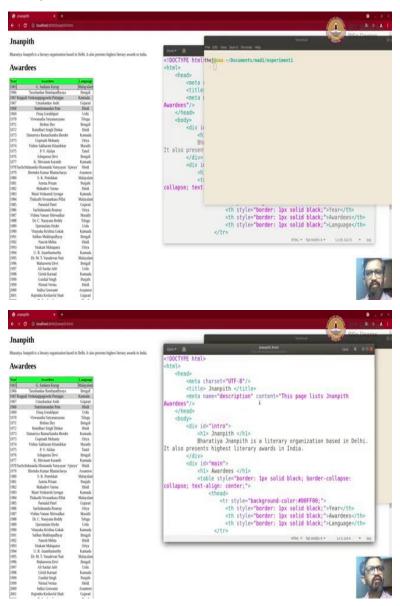


IIT Madras ONLINE DEGREE

Modern Application Development - 1 Professor Thejesh G N Software Consultant IITM BSc Degree Indian Instituted of Technology, Madras Introduction to Internal and External CSS

(Refer Slide Time: 0:16)

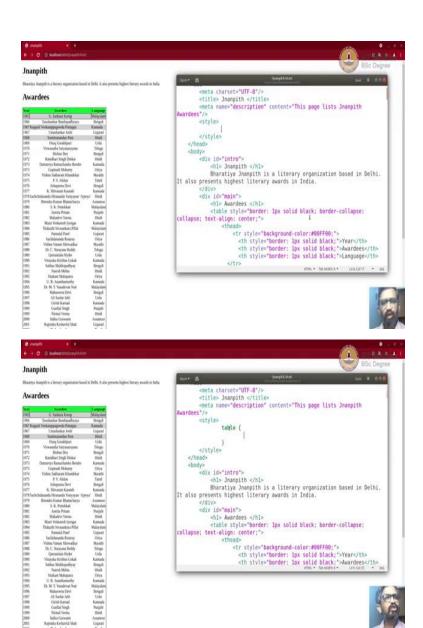


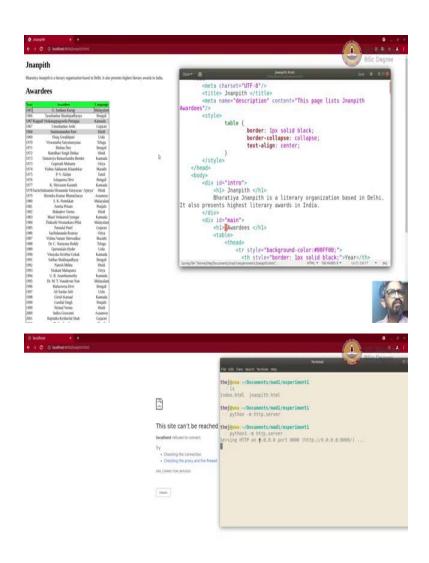


Hi, welcome back to the Modern Application Development screencast. In this short screencast, we will learn how to separate the style and content in our HTML document. Before we start, as usual, make sure you have a browser, you have an editor, and you have a terminal open. And we are going to continue with the file that we left last time, I am going to do a recap small one, just to remind you, we had styled certain items in this HTML with inline styling, like we learned the way we did it is very inefficient, and prone to errors. And it makes us both content and style together, we want to do a better way of doing it.

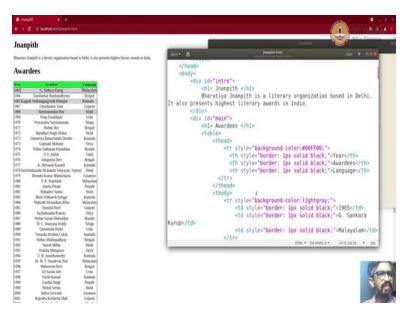
(Refer Slide Time: 1:06)

```
chead charset="UTF-8"/>
chead
```









And we will do that, by separating out the style and content in the, there two ways to do it in the first way, we will add a style tag at the top of the document, let us add that. We added the style tag within a <head>. So, here we are going to define some styles, that is going to be used across.

Now actually, we are going to modify the same document. And let us out like remove the style, inline style and move the style to the style inside the style tags here in the head. And keep this design same, there are many ways to do it. Like one, the one which we are going to use is by the tag name, we are going to style the tag.

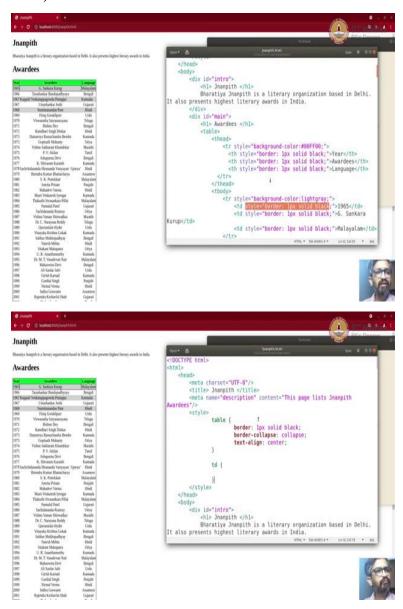
Since we have only one table, we can just know style that table directly. That is done by actual defining the style first, using a tag selector. Now, we want to apply style to a table. So, we will do table and we will open the flower bracket and close the flower bracket, within this will define whatever the style that needs to be applied to all the table documents, all the table elements within this document.

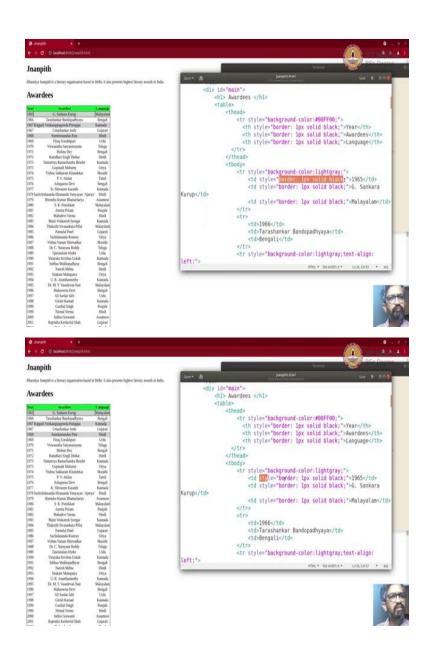
Now, since I already have the style here, I am going to copy this style and paste it here. Just going to make it clean, look clean, by separating each style parameter per line and save it. Now, you can see I have removed the style here and added the table level style here. So, I am going to save this. I am going to reload.

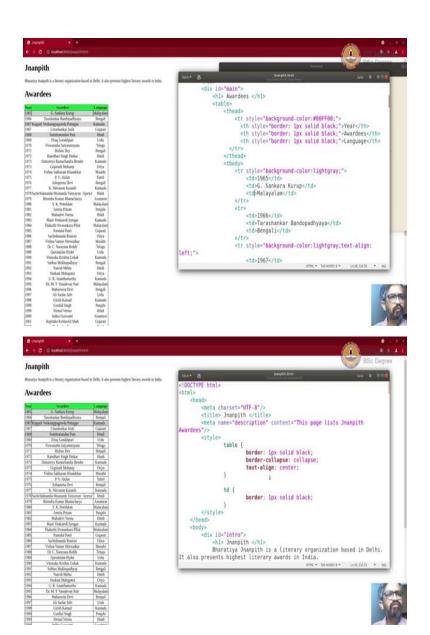
Now, you see how the table looks like it is center aligned. collapsible with border one black pixel here is the border. Yeah. Oh, what happened? Actually, we did not start the server, so let us start a server. That is done by similarly python -m http.server. Python 3. Now, our server is running,

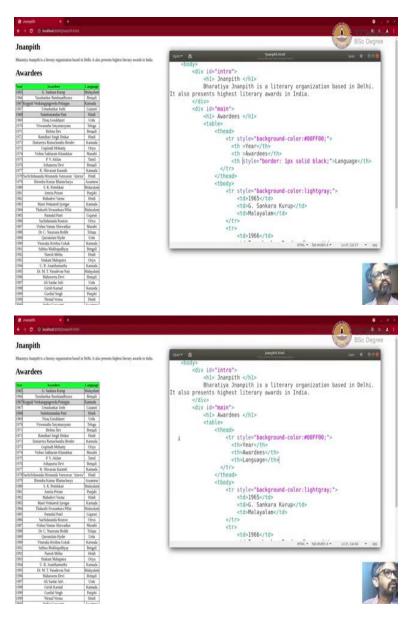
now we should be able to refresh. So, as you can see, it remains the same. And now we are actually removed the style from the tag and let us move it to the top. Now, it is table there was only one table, it did not make much difference in terms of effort.

(Refer Slide Time: 4:04)





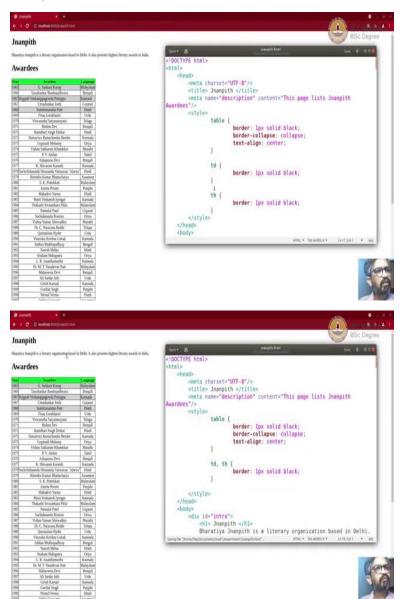


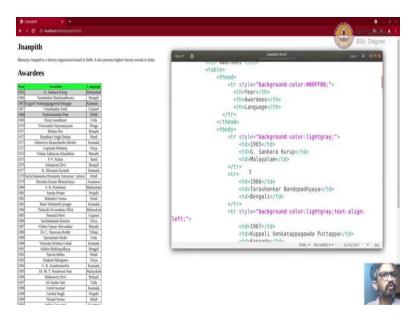


Let us see if we can do the same thing for cells. Because each cell we are saying that, we are going to make it look like border: 1 pixel solid black, we are going to do it for all the cells, yet we are not done. But let us see whether you can do here. So, let us style a t. Similarly, td. I am going to copy the style of the td and put it here. And remove currently all the td styles. So, that is it there is only one td that we had background border.

So, I am going to save and reload. Now, you can see can see the efficiency of it. Now, the style has been applied to all the td, across the document, as you can see, but we still have our th here, it still has a style, if I remove this, say, let us save it and refresh it, it is gone here, you can see it on the border is gone in that header.

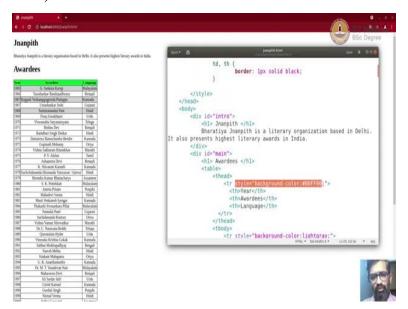
(Refer Slide Time: 5:51)





Now, I can do similarly for th too, it is a similar style. So, you can copy paste th here. And I can refresh it, so it come back. Now, there is one more optimization that we can do see, the style for td and style for th is same. So, what we could do is, we could say td, th for both of them apply the same style. I can save and do refresh. So, the border remains the same. There you go. So, we have efficiently removed that part.

(Refer Slide Time: 6:37)





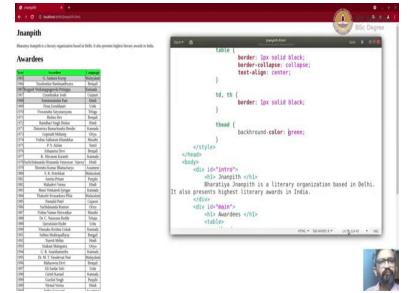
Jnanpith

aurtys frampift is a Birrary organisaton based in DeBs. It also present highest fitniny awards in India.

```
See Grant Service Control of Service Service C
```

```
clocytre html>
chtml>
chtml
c
```





Jnanpith ...

Burates humin's a literary ominisation based in Delhi, it also mesons history Dimary quark in India

```
New Owners | Language | Comment | Co
```

```
table {
    border: lpx solid black;
    border-collapse: collapse;
    text-align: center;
}

td, th {
    border: lpx solid black;
}

thead {
    backhround-color: red;
}

</style=

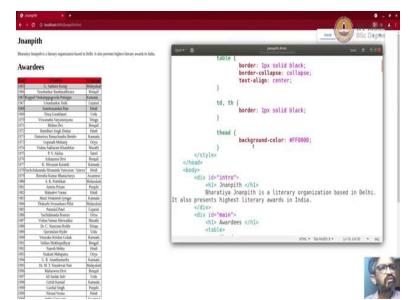
</style=

</style=

//style=

//
```







Jnanpith

ctys Inaquifs in a literary organization based in Delts. It also presents highest literary awards in India.

```
18 G. Sadar Kengy Melajahan
18 G. Sadar Kengya Melajahan
18 G. Sadar Kengya Melajahan
18 G. Sadar Melajahan
```

```
Awardees*/>

**style>

table {
    border: lpx solid black;
    border-collapse: (ollapse;
    text-align: center;
}

td, th {
    border: lpx solid black;
}

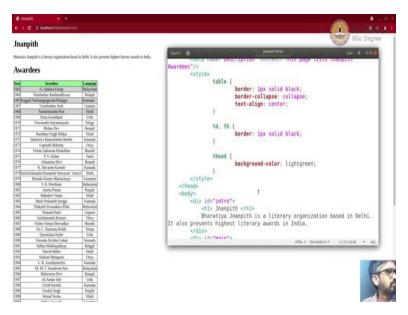
thead {
    background-color: green;
}

**style>

**lead**

**lead*
```

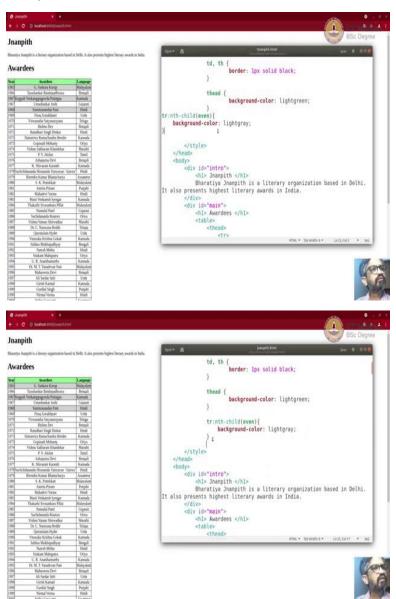


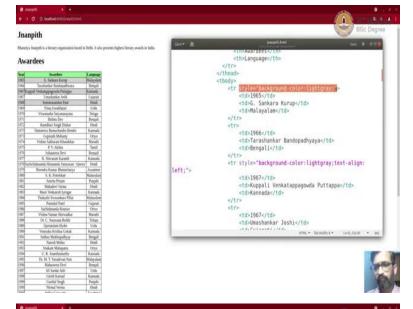


Now we have not removed the tr, background, we could actually define that we can just define the background for a thead instead of for tr. So that, what we apply to tr will not apply everywhere. So, let us remove this style. And say for table thead apply a background color. Let us say green, semi colon, save this. So, it is there, it is done, you can even make it let us say red. So, it is become, oh, red is not a named color, so assume RR is is FF0000. Oh, there is a spelling mistake here. Save here. I will reload it is become red.

So, that is one thing to remember. And I think it will be easy to identify if I make a spelling mistake, let us say it does not highlight. So, it is easy to see in a editor where code is colored or theme is applied to the code. So, background then becomes proper. If you make a spelling mistake, it goes to the default for example like that, and there is no color. So, I am going to apply this and make this green again, save it and refresh it, refresh it. It is green. I am going to make it light green, that seems better.

(Refer Slide Time: 9:06)





Jnanpith

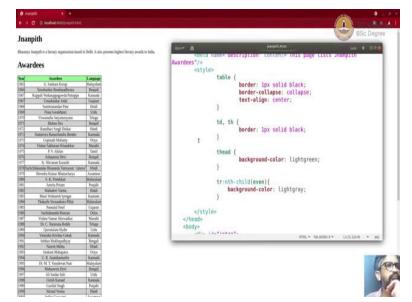
Bhasains framin's is a literary optimization based in TheRs. It also reviews highest literary quark in India

```
Name (C. Sentan Acre)

Service (S. Sentan Ac
```

```
| SSC Degree | Store |
```





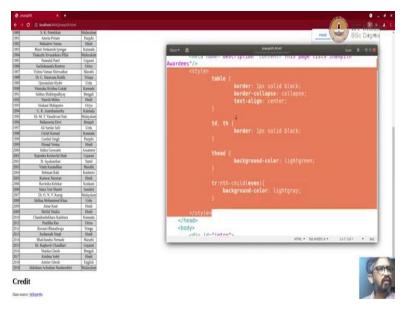


Jnanpith

atyu Inaqidi is a litetay organization hased in Delhi. It also present highest literary awash in Indi

Year	Awarders	Languag
1960	G. Sankara Koray	Malayale
1966	Tanahatkar Bahkstaffrans	Braud
1967	Kappali Verkasappapowda Patrappa Etmashankar Joshi	Kertish
1947	L'mashunkar Joshi	Gejerati
1968	Suntranandan Part	Hindi
1969	Fireq Goraklepart	Unite
1970	Virwanatha Satyanarayana	Triup: Broph
1971	Biolou Dev	Bengdi
1972	Randuct Singly Dicket	Hink
1913	Danatreya Rama/handra Bendire	Kennad
1972	Gepinath Muhanty	Oring
1974	Vidno Sáldurats Khandeker	Metabi
1975	F.V. Alslan	Tend
10%	Ashapuna Devi	Broad
1977	R. Slovaram Karamb	Kamai
19/95	edebolatanda Hirananda Vatayayan "Ajner	W Hodi
1979	Bioridia Kignar Bhatacherya	Assatra
1961	S. K. Potrkkatt	Malayala
1960	America Prisans	Parjols
1962	Mahadrei Varita	Hind
1963	Masii Vinkatesh Iyengar	Kennal
1064	Tlukathi Strasinkara Pillal	Malerala
1985	Femalal Petri	Gejánát
1996	Suchidananda Routray	Orne
1967	Violen-Venae Shirwadker	Manehi
1900	Dr. C. Narayana Bridgy	Teluga
1369	Questalain Nyler	Undu
1090	Vinnyska Kristina Gokuk	Kantud
1990	Solden Muldispalliyay Namb Milta	Dengel
1992	North Metra	Briggi
1993	Stuken Mehapers	Orna
1954	U. R. Atomboneby	Kattud
1995	Dr. M. T. Vasadross Natt	Milaysia
1996	Maharweta Devi	Benjul
1997	Ali Serlar Nht	Cirds
1996	Girch Kenad	Kentud
1999	Combal Singh	Prejate
1999	Nemal Verna	Hink



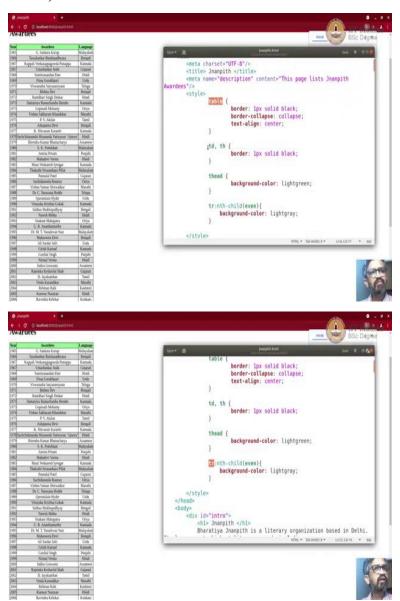


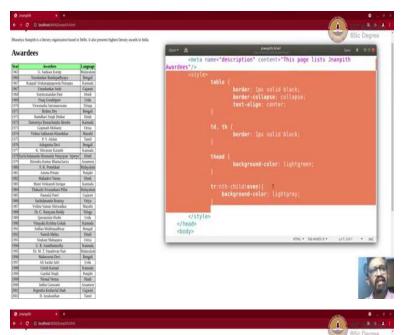
Now, we still have one more problem where we wanted to make each row separate and wanted to add a different color for odd row, different color for even row. There are many ways to do it. I am going to take the simpler way. One way is to do define a class odd and the even and you can add a class. But let us continue and uses another CSS selector.

I am just going to paste that here first. And explain to you later, it says, for every nth child of a tr, if it is even apply gray. First of all, I will just going to remove this and I am going to remove these things, I am going to remove this, saving. Let us bring that back. So, it says nth child n being the number. And then if it is even then apply this background color, light gray. I am going to save this and reload. So, you can see now, every even row is gray. I could do even odd, it changes, starting with the first row. But we were doing Ctrl Z, even. So, we are doing that. That ends actually styling the same thing, using the header.

Now you can see the amount of work, all the style is limited to head part. And it is optimized, it is minimal, and you can edit at the same place. And since it is Cascading Style Sheet, styles at the lower level can overwrite each other. And you might want to be careful when you are styling them. But That is the beauty of Cascading Style Sheets.

(Refer Slide Time: 11:46)





```
Swift Swiften Kompany Medical Control of Control Kompany Medical Control Kompany Medical Control of Control of
```

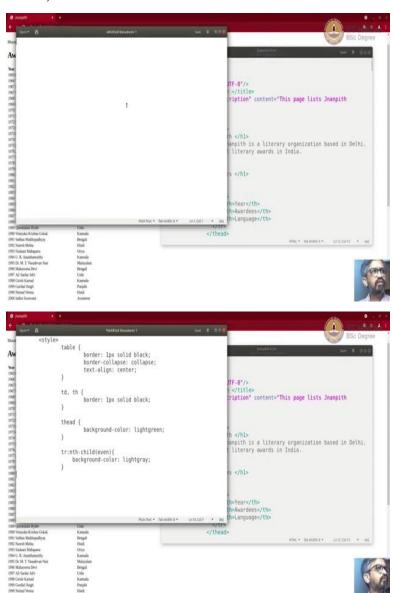


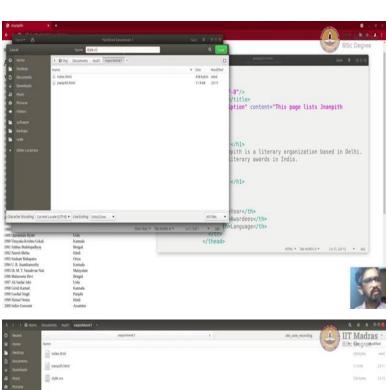


Now, we have applied all the styles based on tag names here, td, th, thead, even tr and maybe nth child but it is still tr. So, there are many ways to do this either by tag name, or class name, or id and many other ways. We are going to learn that in the next screencast. But let us assume this for now and continue.

Now, this is good if there is one only jnanpith.html, which is using this style. Let us say you wanted to use this style across many files. Now either, what you should do is you should copy paste the style part in each of the HTML files and load it and whenever there is a change, you have to go back to each HTML and change it. Or you could externalize into a CSS file or a style file. So let us cut this, let us cut this, it is gone from this. So, to come back to the previous shape.

(Refer Slide Time: 13:00)









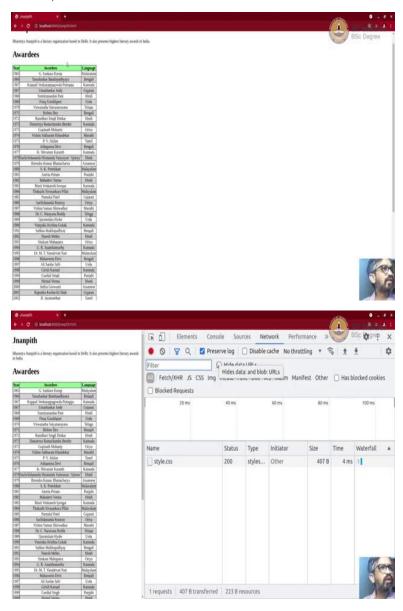
Now, I am going to create a new file edit Ctrl N. gedit opens a completely new document. And I am going to pull this style and remove this style tags. And I am going to save this into the same folder. I want to style save it as style.css, I saved that style.css. Now, if I go to the file browser, you can see it is in the same folder, style.css.

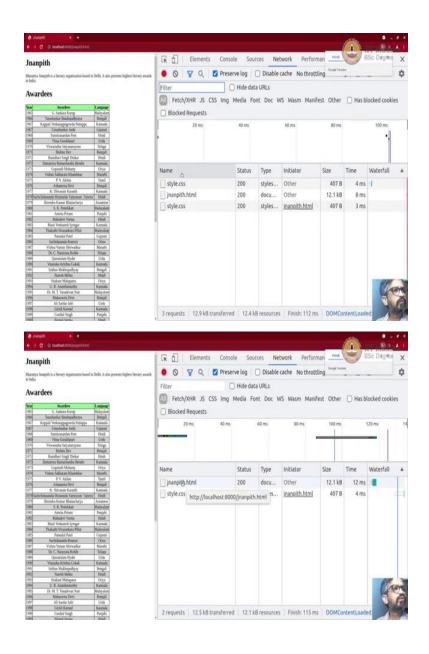
Now, the style.css contains all the style elements, I am just going to make it look little more good looking. Bring it back to the starting space, and then remove the extra white spaces just to make it look clean. Anyway, let us make it readable so that others can understand. And then save it.

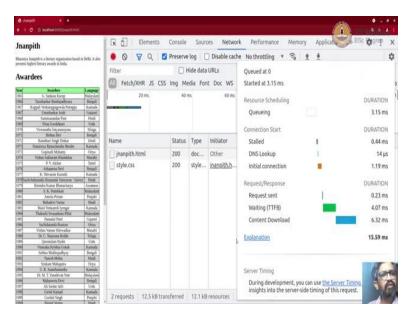
Now, somehow, we have to include this inside this document so that they get used in this. That is done by adding go style link. I am just going to add a single line of code inside the head. It tells

link a stylesheet. And the data comes from, for the style comes from style.css. And this is the name of this thing. And if it hosted somewhere else, you might have to give the complete URL of this style CSS. But if both of them are in the same folder, you can just give style.css.

(Refer Slide Time: 14:46)



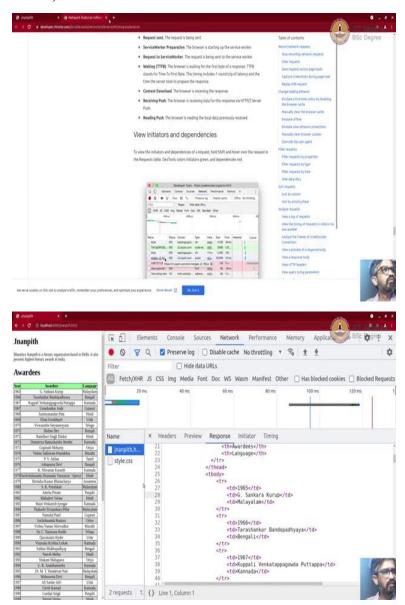


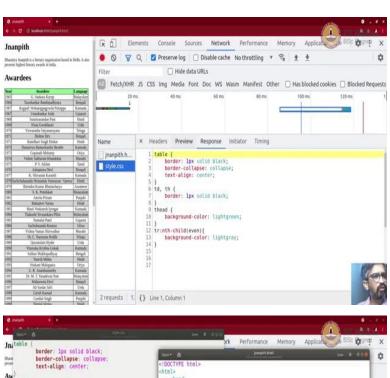


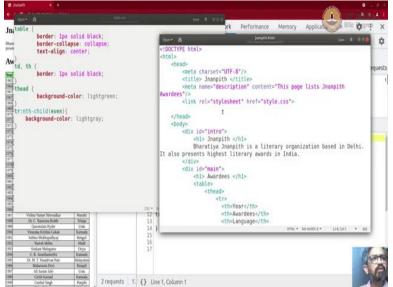
Now, if I save this, and let us go back to the browser and see Ctrl R, you can see style has been applied. Now, how it has been applied. Let us go back to inspect. Now, go to network and let us do all and I am just going to reload again, just for our clarity. See here, the jnanpith, I think this was a little confusing, let me clear it first. I am going to reload it.

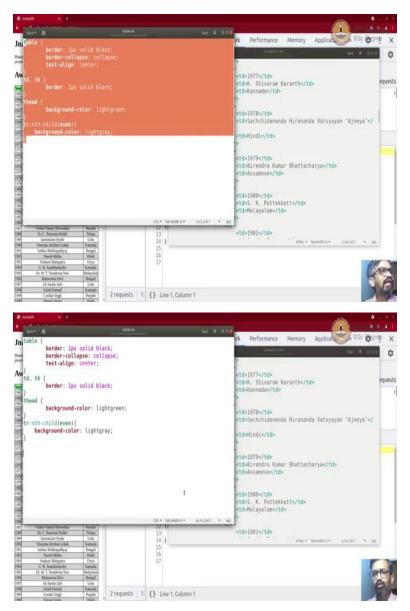
So, jnanpith.html was downloaded first from the server. And then the style.css was downloaded. And you can see that initiator for this style.css download was jnanpith.html. And very interestingly, you can see the waterfall. Waterfall is how it got down later, afterward, so you can see here, how the content were downloaded, if you are interested in it, we can go over it in the next session or something.

(Refer Slide Time: 15:56)









But you can read the explanation here to see how the timing works. But basically, first jnanpith.html got downloaded, you can see the content. And then style got downloaded. And HTML is rendered from top to bottom in a browser. So, the stylesheet got loaded. And then style is available to style and then data got loaded and then got rendered. That is it actually for this session.

So, what we learnt is separating out the style from inline into a header is efficient, but actually make into a separate style dot CSS is much more useful. And if you want to use it across many files, you could use it and you can share the style, even between the project if you want to. That is all for this in this session. Thank you so much. Bye.