://saucelabs.com/platform/platform-configurator
53    //
54    capabilities: [{
55      // capabilities for local Appium web tests on an Android Emulator
56      platformName: "Android",
57      "appium:platformVersion": "14",
58      "appium:udid": "emulator-5554",
59      "appium:automationName": "UiAutomator2",
60      "appium:appPackage": "io.appium.android.apis",
61      "appium:appActivity": "io.appium.android.apis.ApiDemos"
62    }],
63
64    //
65    // =====
66    // Test Configurations
67    //

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[Android #0-0] ✓ this is the first test
[Android #0-0]
[Android #0-0] 1 passing (919ms)

Spec Files: 1 passed, 1 total (100% completed) in 00:00:18

2025-01-21T06:10:25.666Z INFO @wdio/local-runner: Shutting down spawned worker
2025-01-21T06:10:25.933Z INFO @wdio/local-runner: Waiting for 0 to shut down gracefully
2025-01-21T06:10:25.935Z INFO @wdio/local-runner: shutting down
2025-01-21T06:10:25.936Z INFO @wdio/cli:launcher: Run onComplete hook

TMA239@DESKTOP-QNA5JQC MINGW64 ~/Desktop/mobileTest (master)
```

In the wdio.conf.js file:

This close-up shows the 'specs' array in the wdio.conf.js file. It is currently empty, with a comment indicating where to define the location for spec files. The cursor is positioned at the end of the array definition.

```
24  specs: []
25    // ToDo: define location for spec files here
26    "test/specs/locators.js"
27  ],
```

> npx wdio[Error occurs]

Change [await \$("~ Accessibility").click();] to [await \$("~Access'ibility").click();]

After correcting the accessibility id.

> npx wdio

Script in locators.js file:

```
describe('Locators in webDriverIO', () => {
  it('Accessibility Locator', async () => {
```

```

        await $("~Accessibility").click();
        // Wait for 3 seconds and then closes
        await browser.pause(3000);
    })
    it('ClassName Locator', async () => {
        // Wait for 3 seconds and then closes
        await browser.pause(1000);
        await $("~Custom View").click();
        const ele = await $("android.widget.TextView");
        expect(await
ele.getText()).toContain('Accessibility/Custom View');
        //console.log(text);
        await browser.pause(3000);
    })
    it('XPath Locator', async () => {
        await
$("//android.widget.TextView[@content-desc='Views']");
    })
})
> npx wdio

```

Script in locators.js file:

```

describe('Locators in webDriverIO', () => {
    it('XPath Locator', async () => {
        await
$("//android.widget.TextView[@content-desc='Views']");
        await browser.pause(3000);
    })
})
> npx wdio

```

Create a script named 'calculatorApp.js' in test\specs

Script in 'calculatorApp.js' file:

Download the calculator apk and scroll into emulator and similarly for APKInfo

```
1  exports.config = {
54    capabilities: [{
57      "appium:platformVersion": "14",
58      "appium:udid": "emulator-5554",
59      "appium:automationName": "UiAutomator2",
60      "appium:appPackage": "com.android.phone",
61      "appium:appActivity": "com.android.phone.EmergencyDialer"
62    ],
63  },
64
```

Capability Builder Saved Capability Sets 1 Attach to Session...

Capability Name	Type	Value	Remove
platformName	text	Android	✕
appium:platformVersion	text	14	✕
appium:deviceName	text	appDevice	✕
appium:automationName	text	UiAutomator2	✕
appium:appPackage	text	com.sec.android.app.popupcalculator	✕
appium:appActivity	text	com.sec.android.app.popupcalculator.Calculator	✕

JSON Representation

```
{
  "platformName": "Android",
  "appium:platformVersion": "14",
  "appium:deviceName": "appDevice",
  "appium:automationName": "UiAutomator2",
  "appium:appPackage": "com.sec.android.app.popupcalculator",
  "appium:appActivity": "com.sec.android.app.popupcalculator.Calculator"
}
```

```
// Automating native app
describe('Automating Phone App', () => {
  it('Opening the Phone', async () => {
    await $("~5, JKL").click()
    await $("~8, TUV").click()
    await $("~5, JKL").click()
    await $("~8, TUV").click()
    await $("~5, JKL").click()
    await $("~8, TUV").click()
    await $("~5, JKL").click()
    await $("~8, TUV").click()
    await $("~5, JKL").click()
    await $("~8, TUV").click()
    const icon = await $("~backspace");
  });
});
```

```

    for (let i = 0; i < 2; i++){
      await icon.click()
    }

    await $("~dail").click()
    await browser.pause(1000)
    const text = await
    $("//android.widget.TextView[@resource-id='android:ide/message']").getText()
    console.log(text)
    expect(text).toContain("is not an emergency
number.")
    expect(text).toHaveText(`Can't call. ${number}
is not an emergency number.`)
  })
})

```

In the wdio.conf.js file:



```

24     specs: [
25       // ToDo: define location for sp
26       "test/specs/calculatorApp.js"
27     ],

```

> npx wdio