① Type selectors :→

<h1>Namaste world ! </h1>

↓

h1 {

property value ;

}

② Class selector :→

<h1 class = "class name">

. Classname {

Property : value ; }

③ Id Selector :→

<h1 id = "idname">

id name {

property : value ;

}

④ Attribute selectors :→

<a href = "www.example.com">

↓

a [href = "www.example.com"] {

property : value ;

}

⑤ Pseudo Class selector :→

• eg: hover ,

Syntax :- Selector : pseudo-class {

property : value ;

}

<P> Loren ipsum - - -

Dipsum gypsum - - - </P>

⑥ Combinator selectors :→

- Image Tag :→
  (i) used to embed images.
  (ii) utilizes the src attribute for image URL
  (iii) alt attribute for alternative text.
  (iv) can be resized using width & height.
  (v) self-closing, doesn't require an end tag.

  Ex: <image src = "images/gucchi.jpg" alt = "this is an image" width = "150px"> </image>.

  Ex: <vedio src = "Fork.movie" alt = "Sanchic sir scolding" height = "300px" Controls > </vedio>.

  control :→ used for playing button
  autoplay :→ to autoplay the video.

- Anchor Tag :-
   Google
  target = "blank".

- <b> → Bold tag
  <i> → italic tag
  <u> → underline tag
  <s> or <strike> → Ex. ~~priyanshu~~

- Pre Tag :→ (Preserve tags)
  <pre> ..... </Pre> → to write content as it is

- <big> .. .. </big>
- <small> .... </small>.

Ex:⇒ (a+b)² = a² + b² + 2ab.

Ex: $CH_4 + O_2 \Rightarrow H_2O + CO_2$.

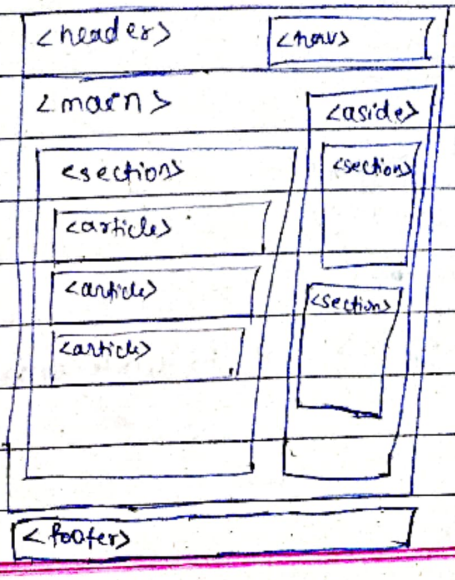
* * * * 3.1 : Character Entity Reference :⇒

Ex:
€
¢

Level-3

1. Browser Tools
   - (i) View page source
   - (ii) Inspect element
   - (iii) HTML without CSS
2. Responsive Design
   - (i) Different screen size
3. Live editing code.

* Semantic tags
  - Ex:- <header>, <footer>, <article>, <section>, <nav>
* Non-Semantic tag
  - , <i>, <b>



(Level 5)

1. List Tag :-
   - (i) ordered lists
   - (ii) Types of ordered lists
   - (iii) unordered list
2. Table tag :-
   - (i) <tr>, <td>, <th> tags
   - (ii) Captions
   - (iii) Col spans
3. forms :-
   - (i) input tag (ii) Action attribute (iii) Name & value property
   - (iv) Label tag (v) exploring types
4. iframe tag :→

(1) list :-

```
<ol>
    <li> HTML </li>
    <li> C++ </li>
    <li> Java </li>
</ol>
```

```
<ul>
    <li> HTML </li>
    <li> C++ </li>
</ul>
```

.... <tr>, <td>, <th> Tags :-

<tr> Table Row :-
<th> Table Header :-
<td> Table data :-

Ex :-

```
<table border = "1">
    <tr>
        <td> C </td>
        <td> 5/10 </td>
        <td> 10 years </td>
    </tr>
```

<!-- DOCSTYPE -->

Ex. <body>
<table border="1">
<caption>Student information</caption>
<tr>
<th> Language </th>
<th> Competency </th>
<th> years </th>
</tr>
<tr>
<td> C </td>
<td> 5/10 </td>
<td> 10 years </td>
</tr> </table>
</body>

Ex:- <tr>
<th Colspan="3">Tech g am good at</th>

Forms

<form>
Text: <input type = "text" placeholder = "enter text">
Password : <input types = "password" >
</form>

Ex:- <form>
UserName: <input type = "text">
Password : <input type = "password">
</form>

Ex: UserName : <input type = "text" placeholder = "email">

required = "True"

Ex: <!DOCType html>
<body>
<form action = "/submit.png" method = "post">
<input type = "text" name = "name">
<input type = "submit" value = "submit">
</form>

Ex: Date: <input type = "date">
file: <input type = "file">
color: <input type = "color">
Price Range: <input type = "range">
Button: <input type = "button" value = "Name">

Ex: <label>male</label> <input type = "radio" name = "gender" value = "male">
<label>female</label> <input type = "radio" name = "gender" value = "female">

Ex: <input type = "checkbox" name = "java" id = "c">.

<select name = "country" id = "country">
<option value = "India"> India </selection>

Ex: <form>
<label for = "country"> select your country </label>
<select name = "country" id = "country">
<option value = "India"> India </option>
<option value = "US"> US </option>
</select>
</form>

Ex: <textarea name = "story" id = "story"
cols = "20" rows = "4">.

Level 5

<iframe src = "youtube video link" height = "300" width = "300" </iframe>.

Bonus level

- Github.

(i) What is version control:-

(i) Definition : A system to track changes in files over time.

(ii) Types :- Centralized (like SVN) and distributed (like Git)

(iii) Purpose : Helps in teamwork & fixes mistakes.

(iv) Snapshots : Each 'Commit' Saves a file version.

(v) Branching :- Lets you work on different tasks separately.

(vi) Merge :- Combines changes from different people.

(vii) Undo : Easy to revert to older file version.

Git	GitHub
→ Software opensource.	→ Web service for hosting
→ Records a snapshot of file changes.	→ create a personal copy of another user's repository
	→ track bugs & feature ideas.