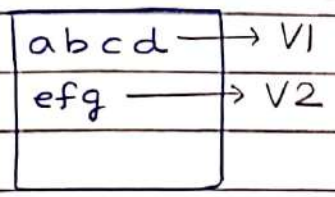
26/12/2022

Version Control Provides total development freedom.

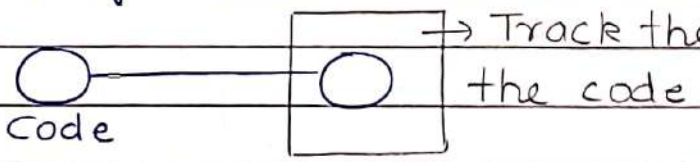


Word document

efg was written accidently so Ctrl+Z (undo) to go back to version 1.

If there was no version control, then we won't be able to go back.

Similarly we need version control in the development



Ex → Git, Subversion, Perforce

How does it work? → commits

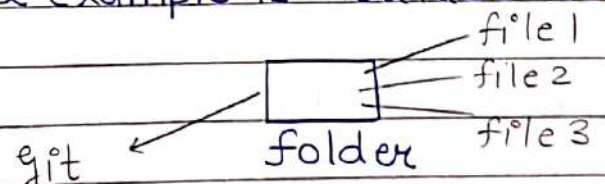
This makes save points that saves our project. We commit these save points.

S1 ·
S2 ·
S3 ·
] Commited changes

What is difference between Git and Github?

Github ⇒ hosting service for git repostory → folder

Parallel example is YouTube as the video hosting service



Tool keeps track of modification.

It (Github) is also used for collaboration.

Commands

1) git --version → Tells the version of our git. Also verify whether git is downloaded or not.

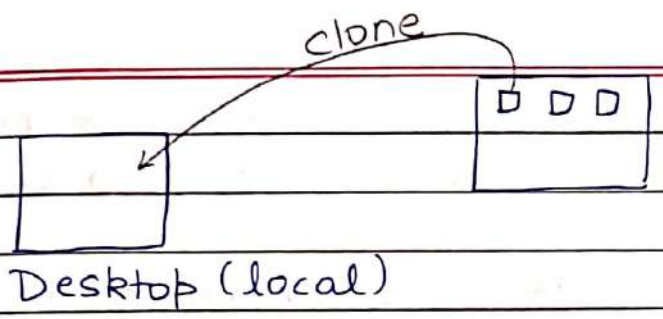
2) git config --global user.name "Bhavya Bhalla-27"
git config -- global user.email "bhavya@gmail.com"
These are used to setup our git
3) git config --list → List of settings & we can see the username & email in this.

Creating a git repository
1) Create an empty folder in Desktop.
2) With the help of command prompt, go to the folder
cd → tells the current working directory
cd Desktop → To go/migrate to desktop
3) When in that folder, run a command
git status → Tells the status of git repo.
Here we get an error → Not a git repository.
4) git init → To initialize the folder as git repo.
A hidden .git folder will be created. We can see this by enabling show hidden items.

[.git] → contain meta data of git repo.

Note → The main difference b/w git repo & folder is the .git folder
5) Now if we run git status, then we don't see an error.
6) Now make a text document & save in that folder. Now we get to see after we run git status that there are untracked files & this is not yet commited.
↳ (change is final if commit is done)
git clone To collaborate with someone, this can be used.

Scanned with Cam



git clone https://github ---- hello/

↑ URL of git repo which we want to clone

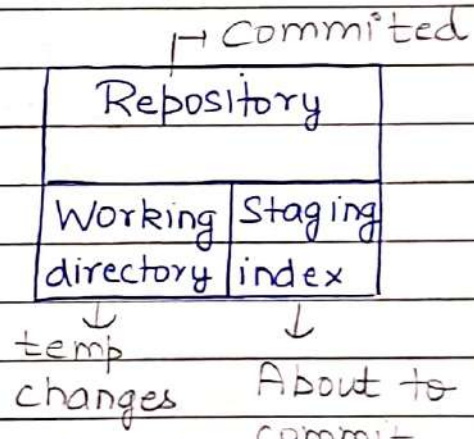
↑ folder name

git diff This command is used to know what changes have been done

Lifecycle of change

Repo

1) file1.js → file1'.js
2) file2.js → file2'.js
3) file3.js → file3'.js



Changes are not done directly. git add and git commit command is used to commit the changes.

Working directory → Staging index → Repository

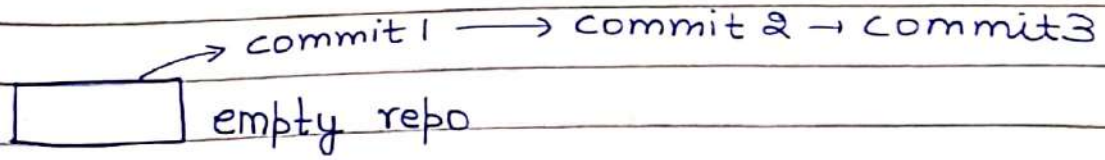
↑ git add ↑ git commit

file1.js → commit id = ABC

file1'.js → commit id = ABC2

Commit Id changes

Reviewing a Repo history



Scanned with Cam

→ gives so much information

1) git log → On the top our latest commit & then we move to other commits previously done.

Press enter to go to other commits also.

Press q to move out

→ Can be any number (non-ve integer)

git log -3 → To see latest 3 commits

git log -p → Commits + git diff (shows what is modified)

git log --oneline → Shows commit Id & commit message only.

git log --stat → Shows commit & also show which files have been changed, how many insertions & deletions are done.

2) git show ----- → To know about a particular commit

↑ commit Id

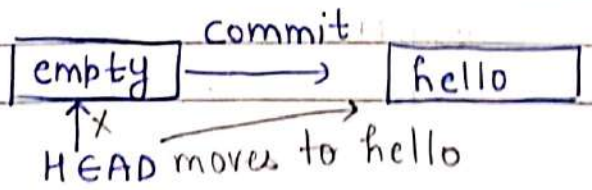
How to commit? git commit -am "Added" ↳ To commit & add

Step-1 git add → Add the file to track (new file)
Puts in stagin area (old file)

Step-2 git commit -m "Added a new file"
↳ commit message

Note- git add. → Add all the files for tracking
↳ This is not recommended

git add file_name
name of file



Scanned with Cam

HEAD points to the latest commit.
If we modified our file & then we realize that we don't need that modification, then we can do git restore file_name & we move to the HEAD.

git repo → .cpp, .js, .css
.doc, .txt → Don't want to track
(How ??)

Use .gitignore

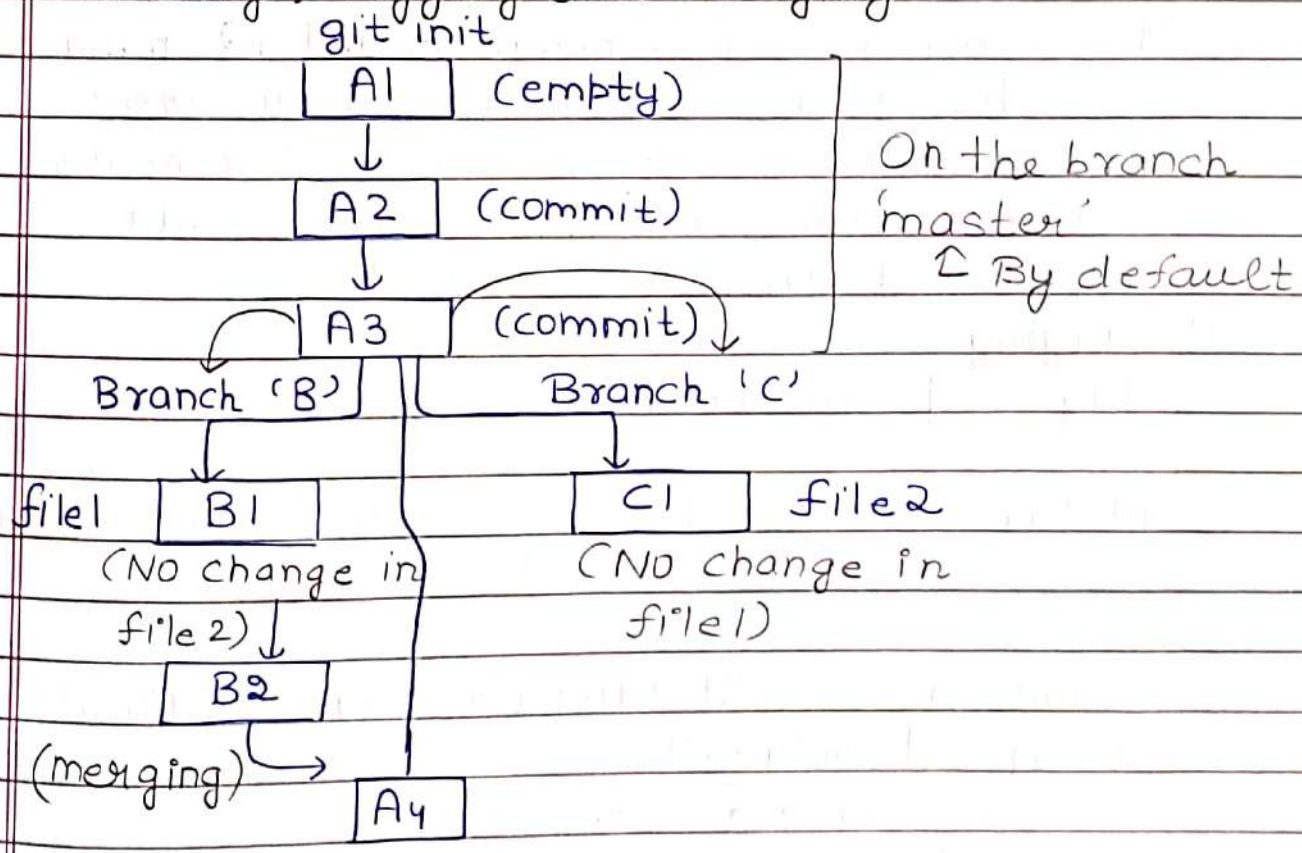
→ means any pattern (name)

```
*.txt
*.doc
```

→ ignores txt & doc files.

.gitignore

Branching, Tagging and Merging



1) Creating a branch
git branch → Tells what all branches are present & at which branch we are *master → (Here)
b1
b2

Scanned with Cam

git branch b1 → Creates a branch b1
git checkout b1 → To go to branch b1
get checkout -b b2 → Makes branch b2 & then we move to b2.

2) Merging the branches to master
Go to branch master
git merge b1 → To merge b1 to master

3) Delete a branch
git branch -d b1 → Deletes branch b1.

Note →

```
1  void b1 () {
2
3  }
4
```

Branch b1

```
1  void b2 () {
2
3  }
4
```

Branch b2

Now when we try to merge b1 and b2 in master, as both code are at same lines in different branches, this is merge conflicts & we need to resolve on our own & then use git add to tell git that conflict is resolved.

4) Tagging
Tag a specific commit.

git tag -a betaV1.0 ---- -m "Beta Release"
↑
commit Id

commit Id (HEAD → master, tag : betaV1.0)
git tag -d betaV1.0
name of tag

5) git stash

Head → | file A | file B |
working dir

Remote repo
↓
Some commits added.

Scanned with Cam

{file A, file B} Stash Area

First git stash & then git pull

git stash { file A, file B } Latest

Latest files are brought in with the help of git pull.

Very useful command

git stash list → (To see the stash area)

git stash apply → To bring back the files from stash to working dir.

Note Good practice is to commit on our own branch rather than on master as we try to keep master super stable.

git push -u origin master → Push to the remote (Github).

Undo commits

It is not a good practice to undo commits & it is rarely done.

1) Git commit --amend → amend most recent commit {HEAD}
2) git revert → reverts given commit
3) git reset → deletes commit (most dangerous command)
   ↳ (Changes HEAD & commit is not visible)

We also have to give commit Id in git revert.

git reset --soft commit Id

HEAD points here now

(i) --soft
(ii) --mixed
(iii) -hard

Scanned with Cam

In --hard, local changes gets discard.

file1	→ local changes
⋮	

Working dir

In --mixed, local changes remains intact
↳ modification will be shown

In --soft, changes will be shown in staged area. (staged modification)

Note → commit Id can be replaced by HEAD~

By git commit --amend, we can change the commit message & that ~~too~~ of most recent commit.

readme.md → File with descriptive text about our repository.

To push from local to remote, we can do
git push -u origin master
↳ branch to push

Note → Commit message should be short & precise. Typically lesser than 50 characters.

Pull request
To merge changes from some other branch to the master. Can be used so that someone else can also review the changes.

Scanned with Cam

W3 schools

MDN reference
↳ Check if stuck at any place.

classmate
Date
Page

27/12/2022

HTML → Not a programming language
Hyper Text Markup Language
Many pages linked — Use of annotations / tags which is used to markup content. ↳ (like paragraph, anchor etc.)
Many websites have linked pages.
↳ Almost all

HTML is the basic building block. It provides skeleton to the webpage. It gives the description of webpage.
CSS → formatting / appearance
JS → Functionality is provided.

HTML doc → Browser → web page
rendering (display)
↳ machine code to human readable format. It is basically loading of web page

HTML + CSS + JS ⇒ full stack application.
Tags in HTML
↳ keyword which tells browser how to render different elements.
<p> → Paragraph
<h1> → Heading
<img> → image
<a> → anchor (link)

index.html
extension

<h1> Namaste Dunia </h1> } This is HTML element
Opening tag — Content — closing tag

Scanned with Cam

Comments in HTML

To make notes. This will not be rendered by browser.

<!-- This is comment -->

Note → HTML is not case sensitive.

How can HTML doc be opened?

1) Text editor
2) Browser

] 2 ways only

Text element

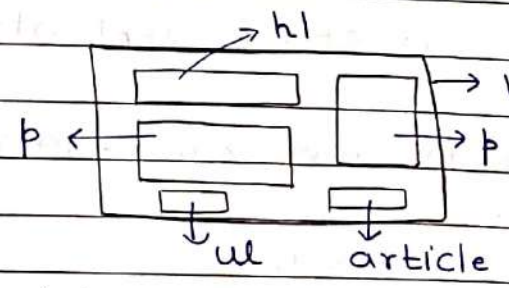
Tags → 100+ html tags (Don't learn all, just google it).

Two types of text element

1) Block element ⇒ <p>, <h1> to <h6>, <ul>, <ol> <article>, <section>
2) Inline element ⇒ , , <a>

emphasize tag

↳ To bold



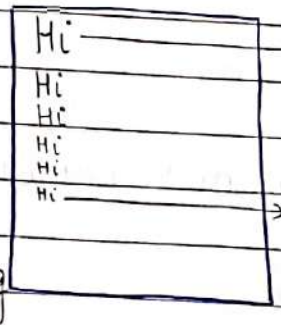
→ Webpage has content.

* Content is divided using the block elements.
* Inline elements ⇒ differentiate specific part of the content

Heading tag

<h1> Hi </h1> → Largest
<h2> Hi </h2>
<h3> Hi </h3>
<h4> Hi </h4>
<h5> Hi </h5>
<h6> Hi </h6> → Smallest

Size decreasing



→ Biggest

→ Smallest

Scanned with Cam

Homework → *Have a look at MDN website → ✓
*Check Description list → ✓
*Any HTML element not having attribute? → ✓

Paragraph tag
 This is a paragraph

div tag → Very useful. (Just like a container)
This is a block level element.
 ---
id is used to identify just as our name

 tag
Line break tag is used to give a new line. It behaves as enter.
Also note that it does not have a closing tag.
Empty tag
It is self closing tag.
Ex →
 , <img/> , <input/>

List
1) Ordered list → Has some order
1. a) <ol> tag is used
2. b)
⋮ ⋮

2) Unordered list → Does not have some order
Bullets
• ◻ <ul> tag is used
• ◻

<ol>
<li> --- </li> → list item
</ol>

3) Description list → <dl> tag is used
<dd> → To give description
<dt> → To give heading (subheadings)

img tag → Where to take (Path of image)
<img src = " " alt = " " height = "100" />
↳ Display text if image not rendered

Scanned with Cam

Homework → * What happens when we close an empty tag?
* UTF-8 kitne character ko support krta hai?

→ property

src, alt, height etc are attributes.

Note → img tag is inline element

All HTML elements can have attributes.
↳ always in opening tag

id →
class → All elements can have them as attribute
src → img tag
href → anchor tag

Style tag
This is used to add CSS to the HTML page.

```
<div id="love"> Hello </div>
<style>
    #love {
        color : red;
    }
</style>
```

→ To access id. For class use dot (.)

anchor tag
This is used to add link to our website.

```
<a href="www.thecodehelp.in">Jeewan</a>
```

link of the website

↑ This will be displayed on the screen.

HTML Boilerplate (Template)

1) <!DOCTYPE html> → Tells the browser that it is a HTML v5 document.
2) <html lang="en">→ whole webpage content will be in english.
3) <head> → have metadata of the HTML document → data about data.
↳ can be title, character set, viewport, style, script.

Scanned with Cam

4) <body> → All the content of webpage will be inside/enclosed within this.

meta tag → Not displayed to user.
↳ Store the information about the webpage. It basically defines metadata about HTML doc.

It comes inside the <head> tag always.

We can specify character set, page description, keyword, author, viewport

→ It supports 2164864 Characters (standard)

<meta charset = "utf-8">
↳ character encoding

viewport ⇒ area visible on the screen
initial-scale → When page is loaded, what should be the initial zoom level.

title tag
It comes under the <head> and it will be the name of our website.

Note → <meta http-equiv = "X-UA-Compatible" content = "IE = edge">

This line is for compatability to the edge browser.

Range of initial scale is 0.1 to 10.0

Favicon icon which appear to the left of the title

CodeHelp

28/12/2022 Favicon

Extra notes from MDN

1) We should not use the heading elements to resize the text.
2) It is not a good practice to use multiple h1 tags in same webpage.
3) It is a good practice that we use the h1 tag, then h2 and so on. Don't skip heading levels.

Scanned with CamS

4) Paragraph tag is a block level element.
5) Do not use br tag to create margins b/w paragraphs.
6) With ol tag, we can use type attribute
   type = "a" → Lowercase
   type = "A" → Uppercase
   type = "i" → Lowercase Roman numerals
   type = "I" → Uppercase " "
7) There is also start attribute from which we can specify where to start numbering
   <ol start = "4"> → Starts numbering from 4
   ↳ Only used for type = "1" i.e default
8) In dl tag, there can be
   (i) single term, multiple description
   (ii) multiple terms, multiple description
   (iii) multiple terms, single description
   (iv) Single term, single description
9) While adding link to our website, if we want that that website should open in other tab, we can use attribute target = "_blank".
10) <head> tag does not have any attribute.
11) If we close an empty tag, then nothing happens & everything works fine.

Scanned with Cam

29/12/2022

Ques Why we don't have h7, h8 and so on?
h6 is already very small & hence h7, h8 -- will be unreadable & hence not there in HTML.

Nested br
 Bhavya is <br> a good boy
↳ nested break

Bhavya is
a good boy

Attributes of anchor tag (target)
1) target = "_blank" → to open the link in new tab.
2) target = "_self" → This is by default
3) target = "_parent" → Page is opened in parent frame.
4) target = "_top" → Opens linked document in full body of window.

hr tag
This tag is used to get / print horizontal line in our webpage

Ques How to make the horizontal line red in color?
<hr color = "red">
attribute

Note → Hi
Hello

} Output will be straight line. This is poet problem.

<pre> tag → To fix poet problem
This will print the text exactly same as written in the HTML document

<pre> Hi
Hello
</pre>
} Output → Hi
Hello

Scanned with Cam

Homework
* Read quotation & citation tags
↳ blockquote, q, cite, address, abbr
* How to add a favicon in HTML?

Inline CSS → style

Text color changes to brown

Changes background color

Changes size of font

Changes font style

Shifts text to the center

Formatting text

<u> → underline

<b> tag → To bold text

<i> tag → To make text italic

<small> tag → To make text smaller

<del> tag → Puts a strike on text [~~para~~]

<mark> tag → Highlights the text

Mathematical expressions (Superscript & subscript)

 x ^{2 → x^2 →}

 a _{1 → a_1 →}

sup & sub have closing tags

Note → li * 2 → Click enter, then <li> </li> <li> </li>] Automatic generated

Tables in HTML
↳ In form of rows & columns

<table> → To create table

<tr> → Table row

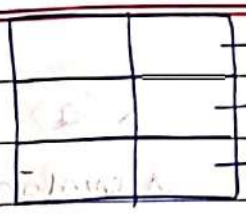
<td> → Table data

* To create lines, then we need to add border
<table border = "2 px">

* border-collapse : collapse ; → single line b/w rows.

Scanned with Cam

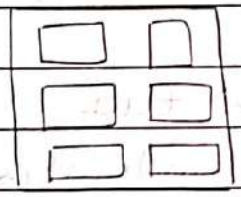
<table>



→ <thead>
→ th tag {Heading}
→ Content {<tbody>}
→ Summary {<tfoot>}

} It is good practice.

```
<thead>
    <th> -- </th>
</thead>
<tbody>
    <tr> -- </tr>
</tbody>
<tfoot>
    <tr> --- </tr>
</tfoot>
</table>
```



Border-collapse

button tag
This is used to add a button in the website

Forms in HTML

```
<label for="uname">Username :</label>
<input type = "text" name :"uname">
```

Username : [ ]

Note → It is good practice to use tables however.

```
<input type = "password">
```
↳ Text will be in dots & hidden.

```
<form action = "____">
```
↳ can be some link & can be anything else

```
<input type = "submit" value = "submit">
```
↳ Creates a button ↳ What is written inside the button

Scanned with Cam

1/01/2023

Extra notes from MDN

1) <a href = " " download> -- </a> → When we click the link, content gets downloaded because of the download attribute.

2) <q> tag

This indicates that the enclosed text is a short inline quotation

<q> This is quotation </q>

"This is quotation"

Double inverted commas are generated.

3) <blockquote> tag

This indicates that the enclosed text is an extendend quotation. This is can identified by indentation.

<blockquote> This is quotation </blockquote>

This is quotation

↑ indentation ↑ indentation

4) <cite> tag

This is used to describe a reference to a cited creative work, for example → book, research paper, essay etc.

 <cite> This is </cite> paragraph

This is paragraph.

(somewhat in italic style).

Note → cite is used as an attribute as well in the blockquote & q tags. This specifies a URL to a document that explains the quote.

URL

<q cite = " " > -- </q>

<blockquote cite = " " > -- </blockquote>

Scanned with Cam

→ O/p will be in italic font style.

5) <address> tag

This indicates that the enclosed HTML provides contact information for a person or people or for an organization

```
<address>
    <a href="mailto:jim@gmail.com">-- </a>
</address>
```

Whenever the link is clicked, mail application automatically opens up.

6) <abbr> tag

This represents an abbreviation or acronym.

```
<abbr title="Hyper Text Markup Language">HTML
</abbr>
```

HTML

Whenever we hover the mouse HTML, we get to see its full form which is written in title attribute.

7) Favicon in HTML

Favicon is a small image displayed next (left) to the page title in the browser tab.

Note → Usually 16 X 16 pixels in size & stored in GIF, PNG or ICO.

Steps to add Favicon → This is general & works for all devices.

1) Save the image on desktop
2) Go to site realfavicongenerator & upload image
3) Scroll down and then click generate favicon & code.
4) Download favicon package
5) Copy code & paste under head tag.
6) Now extract the favicon package and then copy the folder in the working directory & change folder name

Scanned with Cam

to Favicon.

7) Now modify all href by adding Favicon infront of it.

Code of adding favicon

<link rel = "icon" type = "image /x-icon" href = " _ ">

Path where image is present.

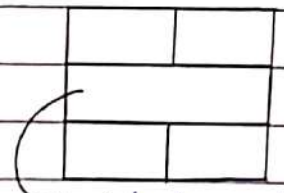
Note → <a href = "#" > -- </a> → We will reach to the top position.

8) Rowspan and colspan in tables

(i) Rowspan

<td rowspan = "3">

(ii) Colspan



<td colspan = "2">

Scanned with Cam

3/01/2023

Depth in tables

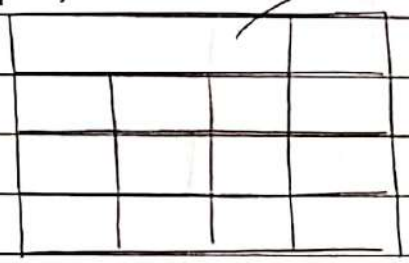
```
table, td, tr {
        border : 2 px solid black;
}
```

Decides the width of the border.

✓ Spanning in tables

(i) Columspan (colspan)

↳ Spans in direxⁿ of columns

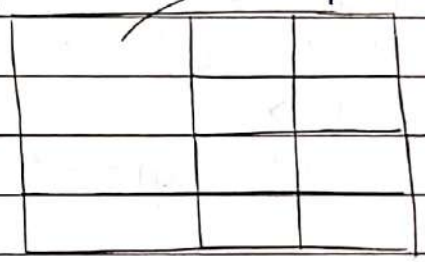


spans to 3 columns

colspan = "3"

(ii) Rowspan (rowspan)

↳ Spans in direxⁿ of rows.



spans to 3 rows

rowspan = "3"

Note → Cells can be both rowspan and colspan.

Caption tag This is used to name the table.

<caption> -- </caption>

✓ Grouping in tables

(i) colgroup To style specific columns, we use colgroup.

```
<colgroup>
    <col span="3" style="background-color:red">
</colgroup>
```

This will change bg color of first 3 columns

span = " " → By default picks value 1.

Note → There is no tag named as <rowgroup>. If we want to style rows either we can use class on rows or if we want to style all rows then thead, tbody etc. tags to be used

Scanned with Cam

Homework 3) Click image & then go to source website

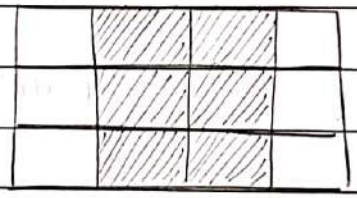
1) Alternate sol^n to rowgroup ?
↳ (does not exist)

2) Last page of cheatsheet HTML

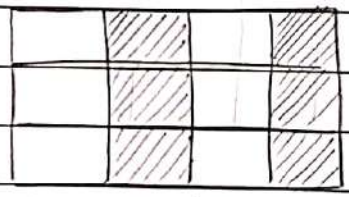
Case-1

<col span = "1"> → This means do not do anything

<col span = "2" style = "-- "> → Will style 2nd & 3rd columns.



Case-2



```
<col span = "1">
<col span = "1" style = "-- ">
<col span = "1">
<col span = "1" style = "-- ">
```

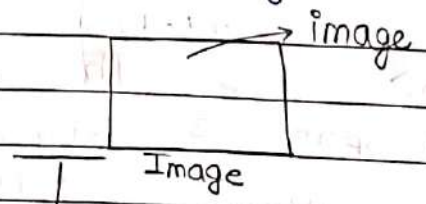
Hidding columns

<col span = "1" style = "visibility : collapse ; ">

To hide column

Caption in image

```
<figure>
    <img src = " _ " alt = " ">
    <figcaption > Image </figcaption>
</figure>
```



↳ Will have indentation as well.

Scanned with Cam

5/01/2023

Characters and symbols in HTML

In HTML documents, some symbols may not be directly available on the keyboard.

1) Copyright symbol : ©
2) Dollar symbol : $
3) Ampersand symbol : &
4) Greater than symbol : >
5) Less than symbol : <

Note → If we want to make image behave as a link, then do this

<a href="__"><img src="__" alt="__"></a>

<img> tag enclosed within the <a> tag.

Links

Hyperlinks can be created via anchor tag → a tag

1) Calling on number by clicking on number.

<a href="tel : +917652924838">call</a>

tel is Used

Phone number

2) Sending mail by clicking on link

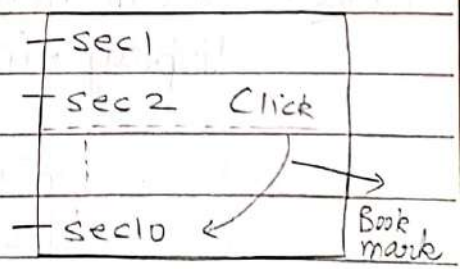
<a href="mailto : ___@gmail.com">mail</a>

mail to is used

mail id

3) Bookmark

Suppose that there are multiple sections, then by bookmark we can jump to a particular section.



Scanned with Cam

for id, # is used.

```
<a href="#Resources"> Click </a>
```

```
<h2 id="Resources"> ... </h2>
```

Transition is not smooth

But this will be like immediate switch & it feels like a jerk. To make this scrolling smooth, we can do this.

```
<style>
    html {
        scroll-behaviour: smooth;
    }
</style>
```

div tag

1) It is also known as division tag.
2) It is a generic tab. It can be used on text, image, link
   (generic container)

container that can be styled.

3) It is a block level element.
   Covers 100% width.
4) We can have any tag inside div tag, like div tag can be inside a particular div.

```
<div> First </div>
<div> Second </div>
```

First
Second

Note → We can make div behave as inline element by display: inline; in CSS (style)

id → Unique
class → Not unique

Scanned with Cam

HomeWork

1) Check div + display : inline ; → span
   span + display : block ; → div
2) Difference b/w header & h1 tag ?
3) Multiple header tag can be used?
4) Header inside footer?
5) How to put aside content to center?
6) Diff b/w main & body?

span tag
It is same as div but it is inline element.

Semantic tags
↳ Meaningful

div and span → non-semantic tags as it does not provide meaning.

1) <article> → article is independent text.
2) <aside> → content in sidebar. For example → Ads
3) <figure>
4) <figcaption>
5) <header> → introductory part of our webpage → (brand name, logo etc.)
6) <footer> → end of webpage. Author, copyright, terms etc.
7) <main>
8) <nav> → For navigation links.
9) <section> → To define a section within our webpage

Note → <article>
<section> </section>
</article>

<section>
<article> </article>
</section>

Both of them will work.

link tag → contains only attributes & is empty element.
1) To add favicon.
2) To link external style sheets.

DOM → Document object model
It is a tree like structure. Hierarchy of tags is DOM.

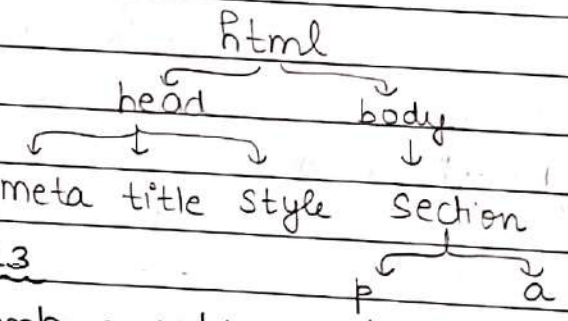
<html>
<head>
<meta>

Scanned with CamScanner

7) Create DOM for Linus Torvald's website.
8) HTML Quiz
9) HTML debugging exercise


```
<title>
<style>
</head>
<body>
<section>
    <p>
    <a>
</section>
</body>
</html>
```

Create DOM



6/1/2023

Homework questions of 5/1/23

① Div + display : inline ; → span tag
Span + display : block ; → div tag

② <header> tag → Defines the header for a document or a section. It contains information related to title and heading of related content.
<h1> tag → Defines some heading for a paragraph or some content.

③ We can have multiple header tags in our webpage but using multiple h1 tags is not recommended.

④ <header> tag can not be placed within a footer, address or another header tag.

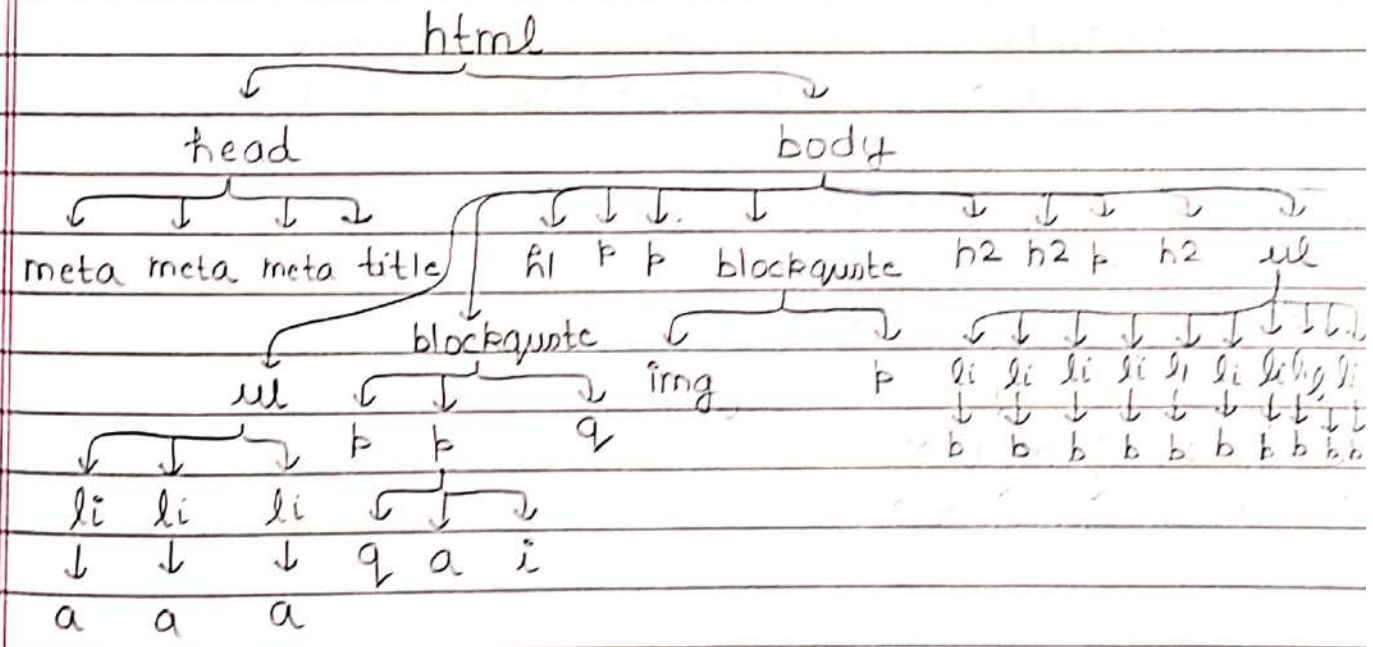
Scanned with Cam

⑤ aside {
width : 40% ;
margin : auto ;
}

This will make content inside aside to the center of the webpage.
↳ Should not have sidebar, nav links, site logos etc.

⑥ main tag → The content inside main tag should be unique to the document.
body tag → This contains all contents of the HTML document such as hyperlink, text, images etc.

⑦ DOM for Linus Torvald's website



Extra tags from cheatsheet

1) script tag (Javascript)
✓ Used for linking document to external script
✓ It is used to write code snippets (usually Javascript).

2) border attribute of img tag
We have to specify thickness of border. By default it

Scanned with Cam

is set to 0.

3) code tag
This tag formats text as code snippets.
4) ins tag
It defines a text that has been inserted into a document. Browser usually underlines the text.
5) Important attributes in forms
(i) Action → where form data should be sent when submitted.
(ii) target → where to display the form's response
(iii) autocomplete → has value on or off & determines whether browser should automatically fill in the form.
(iv) novalidate → specifies that form should not be validated
(v) method → specifies which HTTP method to use when sending form data.
(vi) required → tells that input element can not be left blank.
(vii) autofocus → gives focus to input elements when page is loaded.
(viii) disabled → Prevents user from interacting with it.
(ix) placeholder → Provides hint to user what information is to be filled here.

6) textarea tag
It allows user to enter multiple lines of text as input.
7) select tag
It provides a list of options for users to choose from.
8) option tag
Creates a single option with select tag.
9) input tag

Scanned with Cam

Provides an input field for users to enter data. It has a type attribute which is used to specify which type of file /data can be entered.

10) button tag Creates a button that can be clicked to perform an action.

11) meter tag This is used to measure data within given range.

12) progress tag This is used as a progress bar to indicate completion of task.

13) dialog tag This is used to create a dialog box.

14) audio tag This is used to embed an audio file in the webpage.

15) video tag This is used to embed a video on webpage

16) time tag This is used for formatting dates & times.

17) iframe tag This is used to embed YouTube videos in webpage.

Scanned with Cam

7/01/2023

CSS : Cascading style sheets
↓
Focus on how the content is presented.

appearance, Styling, formatting

Selectors in CSS
It is a way to select element/s.

```
↱ selector
table {
    border : 1px solid black ;
}
```

property, value

```
↱ selector
p {
    font-size : larger ;
}
```

property, value

Types of selector

1) Simple selector

(i) element selector → type / tag selector
Alternative name

```
<style>
    p {
        color : red ;
    }
</style>
```

name of tag

property, value

* CSS can select HTML elements by using element's tag name.
* This will apply to all the tags like if we apply CSS on p tag, then that property will apply on all paragraphs.

(ii) class selector
We can use this if we want to style a particular thing. To select element by its class, . (period) is used.

Scanned with Cam

```
<style>
    .green {
        color : green ;
    }
    .blue {
        color : blue ;
    }
</style>
<p class = "green" > Hello </p>
<p class = "blue" > Hello </p>
```

→ This is used while selecting with the help of class.

↳ Will be green in colour

↳ Will be blue in colour.

* Class is not unique to any element
↳ used for grouping

Note → Comments in CSS can be made via /*___*/
Write comment here.

(iii) id selector

id should be unique to a particular element. We can use this to style a particular thing. To select element by its id, # is used.

```
<style>
    #red {
        color : red ;
    }
</style>
<p id = "red" > Hello </p>
```

→ This is used while selecting with the help of id.

↳ This will be red in colour

2) Pseudo-classes selector

```
Selector : pseudoclass {
        Keyword that represents a special state of
}       that element
```

Scanned with Cam

Homework
1) Universal, nested, attribute selector.
2) Play with margin, padding etc.

(Pseudoselector)

```
button : hover {
        background-color : yellow;
}
```

On hovering over the button, background-color will become yellow

Note →
```
.red : hover {
    --
}
#green : hover {
    --
}
```
Can work with both classes & ids.

3) Multiple selector (Grouping selector)
We can use / select multiple elements in the same CSS rule by seperating them with commas.

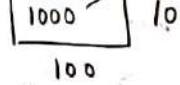
```
table, tr, td {
    --
}
```
→ multiple selectors

How to add styling in HTML?
1) Inline CSS → Hi
2) Internal CSS → style tag is used
3) External CSS → sheet is linked

Inline CSS
This is applied on the opening tag by using style attribute
Hi
Internal CSS
style tag is used and this tag is under the head tag.

```
<style>
    .green {--}
</style>
```

Scanned with Cam

3) Fonts in CSS
4) Overflow 

1000 words does not fit, then what happens?

5) Generate same ~~to~~ same card of webdev codehelp batch.

External CSS This is the most widely used. We have to make .css file & link that to HTML file.

HTML ——— <link> ———→ CSS

<link> tag is used here to link .css file to the HTML file.

Whatever we write under <Style> tag can be written in the .css file.

Link external .css file

<link rel = "stylesheet" href = "Style.css">
path of .css file

Specificity

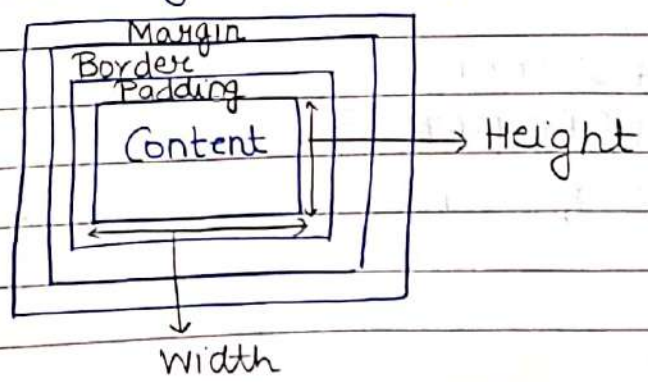
As we have multiple selectors, which of them is given priority, this is specificity.
↳ When selectors will clash.

element selector < class selector < id selector < inline CSS
Least priority ↑ ↑ Highest priority

Note → To over-ride the priority, we can use ! important & this will be applied however this is bad practice.

Box-model in CSS

This is the basic building block of CSS. According to this concept, every element on page is rectangular box and may have width, height, padding, borders & margins.



Outer side of border → margin

Scanned with Cam

Note → box-sizing : border-box; Size of box will not increase when we apply padding and content becomes smaller.

margin

* margin : 10px;
  ↳ of left, right, top, bottom
* margin-left : 10px;
  ↳ of particular side
* margin : 10px 20px;
  ↳ top, bottom ↳ left, right

Note → By default 8px margin is there.

colors in CSS

cOlors can be specified via

(i) Hexadecimal colors
(ii) RGB colors
(iii) Predefined / cross-browser color names
(iv) RGBA colors
(v) HSL colors
(vi) HSLA colors

1) Hexadecimal colors

#(R)(G)(B)

All values from 0 to F.

Blue → #0000FF
Green → #00FF00
Red → #FF0000
Black → #000000
White → #FFFFFF

2) RGB colors

rgb (_, _, _) 0 to 255

red green blue

Blue → rgb (0, 0, 255)
Green → rgb (0, 255, 0)

Scanned with Cam

6) Difference b/w 1% and 1 viewport (1vh, 1vw)

Red → rgb (255, 0, 0)
Black → rgb (0, 0, 0)
White → rgb (255, 255, 255)

3) Pre-defined color names
140 colors already in HTML and CSS.
blue, black, green, aqua etc.

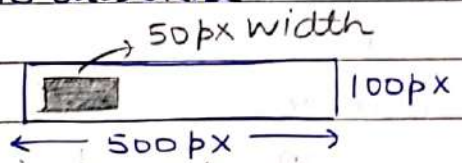
Units in CSS

1) Absolute unit → fixed
10px, 1px is absolute unit.
↳ $\frac{1}{96}$ of inch

Pixel is smallest single unit of the display but px of CSS is of fixed size i.e $\frac{1}{96}$ of inch.

mm, cm, in, px are absolute units.
Pixel of physical device & CSS is different

2) Percentage unit
div { width : 10% }
(dimension of parent)



$\frac{10}{100} \times 500 = 50$ px

3) Relative units
(i) Relative to font size
em → relative to parent element font size. If no parent, then font size of body tag
rem → relative to root (html).

Note → Default font size of body tag is 16px
1em → 16px
2em → 32 px

(ii) Relative to viewport → display area
vw → $\frac{1}{100}$ x width of viewport

Scanned with Cam

vh → $\frac{1}{100}$ x height of viewport

8/01/2023

Universal selectors (*)

The CSS universal selector matches elements of any type.

```
* {  → Selects all elements
    back-ground-color : yellow;
}
```

```
div * {  → Selects all the elements inside div.
    background-color : blue;
}
```

Nested selectors

To nest a selector, we simply seperate them with a space.

```
header h1 {  → This property will be applied on h1 of header tag.
    background-color : aqua;
}
```

Attribute selector

The [attribute] selector is used to select elements with a specified attribute.

```
a [target] {  → This property will be applied to anchor tag having target attribute
    background-color : green;
}
```

```
a [target = "_blank"] {  → This property will be applied on anchor tags having target = "_blank" only.
    background-color : yellow;
}
```

Scanned with Cam

Margins in CSS

* margin-top, margin-bottom, margin-right, margin-left. This is basically applying margins to specific directions.
* margin : 25px 50 px 75 px 100 px ;
  top right bottom left
* margin : 25 px 50 px 75 px ;
  top right & left bottom
* margin : 25 px 50 px ;
  top & bottom right & left
* margin : 25 px ;
  top, bottom, right, left

Note → The auto value is used to horizontally center the element within its container.

margin : inherit ; → Now margin property will be inherited from the parent container.

Padding in CSS

Padding is used to create space around an element's content, inside of any defined borders. Shorthand notation will be same as that of margin property

Fonts in CSS

(i) Font-family

This property is used to specify the font of the text.

font-family : "Times New Roman" ; → Should be in inverted commas if more than one word.

font-family : Arial ;

(ii) Font-weight

This property sets how thick or thin the text should be displayed.

font-weight : normal ;

font-weight : bold ;

Scanned with Cam

→ number b/w 100 to 900

font-weight : 100 ;
font-weight : initial ; → default value (400)
font-weight : inherit ; → inherits property from parent.

(iii) Font-style
font-style : normal ; → default
italic ; → italic font style
oblique ; → oblique font style
initial ; → default
inherit ; → font style of parent container

(iv) Emphasis and importance
Emphasis → em tag
This can be replicated in CSS via the font-style italic or oblique.

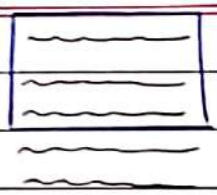
Importance → b tag
This can be replicated in CSS via the font-weight

(v) Adding external fonts
We can add external fonts via google fonts. Go to google fonts website & search for the font you want to add in website. Then select a particular style & then embed the link tag in to the <head> tag & then add font-family in the .css file or the <style> tag or the inline CSS.

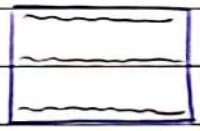
Scanned with Cam

Overflow → overflow-x, overflow-y } Can be managed also

By default overflow : visible ;



overflow : hidden ;



overflow : scroll ;

→ scroll bar is added

overflow : auto ; → Similar to scroll, but adds scrollbar only when necessary

Note → Viewport → it is calculated as % of browser's current viewport size.
% units → Calculated as % of its parent elements.

8/01/2023
HTML doubt clearing session

1) In rowspan and colspan, we can not put decimal values.
2) Monthly earnings table → Just an example. where thead tbody & tfoot can be used.

thead →	Month	Earning
tbody →	Jan	100
	Feb	200
tfoot →	Total	300

3) div is non-semantic tag
p is semantic tag
provides meaning
4) When we write CSS, then default properties will be over-ridden by what we have written.
5) It is not a good practice to make div behave as span & vice-versa.

iframe tag
Used to embed something like adding a window i.e embedding a seperate HTML document.

Scanned with Cam