



Find element by alt attribute – **const logo=await page.getByAltText('company-branding')
await expect(logo).toBeVisible();**

[**//page.getByPlaceholder\(\)](#) - to locate an input by placeholder. await
page.getByPlaceholder('Username').fill("Admin") await
page.getByPlaceholder('Password').fill("admin123")**

get by role-

await page.getByRole('button', {type: 'submit'}).click()

pass in role like button, alert etc.

then pass in attribute:value

get the visible text from the element and then perform the operation-

await expect(await page.getByText(name)).toBeVisible()

to get text from any element-

const name=await page.locator('//p[@class="oxd-userdropdown-name"]').textContent()

get by label-

if the element has label tag.

get by title-

if the element has title attribute.

get by test id-

when we have attribute called data-testid.

```
1 test('Built-inLocators', async({page}) => {
2
3   await page.goto('https://opensource-demo.orangehrmlive.com/web/index.php/auth/login')
4
5   //page.getByAltText() - to locate an element, usually image, by its text alternative.
6   const logo = await page.getByAltText('company-branding')
7   await expect(logo).toBeVisible();
8
9   //page.getByPlaceholder() - to locate an input by placeholder.
10  await page.getByPlaceholder('Username').fill("Admin")
11  await page.getByPlaceholder('Password').fill("admin123")
12
13  await page.getByRole('button', {type: 'submit'}) .click()
14
15  const name = await page.locator('//p[@class="oxd-userdropdown-name"]').textContent()
16
17  await expect(await page.getByText(name)).toBeVisible()
18
19 })
```