



DATE _____

PAGE _____

Dev Notes

Lecture Of - 17th October

// Recap of 16th October

// ~~Login~~ → Go To Hackerrank.com

// Click Login (1st time)

// On 2nd page click login button again under developers option

// On 3rd page enter username &

// password then click LOGIN button

// Go To Compete Tab

// Click ON → Manage Contests

// Then Contest Page Opens

// we click ON contests then add moderator

// Moderator is added by clicking moderator tab then typing moderator's username

// In the end we press enter key & work is done.



* what are we going to do Today? /

3 तक 421 के 20 से ?

⇒ Moderators / Moderator $\frac{2}{21^2}$ contests
+ add होना वाले

⇒ जब page तक होने के बाटुं next

page them $\frac{1}{2}$ page $\frac{1}{2}$ contests main

add होना वाले Moderator till last page

⇒ every contest is opened in a new tab & then tab is closed and this process repeats.

⇒ Wait for 3 seconds on every page to avoid popup.

* Project class 2 till manage contests
everything is same and also
the part of adding moderators
to a single contest is same
and all other changes will be
discussed now in this lecture :-



(*) Repeating Code Till Manage (Antests)

// node1 - HackerrankAutomation.js -- url=
// https://www.hackerrank.com -- config =
config.json.

(*) url \neq at hackerRank.com \neq slash remove
কোডটি কোথায় কীভাবে আসবে।

// npm init -y [say yes to permission]

// npm install minimist [parse কোডটি]

// npm install puppeteer [automation কোডটি]

// Create new folder \neq install \neq install \neq

```
let minimist = require("minimist");
```

```
let puppeteer = require("puppeteer");
```

```
let fs = require("fs");
```

```
let args = minimist(process.argv);
```

```
let configJSON = fs.readFileSync(args.config, "utf-8");
```

```
let configJSO = JSON.parse(configJSON);
```



```
async function run () {
```

```
// Start the browser
```

```
let browser = await puppeteer.launch ({
```

```
headless: false,
```

```
args: [
```

```
--start-maximized
```

```
],
```

```
defaultViewPort: null
```

```
});
```

```
let pages = await browser.pages();
```

```
let page = pages[0];
```

```
await page.goto(args.url);
```

```
await page.waitForSelector("a[data-event-action='login']");
```

```
await page.click(a "[data-event-action='login']);
```

```
await page.waitForSelector("a[href='https://www.hackerrank.com/login']);
```

```
await page.click ("a[href='https://www.hackerrank.com/login']);
```

```
await page.waitForSelector("input[name='username']");
await page.type("input[name='username']", configJSO.username,
    { delay: 50 });
await page.waitForSelector("input[name='password']");
await page.type("input[name='password']", configJSO.password,
    { delay: 50 });
// Can add for wait if get error while log in
await page.waitForSelector("button[data-analytics='LoginPassword']");
await page.click("button[data-analytics='LoginPassword']");
// Click on Complete
await page.waitForSelector("a[data-analytics='NavBarContests']");
await page.click("a[data-analytics='NavBarContests']");
// Click on manage contest
await page.waitForSelector("a[href='/administration/contests']");
await page.click("a[href='/administration/contests']");
(* Continued in end // find no. of pages
// The above code is same for lecture of 16th & 17th Oct
// Now changes begin
// Opening contest moderator in separate tab
```

|| 3101 एक page के सभी क्वेरी सीरीजें

|| we will how so, add moderators for all contexts of a single page.

|| find all urls of same Page. ② 2 classes are present

|| await page.waitForSelector("a.backbone.block-center");

|| let urls = await page.\$\$eval ("a.backbone.block-center", function (tags) {

③ \$\$eval function द्वाया कोड देता है जो जो 314

इस function को दिया जाता है 32 पर 342 query selector

all wala function तो 12 तक and 42 पर page

34 को find करता है and it gives that value

to the function next to it i.e. atags in this case

|| now we get all tags and it is returned to urls as an array.

|| let url = []; (empty array to store urls)

* stuff is just being console logged to check.

DATE _____

PAGE _____

// for (let i = 0; i < atags.length; i++) {

// let url = atags[i].getAttribute("href");

// मैं यह अल्ला को url से atag की value mila

// if this runs for all atags it will give us

// link for all i.e. urls of all contexts.

// urls.push(url); (inserted all urls in urls array)

//]

// return urls;

//);

// यह काम करने से सब anchor tags की array में

// जारी होता है

* console.log(urls);

// Same array की print कर दिया गया

// Ab urls पर loop लगाया गया जब उसकी मद्दत से

// होता है

* Isko ~~2~~E1 comment kia ~~se~~ because
baad ~~se~~ before close tab, we
have to add moderators, make function

```
// for (let i = 0; i < urls.length; i++) {  
    // let curl = urls[i];  
    // let ctab = await browser.newpage();  
    // await ctab.goto(args.url + curl[i]);  
    // await ctab.bringToFront(); // new tab focus  
    // await ctab.waitFor(1000);  
    // await ctab.close();  
    // await page.waitFor(3000);  
}  
  
// This will open all contests in other  
// tabs i.e. 2nd tab open status (1 second) wait  
// Then 2nd tab obi druk kar dena & then again  
// wait on the contest page for 1 second.
```

- * This work can also be done by
Axios and JS DOM libraries but
because this is inbuilt in puppeteer we
use puppeteer's functionality only
- * Here, This work refers to work of
getting url's and reading that data and
using it in curl anyway.

// ~~function~~ 37वां पृष्ठ का , वाले code

// 27 moderation of a single contest add

// करने वाला को महसूस हो दें पर्से करो

// Just before we close out our c-tab.

// Then 32वां new function का लिख दें

// If you get error while giving id &

// password i.e. during login then,

// increase delay & just after entering

// Password (waitFor(3000)) should be added

// for (let i=0; i<curls.length; i++) {

// let cTab = await browser.newPage();

// await saveModeratorInContest({ cTab, args.url +
// curls[i], config
// await page.waitFor(3000); } JSO.moderator)



DATE _____

PAGE _____

a sync function Save Moderator In Contest (ctab, full
curl, moder)

Await ctab.bringToFront(); // tab later {
// focus to new page

Await ctab.goto(fullcurl); // curl

Await ctab.waitFor(3000);

// click on moderators tab

Await ctab.waitForSelector("li[data-tab = 'moderators']);

Await ctab.click("li[data-tab = 'moderators']);

// type in moderator

Await ctab.waitForSelector("input#moderator");

// if you forgot just a reminder # is for ids

Await ctab.type("input#moderator", moderator,
{ delay: 50 });

// press enter

Await ctab.keyboard.press("Enter");

}

// 3-4 ek page cr handle हो गया

run(); // To run / call async.

|| 3-a 212 pages handle ~~obj~~

④ To find the last page we have
to inspect element this arrow
which will take us to the last
page.

|| 3-a gt child elements of ek page ~~obj~~

|| 212 wala se phle ejjj just after

|| manage contexts

continued

⑤ || find number of Pages

await page.waitForSelector("a[data-attr1='last']);

let numPages = await page.\$eval('a[data-attr1="last"]',
function (atag) {

|| \$\$eval & \$eval surf query selector all

|| and query selector ~~obj~~ wala obj

|| \$\$eval gives array like query selector all

* means unnecessary code just pointing on log with console.log



DATE _____
PAGE _____

// \$eval returns a specific value like query selector.

```
let totPages = parseInt(atag.getAttribute('data-page'));  
return totPages;  
});
```

* console.log(numPages); // will print value of last page

* return; // will bring out of code i.e. इसके पास तो कोड नहीं चलेगा

```
for(let i=1, i <= numPages, i++) {
```

await handleAllContentsOfAPage(Page, browser);

if (i != numPages) { // coz for last page right arrow not

await page.waitForSelector("a[data-attr='Right']");

```
await page.click("a[data-attr='Right']");
```

] // Perform work of clicking right arrow to go to next page

```
]
```

async function handleAllContents of A Page
(page, browser) {

// find all works of same page

await page.waitForSelector(`a.backbone-block-center`, function
atags
that)

let works = await page.\$\$eval(`a.backbone-block-center`, function
(atags) {

let works = [];

for (let i = 0; i < atags.length; i++) {

let url = atags[i].getAttribute("href");

works.push(url);

}

return works;

});

for (let i = 0; i < works.length; i++) {

let ctab = await browser.newPage();

await saveModeratorInContent({



```
(ctab, args.url + urls[i], configJSO.moderator);
```

```
await ctab.close();
```

```
await page.waitFor(3000);
```

3

3

Basic Run Of CODE

- * Go To ~~HackerRank~~ HackerRank.com
- * Click login button
- * Click login button under developers
- * Enter user id & password
- * Click login.
- * Click on ~~Manage~~ Complete
- * Click on manage contests.
- * Contest page opens.
- * Use loop to get last page & use that to call a function "handleAllContests" which contains functionality to switch pages after all contests of single page are handled
- * It also has a loop which gets all urls of all contests of a single page
- * It also has a second loop which calls

another function "Save Moderator In Contest"

④ Save moderator In Contest function
Opens new tab for each contest,
add a moderator or list of moderator
then, it closes the tab.

* This brings us to the end of
our activity just don't forget
calling async function in the
end of program.