Assignment – 4

1.How are inline and block elements different from each other?

- A block element always starts on a new line and takes up the full width available that is it stretches out to the left and right as far as it can. The element is a block-level element. While an inline element does not start on a new line and only takes up as much width as necessary. This element is an inline-level element.
- The **display** property sets or returns the element's **display** type. ... However, if you set **display**:**none** , it hides the entire element, while visibility:hidden means that the contents of the element will be invisible, but the element stays in its original position and size.

2.Explain the difference between visibility:hidden and display:none

- The **display:none** property hides the entire element, while **visibility:hidden** means that the contents of the element will be invisible, but the element stays in its original position and size.
- The d**isplay:none** property keeps the property in the HTML file but removes it from the webpage , while **visibility:hidden** property keeps it on the webpage but doesn’t show it.

3. Explain the clear and float properties.

- The CSS float property specifies how an element should float.
- The CSS clear property specifies what elements can float beside the cleared element and on which side.

4. Explain difference between absolute, relative,fixed and static.

The position property specifies the type of positioning method used for an element. There are five different position values:

1. **Static**
2. **Relative**
3. **Fixed**
4. **Absolute**

- **Static** : An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page.
- **Relative** : An element with position: relative; is positioned relative to its normal position.
- **Fixed** : An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled.
- **Absolute** : An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

5. Write the HTML code to create a table in which there are 4 columns(ID , Employee Name, Designation, Department) and at least 6 rows. Also do some styling to it.

Demo.html

```
<!doctype html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport"
         content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <link rel="stylesheet" href="style.css">
   <title>Document</title>
</head>
<body>
<table>
   <th>ID
   </th>
   <th>
      Employee Name
   </th>
   <th>
      Designation
   </th>
   <th>
      Department
   </th>
   <tr>
      <td>1</td>
      <td>Abhilesh</td>
      <td>Senior Developer</td>
      <td>Java</td>
   </tr>
   <tr>
      <td>2</td>
      <td>Shubham</td>
      <td>Traniee</td>
      <td>Mean</td>
   </tr>
   <tr>
      <td>3</td>
      <td>Nirbhay</td>
      <td>Traniee</td>
      <td>JVM</td>
   </tr>
   <tr>
      <td>4</td>
      <td>Sagar</td>
      <td>Junior Developer</td>
      <td>Android</td>
   </tr
   <tr>
      <td>5</td>
      <td>Sachin</td>
      <td>Senior Software Engineer</td>
      <td>JVM</td>
   </tr>
   <tr>
      <td>6</td>
      <td>Ishpreet</td>
      <td>Traniee</td>
```

```
        <td>ML</td>
    </tr>
</table>
</body>
</html>
```

Style.css

```
table{
    border: 1px solid black ;
    background-color: red;
}
th{
    border: 1px solid black ;
    background-color: blue;
}
td{
    border: 1px solid black ;
    background-color: yellow;
}
```

Output -

ID	Employee Name	Designation	Department
1	Abhilesh	Senior Developer	Java
2	Shubham	Traniee	Mean
3	Nirbhay	Traniee	JVM
4	Sagar	Junior Developer	Android
5	Sachin	Senior Software Engineer	JVM
6	Ishpreet	Traniee	ML

6. Why do we use meta tags?

The <meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.

The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.

<meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript">

7. Explain box model.

CSS box model is a container which contains multiple properties including borders, margin, padding and the content itself. It is used to create the design and layout of web pages. It can be used as a toolkit for customizing the layout of different elements. The web browser renders every element as a rectangular box according to the CSS box model.
Box-Model has multiple properties in CSS. Some of them are given below:

- borders
- margins
- padding
- Content

Border Area: It is the area between the box's padding and margin. Its dimensions are given by the width and height of border.
Margin Area: This area consists of space between border and margin. The dimensions of Margin area are the margin-box width and the margin-box height. It is useful to separate the element from its neighbors.

Padding Area: It includes the element's padding. This area is actually the space around the content area and within the border box. Its dimensions are given by the width of the padding-box and the height of the padding-box.

Content Area: This area consists of content like text, image, or other media content. It is bounded by the content edge and its dimensions are given by content box width and height.

8. What are the different types of CSS Selectors?

CSS selectors are used to find the HTML elements you want to style.

CSS selectors can be divided into five categories:

- Simple selectors (select elements based on name, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo class selectors (select elements based on a certain state)
- Pseudo-elements selectors(select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

9. Define Doctype.

The <!DOCTYPE> declaration must be the very first thing in your HTML document, before the <html> tag.

The <!DOCTYPE> declaration is not an HTML tag; it is an instruction to the web browser about what version of HTML the page is written in.

10. Explain 5 HTML5 semantic tags.

Footer -
A <footer> is generally found at the bottom of a document, a section, or an article. Just like the <header> the content is generally metainformation, such as author details, legal information, and/or links to related information. It is also valid to include <section> elements within a footer.

Small -

The <small> element often appears within a <footer> element which would usually contain copyright information or legal disclaimers, and other such fine print. However, this is not intended to make the text smaller. It is just describing its content, not prescribing presentation.

Header -

The <header> element is generally found at the top of a document, a section, or an article and usually contains the main heading and some navigation and search tools.

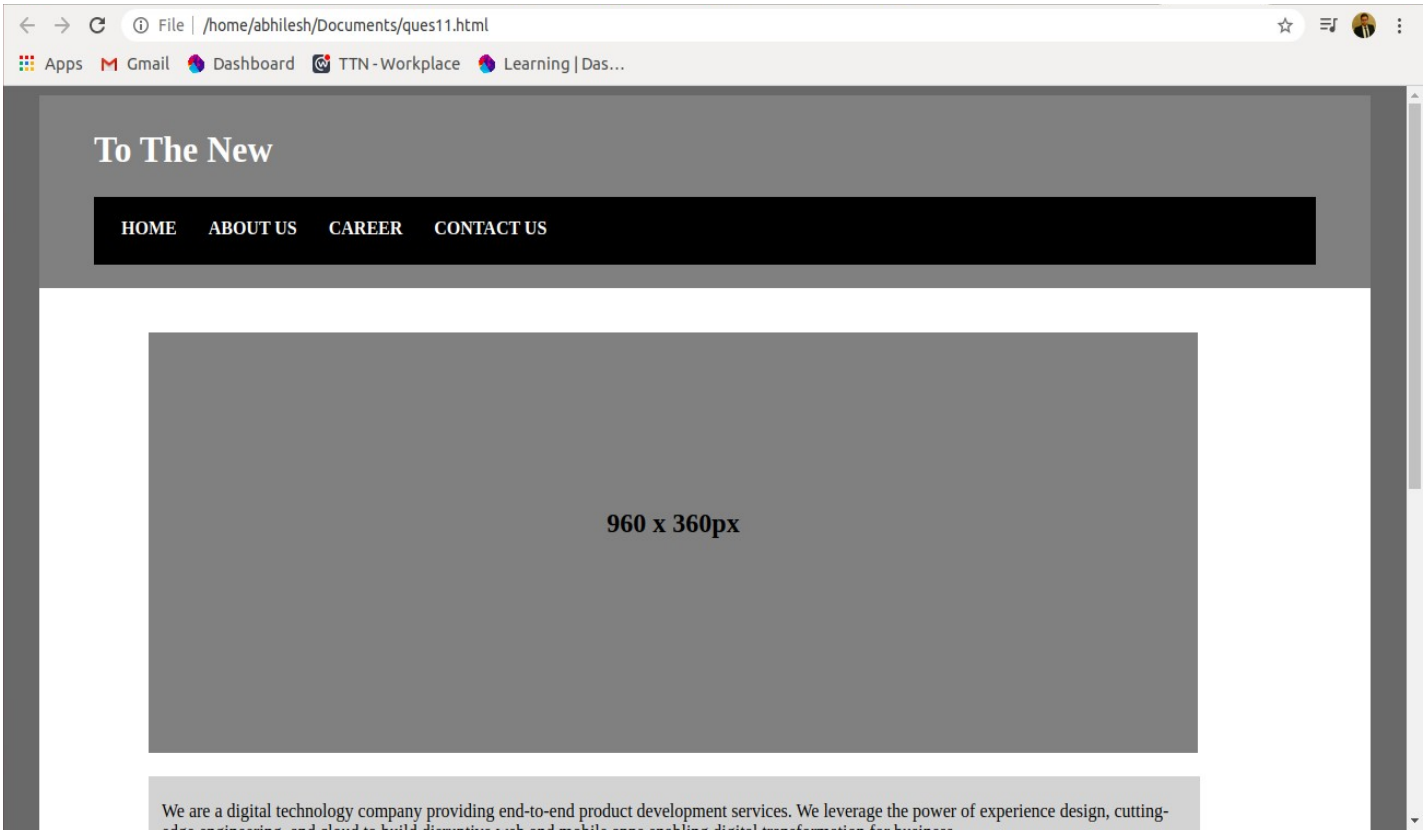
Nav -

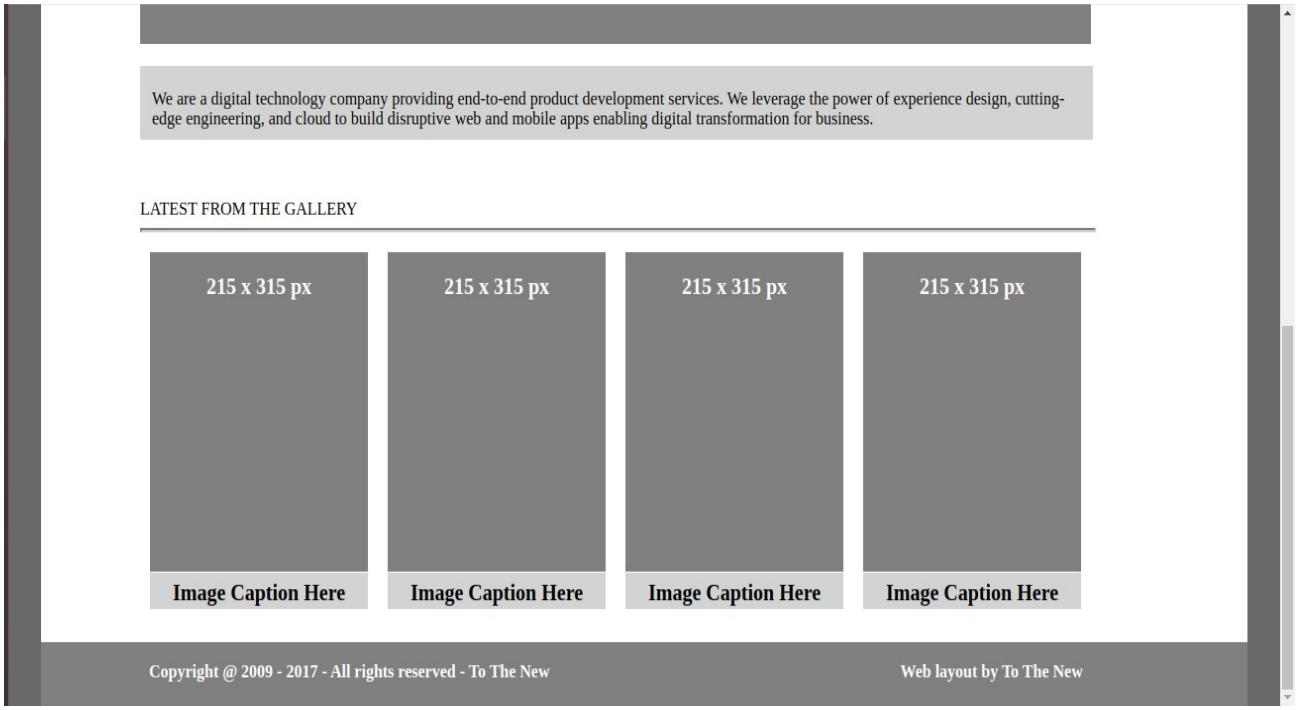
The <nav> tag defines a set of navigation links.

Section -

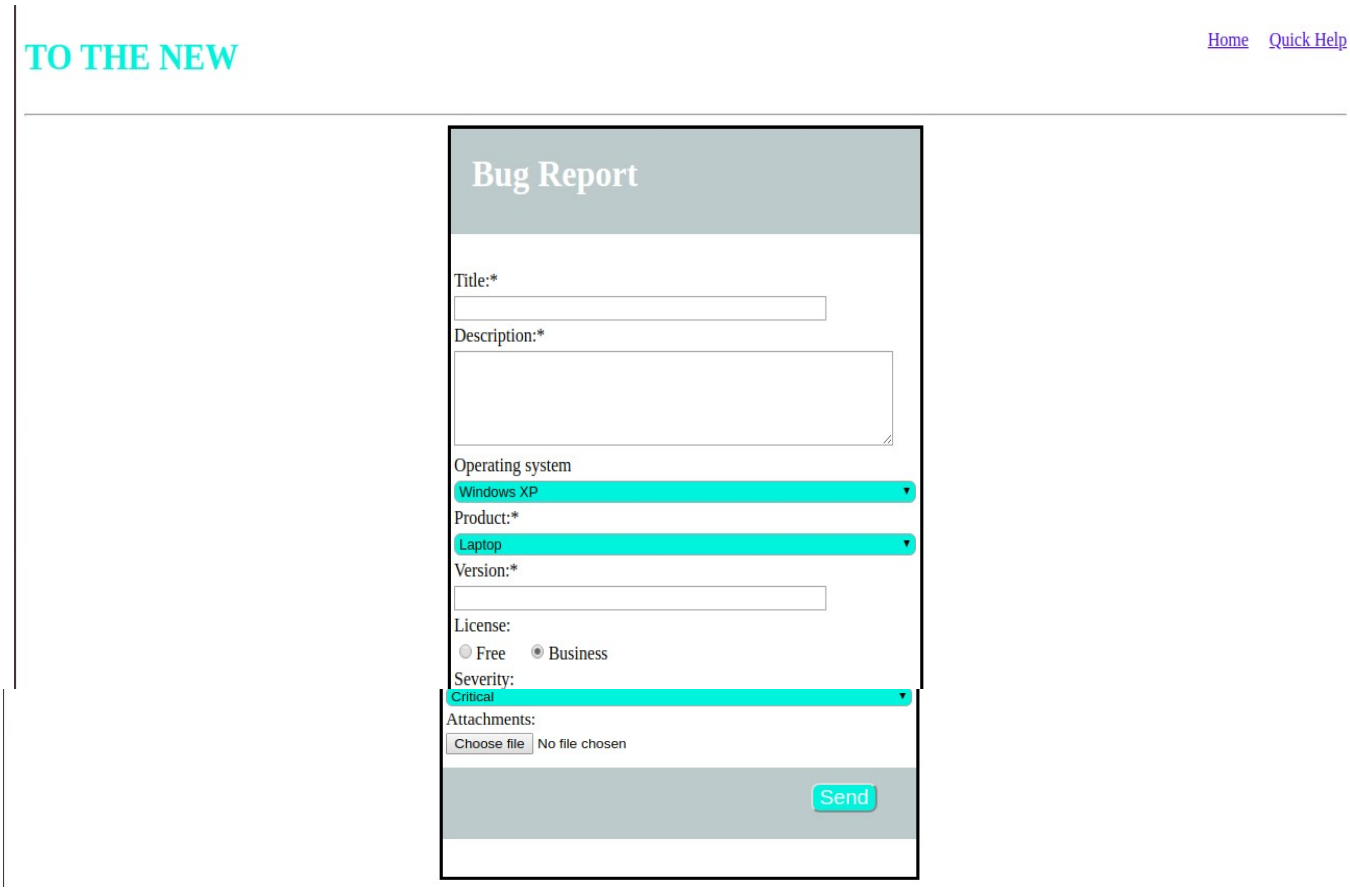
The <section> element defines a section in a document.A home page could normally be split into sections for introduction, content, and contact information.

11. Create HTML for web-page.jpg(check resources, highest weightage for answers)





12. Create HTML for form.png (check resources, highest weightage for answers)



I am also attaching the .html and .css files of ques 11 and ques 12.