Frontend web development (HTML)
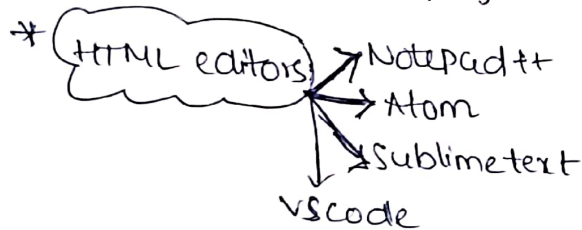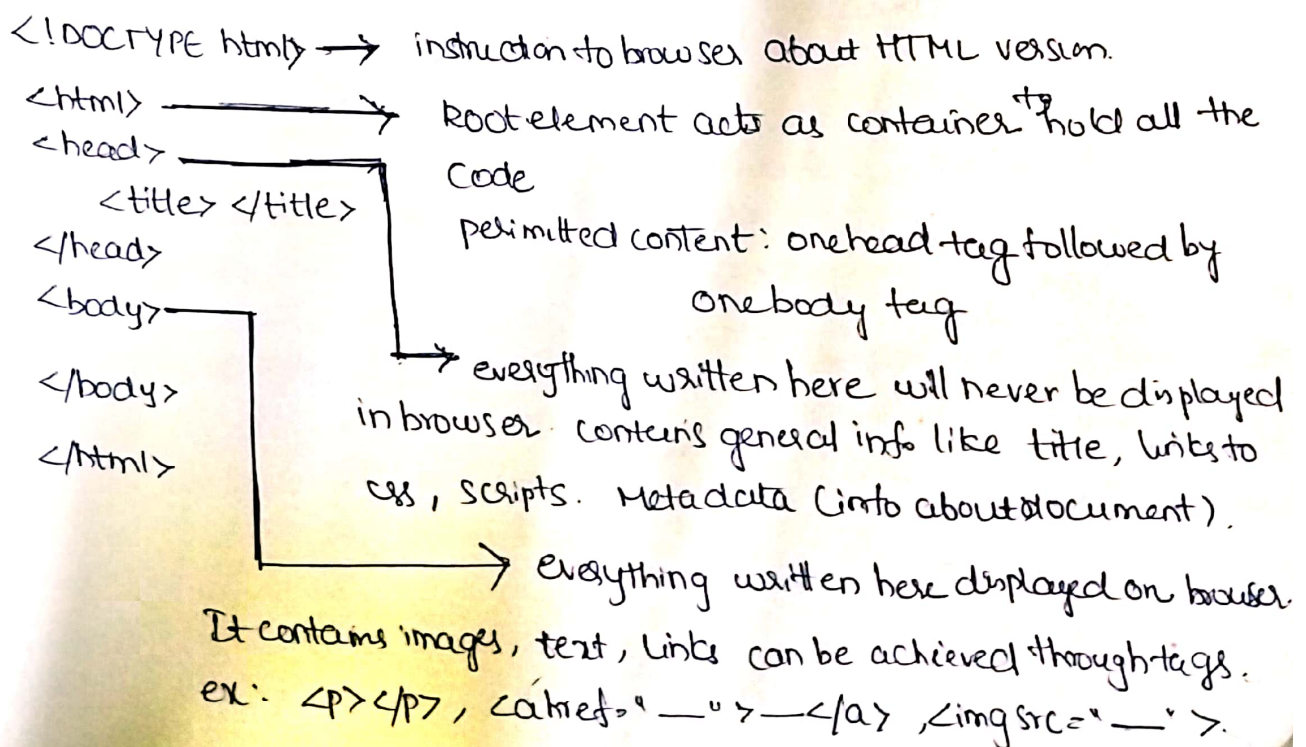
* HTML:- Hyper Text Markup language

* Hypertext:- link b/w webpages

* Markup language:- Text b/w tags which defines structure

* language to create web pages.

* HTML defines how webpage looks and how to display content with help of elements

* defines structure of our webpage

* save file with .html extension

* features of HTML:-
→ add videos/images/audios looks attractive
→ Case insensitve language
⇒ Can add links we can add references
→ Can display documents on platforms like Mac/windows/Linux etc

* (HTML editors) → Notepad++
→ Atom
→ Sublimetext
vscode

* HTML Skelton:-

```
<!DOCTYPE html> → instruction to browser about HTML version.
<html> ──────→ Root element acts as container to hold all the code
<head>
    <title></title>
</head>
<body>

</body>
</html>
```

permitted content: one head tag followed by one body tag

→ everything written here will never be displayed in browser. contains general info like title, links to css, scripts. Metadata (info about document).

→ everything written here displayed on browser.
It contains images, text, links can be achieved through tags.
ex:- <p></p>, —, <img src="___">.

* HTML Comments

→ Comments donot render on browser

→ helps to understand our code better and make it readable

→ helps to debug our code

→ 3 ways ← single line
                Multiple line

comment tag //

<!--> <p>——</p> <--> singline comment

<!-- <p>————
        ————
        ————
        ————</p> ——> Multiline comment

* HTML element

→ elements are created using tags

→ elements are used to define semantics

→ can be nested and empty

<p color= "yellow" > This is content </p>
 ↑   ↓   ~~~~~~
start   attribute   Content   └ endtag
tag   (property = value)

nested { <p>
         <p> </p>
         </p>

Types of elements :-

element ← Block level [takes up full block or width, always start from newline, always end before new line]
           Inline [takes only it requires, always start from where prev element ended]

Block level ex:-   Inline ex:-
    <p>   <span>
    <div>   < strong >
    <h1>...<h6>   <em>
       <img>
       <a>

\# Seems like we have
entered and written and
entered

* Basic tags

→ **<title>**

 * written here displayed on webpage tab

 * defines title of web page

 * <title> xyz </title>

→ **<p>**

 * defines paragraph

 * <p> xyz </p>

→ **Self closing tags** (closingtag dont require)

<hr>	** **	**<img>**
* horizontaral rule	* breakline	* to add image
* divide webpage	* Move to nextline	
	* like enter button	

→ **<h1>tag ... <h6>**

 * headingtag * h1→ most impheading * h6 leastimpheading

 h1→ more size, more bolder

 h6→ less size less bolder

→ **<strong>**

 * text to be bold   * replaced <b> tag //Html5

→ **<em>**

 * text to be italic * replaced <i> tag //Html5

 <strong>, <em> strong and emphasis tag doesnt deal with styling of website.

 <b> ⇒ bold   <strong> ⇒ important

→ **<ol>**

 * orderd list

 * seq of events that takes place in some order

 * --

 * eg. making tea

→ **<ul>**

 * unordered list

 * order is not imp

 * --

 * eg. hobbies

→ **<li>**

 * list item

 * inside ol, ul

 * --

* `<div>` and `<span>` tags

→ group diff tags

→ acts like container

→ `<div>` ---`</div>` `<span>` -- `</span>`

→ effective while styling

→ `<div>` block level `<span>` inline level

* `<img>`

→ self closing tag

→ to add images

→ `<img src="url">`.

* `<a>`

→ `<a href="url">` --`</a>`

→ add links in webpage

`<b>` ⇒ To show text in bold

`<i>` ⇒ to show text in italics

`<strong>` ⇒ To show imp text

`<em>` ⇒ to cell attentn.

* Table tag

→ to create table on webpage

→ `<tr>` → table row

`<th>` → table header

`<td>` → table data

→ enclosed in `<tr>`

`<thead>` → header data

`<tbody>` → body data

`<table>`

`<thead>`

`<tr>` `<th>` - - `</th>`

`</thead>` `</tr>` `<th>` . . `</th>`

`<tbody>`

`<tr>` `<td>` -- `</td>`

`<td>` - - `</td>`

`</tbody>` `</tr>`

`<table border="1">`

* <form tag>

→ action attribute : specifies url to send form data to

→ Method attribute : specfics type of http req (GET / POST)

< form action : " / ..... " method = "POST" >

→ <input> : to accept data from user

→ type of inputs :

<input type : " " > .

→ text : text data stores password #type = "text".

→ password : enter secure password #type = "password".

→ placeholder : temp text in input fields. #placeholder = " enter text".

→ button : include button in form # type = "button" value = " text"

→ Submit button : create submit button # type = "submit"
            all data submitted when clicked.

→ Checkbox : ability to check multiple options. # type = " checkbox".
            To check options by default set it with checked attribute.

→ Radio button : choose only single option : # type = "radio".
            keep name attribute of all options the same

→ textarea : <textarea> using rows, cols how large textarea should be.

* <label for = " " > Birthday : </label>
   <select id = "birthday" name = "Month" >
         <option> ── </option>

   </select>
   <select id = "birthday" name = "Day" >
         <option> ── </option>

   </select>

* \<label> paxwnd: \<input type="paxword" name="paxwnd"
Placeholder. "yourpaxword" pattern="of 5,10}" required title="enter
b/w 5-10">\</loabel>.

* \<label> Terms and condns.
    \<input type="checkbox" name="check">
\<label>

→ Attibutes:
* \<tag name="value"> \<tag> structure
*globalattibub: Title // add extra info.
            Style // add style info (size, color, font)
* \<img src="url" width="100"> attibute to define width in pixey
        ↳ attibute to define path

* \ link \
        ↳ define path of the links.