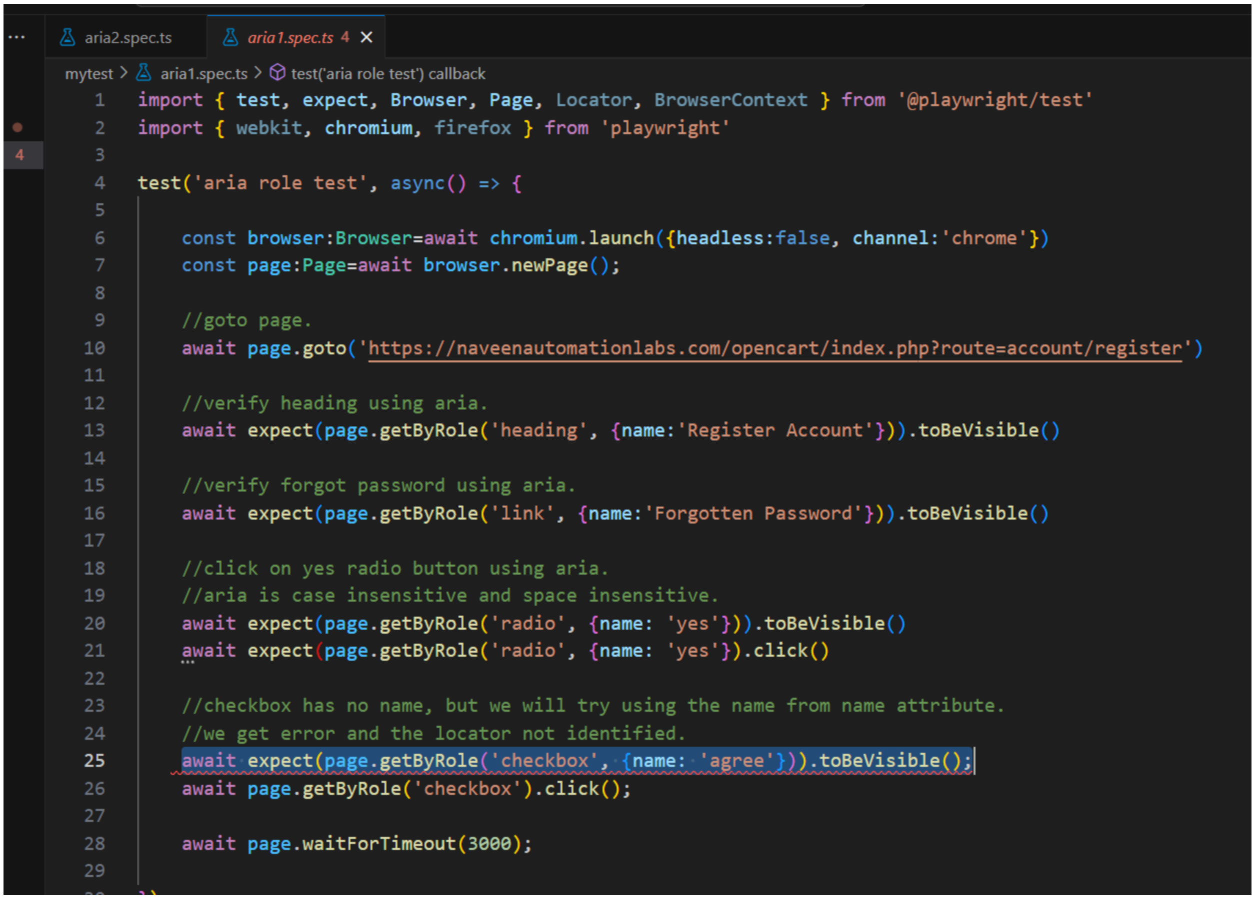
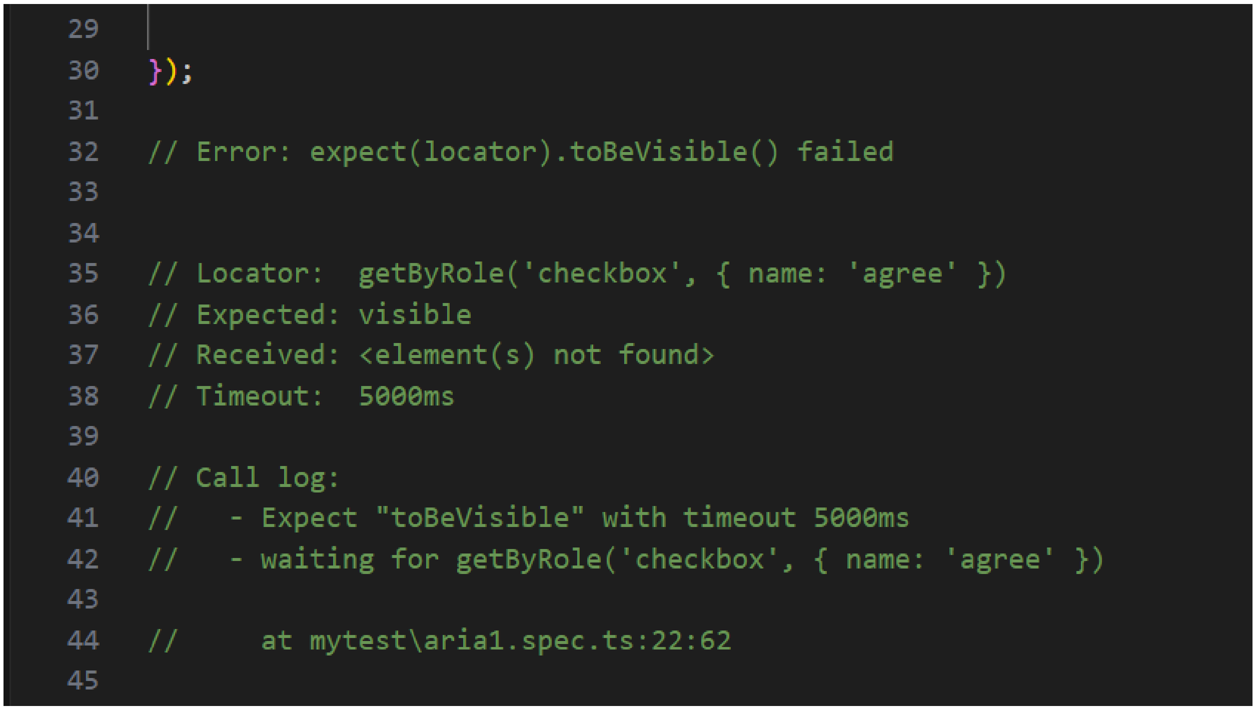
Aria roles can be applied on items which are seen on browser-
First aria for heading is covered below-
Forgotten password aria is covered below-
Aria roles are case insensitive-
Click on yes radio button is covered below.
Lets see for checkbox is covered below-
There is no name for it.
We get time out error when we give wrong name.

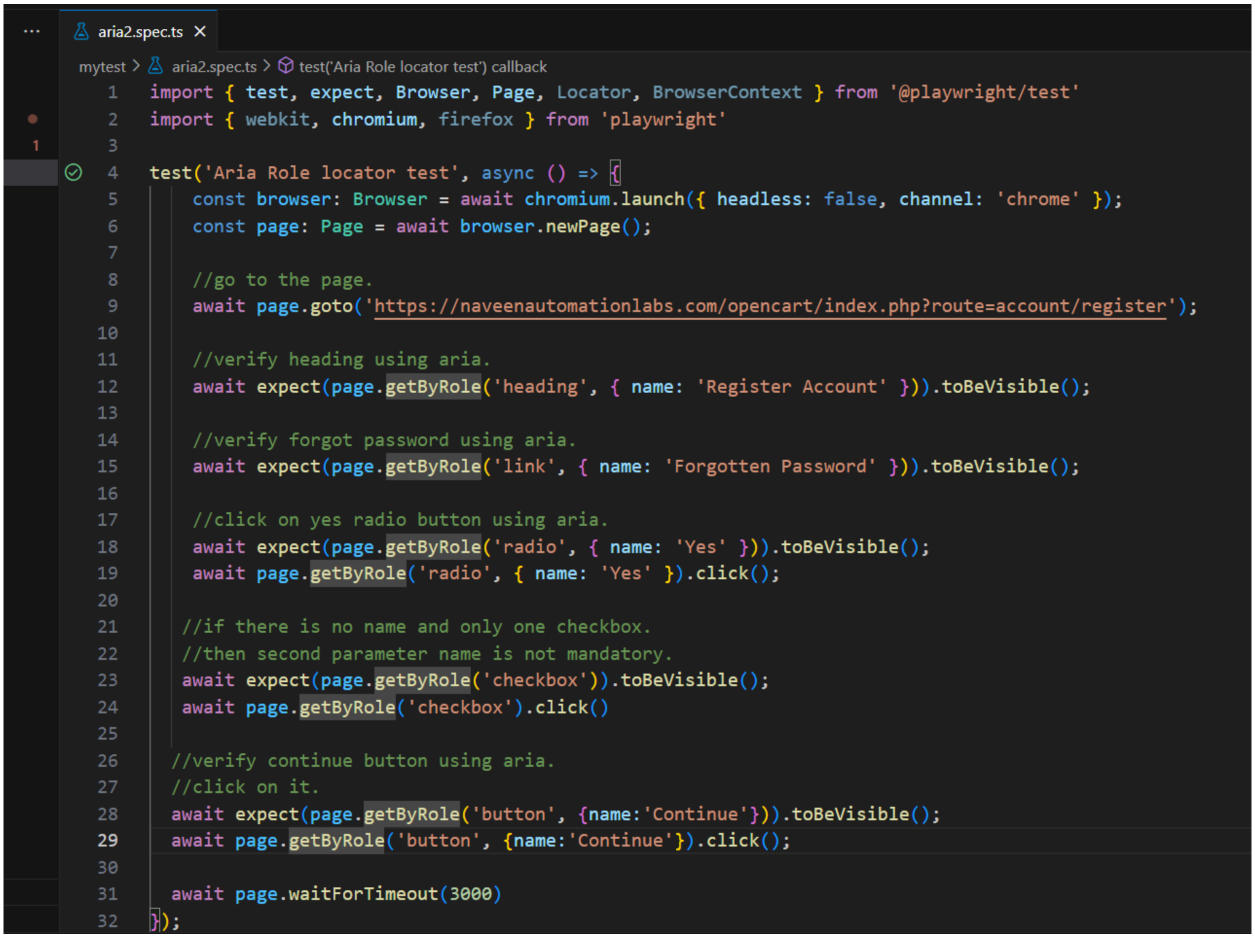


```
import { test, expect, Browser, Page, Locator, BrowserContext } from '@playwright/test'
import { webkit, chromium, firefox } from 'playwright'

test('aria role test', async() => {

    const browser:Browser=await chromium.launch({headless:false, channel:'chrome'})
    const page:Page=await browser.newPage();

    //goto page.
    await page.goto('https://naveenautomationlabs.com/opencart/index.php?route=account/register')

    //verify heading using aria.
    await expect(page.getByRole('heading', {name:'Register Account'})).toBeVisible()

    //verify forgot password using aria.
    await expect(page.getByRole('link', {name:'Forgotten Password'})).toBeVisible()

    //click on yes radio button using aria.
    //aria is case insensitive and space insensitive.
    await expect(page.getByRole('radio', {name: 'yes'})).toBeVisible()
    await expect(page.getByRole('radio', {name: 'yes'}).click()

    //checkbox has no name, but we will try using the name from name attribute.
    //we get error and the locator not identified.
    await expect(page.getByRole('checkbox', {name:  'agree'})).toBeVisible();
    await page.getByRole('checkbox').click();

    await page.waitForTimeout(3000);

})
```

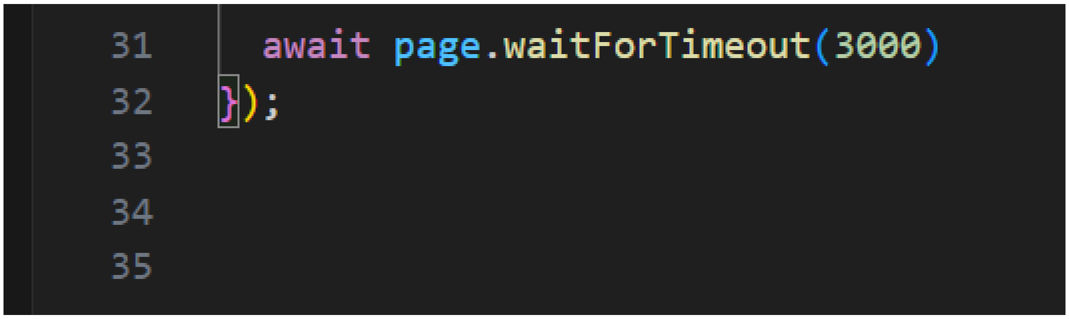


```

});

// Error: expect(locator).toBeVisible() failed


// Locator:  getByRole('checkbox', { name: 'agree' })
// Expected: visible
// Received: <element(s) not found>
// Timeout:  5000ms

// Call log:
//   - Expect "toBeVisible" with timeout 5000ms
//   - waiting for getByRole('checkbox', { name: 'agree' })

//     at mytest\aria1.spec.ts:22:62

```

If there is no name for checkbox, then dont use it-
Second parameter name is not mandatory. But the element has to be unique else aria wont work.
Lets see for continue button is covered below-


aria2.spec.ts
mytest > aria2.spec.ts > test('Aria Role locator test') callback

```
import { test, expect, Browser, Page, Locator, BrowserContext } from '@playwright/test'
import { webkit, chromium, firefox } from 'playwright'

test('Aria Role locator test', async () => {
    const browser: Browser = await chromium.launch({ headless: false, channel: 'chrome' });
    const page: Page = await browser.newPage();

    //go to the page.
    await page.goto('https://naveenautomationlabs.com/opencart/index.php?route=account/register');

    //verify heading using aria.
    await expect(page.getByRole('heading', { name: 'Register Account' })).toBeVisible();

    //verify forgot password using aria.
    await expect(page.getByRole('link', { name: 'Forgotten Password' })).toBeVisible();

    //click on yes radio button using aria.
    await expect(page.getByRole('radio', { name: 'Yes' })).toBeVisible();
    await page.getByRole('radio', { name: 'Yes' }).click();

   //if there is no name and only one checkbox.
   //then second parameter name is not mandatory.
   await expect(page.getByRole('checkbox')).toBeVisible();
   await page.getByRole('checkbox').click()

  //verify continue button using aria.
  //click on it.
  await expect(page.getByRole('button', {name:'Continue'})).toBeVisible();
  await page.getByRole('button', {name:'Continue'}).click();

  await page.waitForTimeout(3000)
});
```



```
  await page.waitForTimeout(3000)
});
```