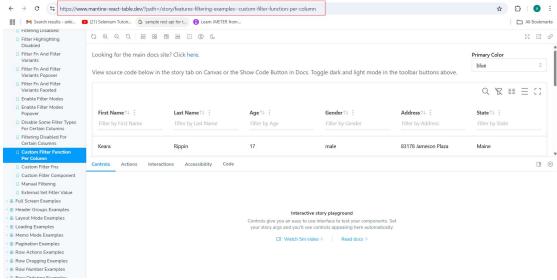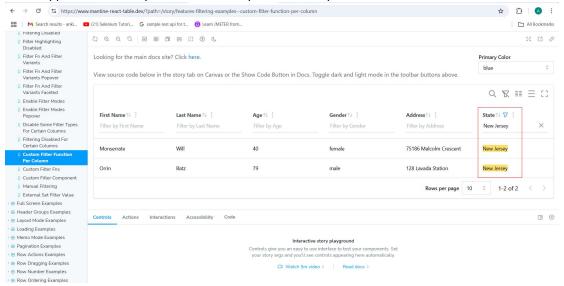# Filtering Test Case Expected result screenshots
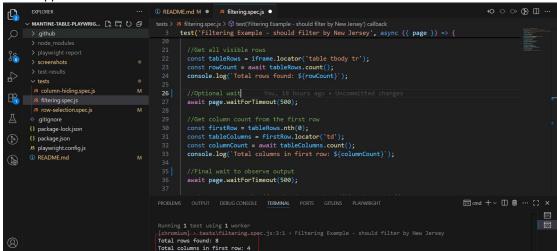
1. Open URL: Application Opens the URL after user confirmation.

2. The Application Filter by state: "New Jersey"automatically.

3.The Application will Get all visible rows


```
tests > JS filtering.spec.js > ✪ test('Filtering Example - should filter by New Jersey') callback
  3    test('Filtering Example - should filter by New Jersey', async ({ page }) => {
 20
 21        //Get all visible rows
 22        const tableRows = iframe.locator('table tbody tr');
 23        const rowCount = await tableRows.count();
 24        console.log(`Total rows found: ${rowCount}`);
 25
 26        //Optional wait        You, 18 hours ago • Uncommitted changes
 27        await page.waitForTimeout(500);
 28
 29        //Get column count from the first row
 30        const firstRow = tableRows.nth(0);
 31        const tableColumns = firstRow.locator('td');
 32        const columnCount = await tableColumns.count();
 33        console.log(`Total columns in first row: ${columnCount}`);
 34
 35        //Final wait to observe output
 36        await page.waitForTimeout(500);
 37
```

```
Running 1 test using 1 worker
[chromium] › tests\filtering.spec.js:3:1 › Filtering Example - should filter by New Jersey
Total rows found: 8
Total columns in first row: 4
```

4.The Application will Get column count from the first row


```
tests > JS filtering.spec.js > ✪ test('Filtering Example - should filter by New Jersey') callback
  3    test('Filtering Example - should filter by New Jersey', async ({ page }) => {
 20
 21        //Get all visible rows
 22        const tableRows = iframe.locator('table tbody tr');
 23        const rowCount = await tableRows.count();
 24        console.log(`Total rows found: ${rowCount}`);
 25
 26        //Optional wait        You, 18 hours ago • Uncommitted changes
 27        await page.waitForTimeout(500);
 28
 29        //Get column count from the first row
 30        const firstRow = tableRows.nth(0);
 31        const tableColumns = firstRow.locator('td');
 32        const columnCount = await tableColumns.count();
 33        console.log(`Total columns in first row: ${columnCount}`);
 34
 35        //Final wait to observe output
 36        await page.waitForTimeout(500);
 37
```

```
Running 1 test using 1 worker
[chromium] › tests\filtering.spec.js:3:1 › Filtering Example - should filter by New Jersey
Total rows found: 8
Total columns in first row: 4
```