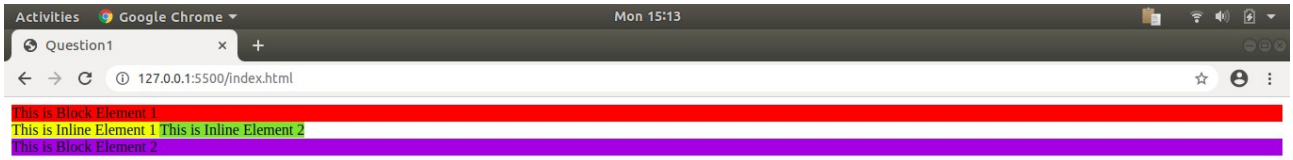
1.How are inline and block elements different from each other?

- 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 display: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.

Display:none

visibility:hidden

3. Explain the clear and float properties.

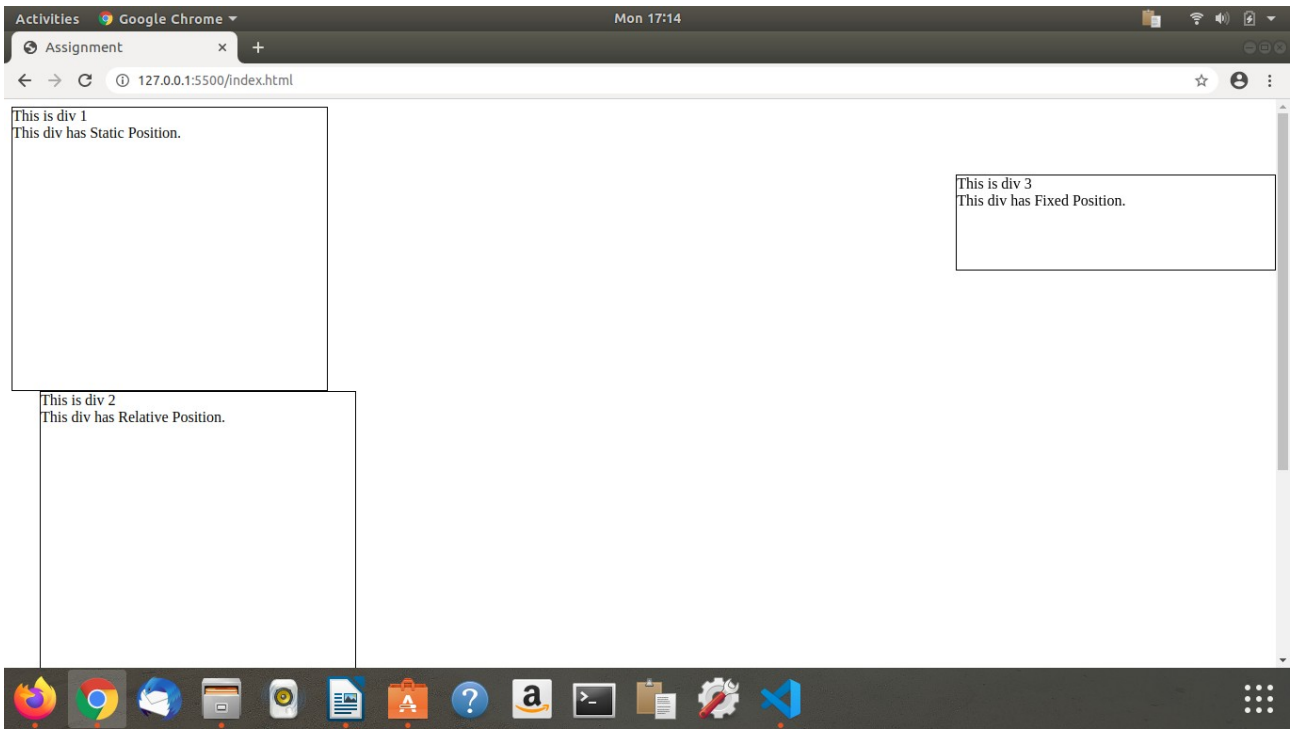
- The CSS float property specifies how an element should float.
- The CSS clear property specifies what elements can float beside the cleared element and on which side.

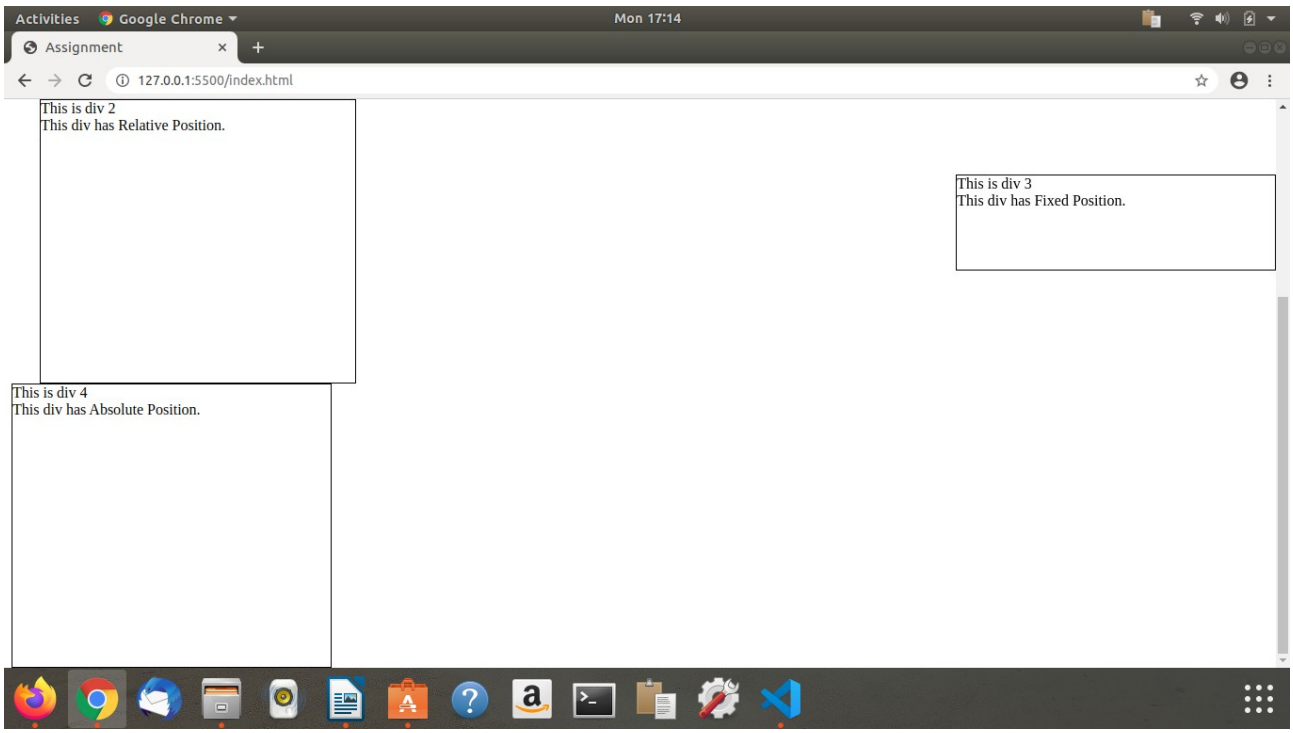
Left div uses float and Right div uses clear

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

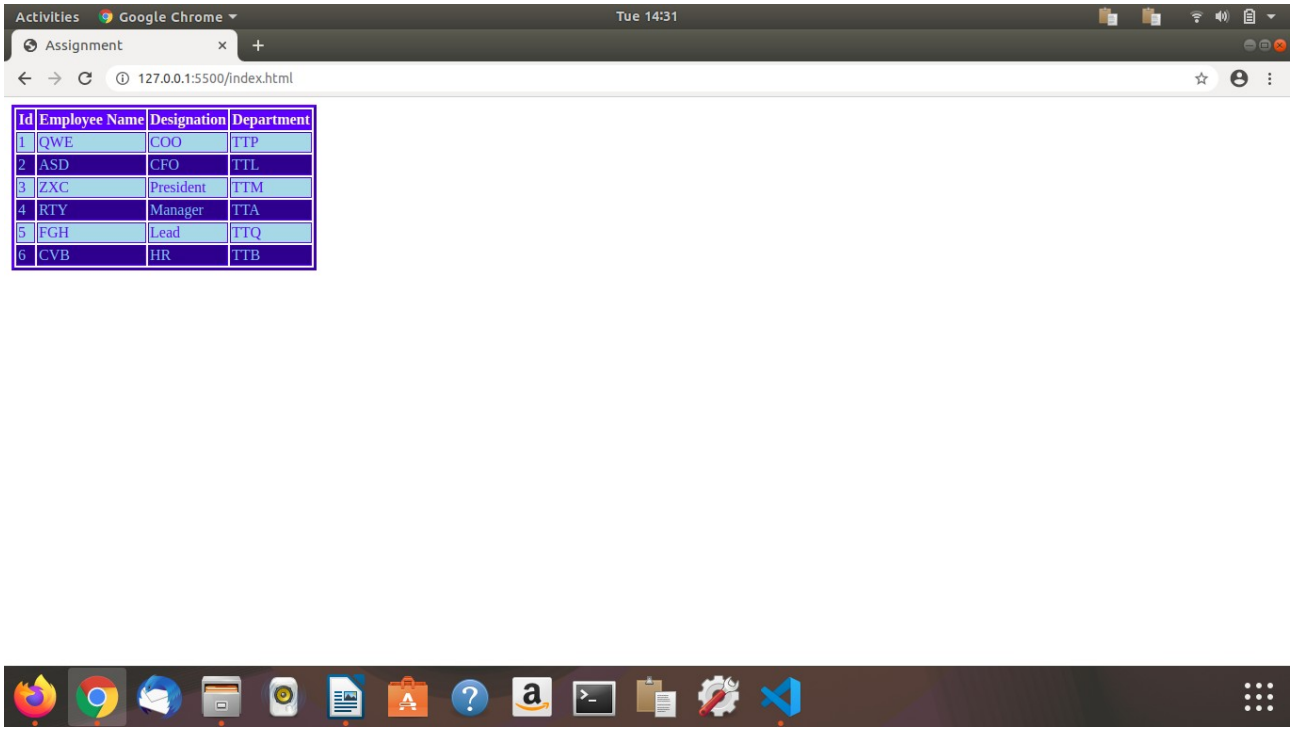
- The position property specifies the type of positioning method used for an element. There are five different position values:

- 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):


Activities
Google Chrome
Mon 17:14
Assignment
127.0.0.1:5500/index.html
This is div 1
This div has Static Position.
This is div 3
This div has Fixed Position.
This is div 2
This div has Relative Position.


Activities
Google Chrome
Mon 17:14
Assignment
127.0.0.1:5500/index.html
This is div 2
This div has Relative Position.
This is div 3
This div has Fixed Position.
This is div 4
This div has Absolute Position.

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.



Id	Employee Name	Designation	Department
1	QWE	COO	TTP
2	ASD	CFO	TTL
3	ZXC	President	TTM
4	RTY	Manager	TTA
5	FGH	Lead	TTQ
6	CVB	HR	TTB

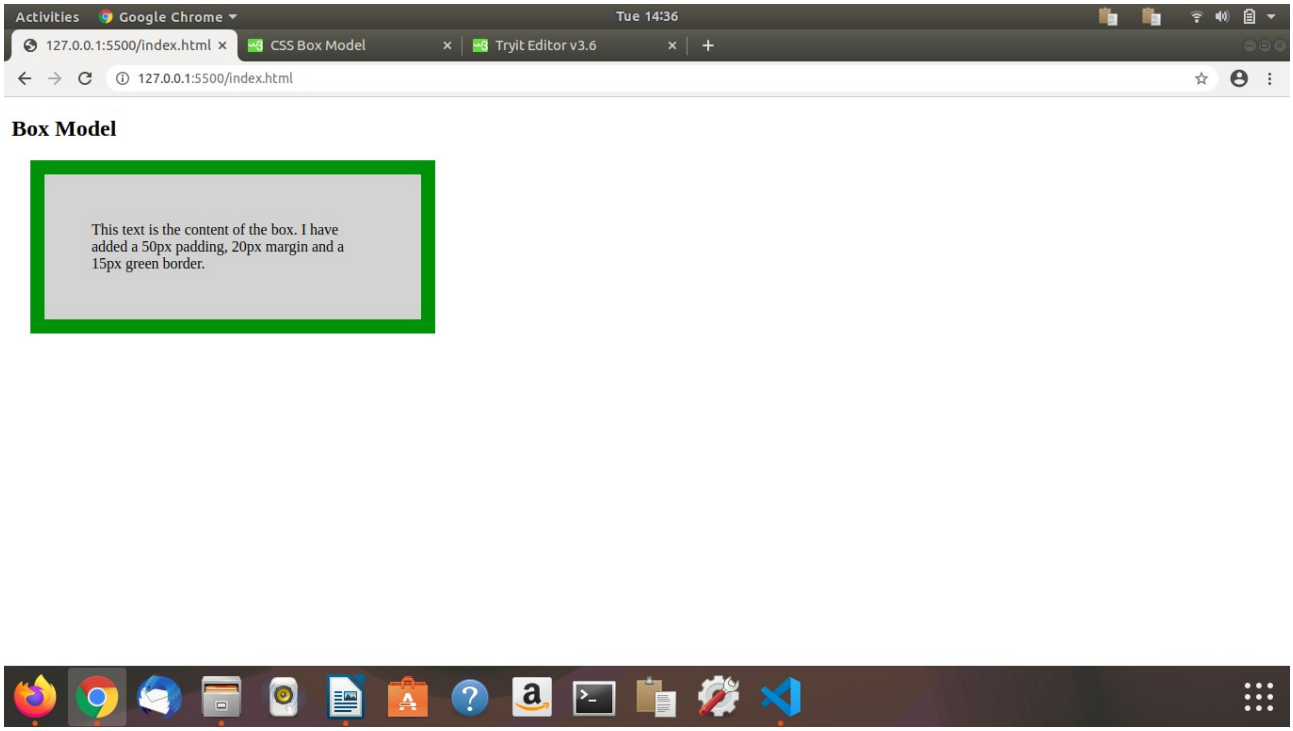
6. Why do we use meta tags?

- Metadata is data about data.
- The <meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.
- Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata.
- The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.
- HTML5 introduced a method to let web designers take control over the viewport (the user's visible area of a web page), through the <meta> tag.

```
<head>
 <meta charset="UTF-8">
 <meta name="description" content="*description about the webpage*">
 <meta name="keywords" content="*Keywords for SEO*">
 <meta name="author" content="*author of the content*">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
```

7. Explain box model.

- All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.
- The CSS box model is essentially a box that wraps around every HTML element. It consists of:
- margins: Clears an area outside the border. The margin is transparent.
- borders: A border that goes around the padding and content.
- padding: Clears an area around the content. The padding is transparent.
- content: The content of the box, where text and images appear.



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.
- In HTML 4.01, the <!DOCTYPE> declaration refers to a DTD, because HTML 4.01 was based on SGML. The DTD specifies the rules for the markup language, so that the browsers render the content correctly.
- HTML5 is not based on SGML, and therefore does not require a reference to a DTD.

10. Explain 5 HTML5 semantic tags.

- A semantic element clearly describes its meaning to both the browser and the developer.
- Examples of non-semantic elements: and - Tells nothing about its content.
- Examples of semantic elements: <form>, <table>, and <article> - Clearly defines its content.
- New Semantics offered by HTML5:

  •<aside>
  •<article>
  •<details>
  •<figcaption>
  •<figure>
  •<footer>
  •<header>
  •<main>
  •<mark>
  •<nav>
  •<section>
  •<summary>
  •<time>

The `<article>` element specifies independent, self-contained content. An article should make sense on its own, and it should be possible to read it independently from the rest of the web site

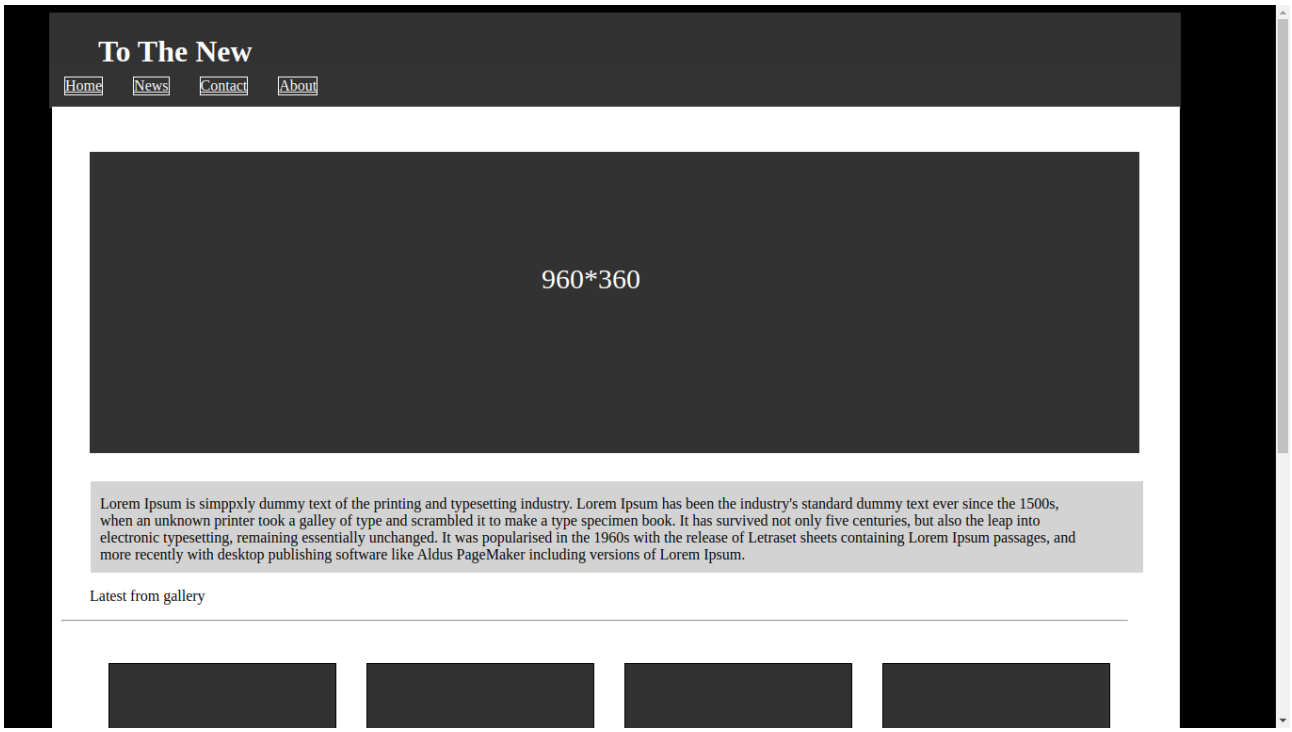
The `<header>` element specifies a header for a document or section. The `<header>` element should be used as a container for introductory content. You can have several `<header>` elements in one document.

The `<footer>` element specifies a footer for a document or section. A `<footer>` element should contain information about its containing element. A footer typically contains the author of the document, copyright information, links to terms of use, contact information, etc.

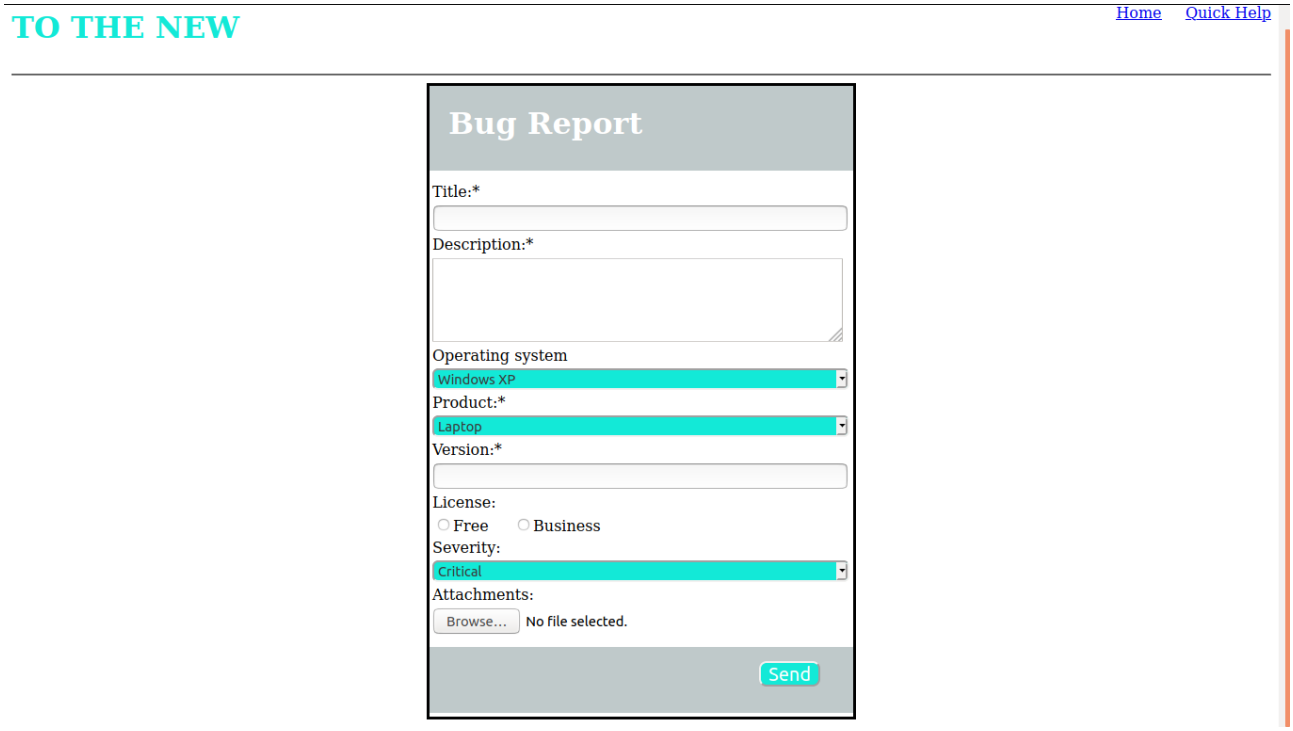
The `<nav>` element defines a set of navigation links. The `<aside>` element defines some content aside from the content it is placed in (like a sidebar).

The `<aside>` content should be related to the surrounding content.

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)



Note: 50% of marks are for last 2 exercises of creating html pages