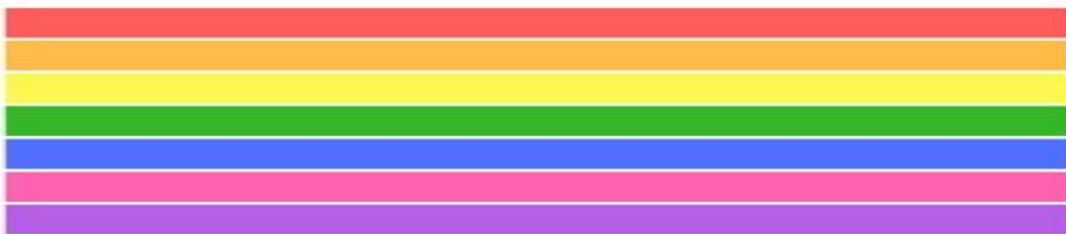


Exercise

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

`` element is used as an inline element and a `<div>` element as a block level element. Basically, an inline element does not cause a line break (start on a new line) and does not take up the full width of a page, only the space bounded by its opening and closing tag. It is usually used within other HTML elements. A block-level element always starts on a new line and takes up the full width of a page, from left to right. A block-level element can take up one line or multiple lines and has a line break before and after the element.

BLOCK-LEVEL ELEMENTS:



INLINE ELEMENTS:



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

`display:none` removes the element from the normal flow of the page, allowing other elements to fill in.

`visibility:hidden` leaves the element in the normal flow of the page such that it still occupies space.

3. Explain the clear and float properties.

The float property is used for positioning and formatting content e.g. let an image float left to the text in a container. The float property can be used to wrap text around images. The clear property specifies what elements can float beside the cleared element and on which side.

Usage:

```
img {  
  float: none;  
}
```

```
div {  
  clear: left;  
}
```

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

These are the position property specifies the type of positioning method used for an element.

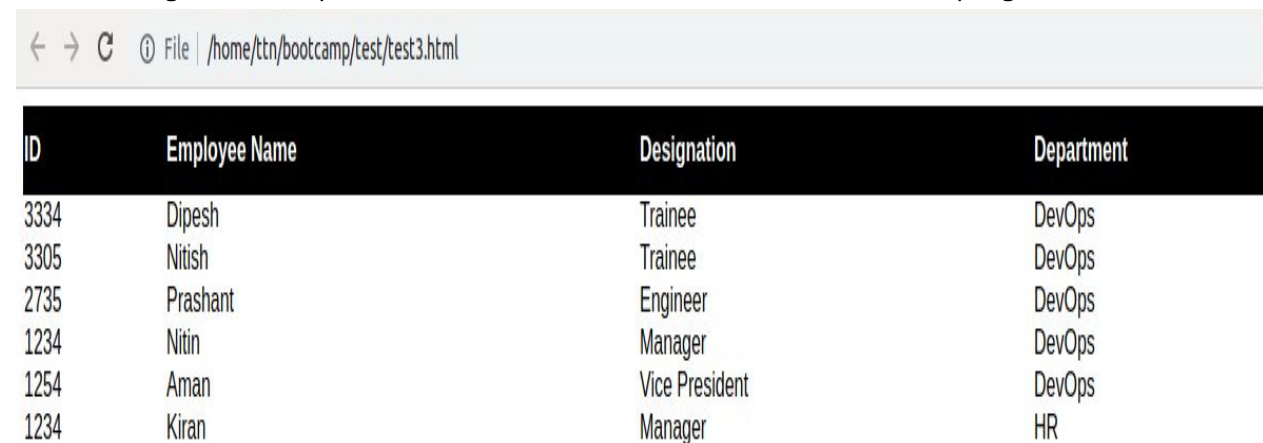
Relative: Making an HTML element relative, gives you the privilege to move the element from its current position. It does not refer to a different element's position.

Absolute: Absolute positioning an HTML element positions the element to its nearest positioned parent. Thus it refers to the nearest parent's position.

Fixed: An HTML element positioned fixed is relative to the viewport and not to any other element.

Static: Static positioned elements are not affected by the top, bottom, left, and right properties. An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page.

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
3334	Dipesh	Trainee	DevOps
3305	Nitish	Trainee	DevOps
2735	Prashant	Engineer	DevOps
1234	Nitin	Manager	DevOps
1254	Aman	Vice President	DevOps
1234	Kiran	Manager	HR

6. Why do we use meta tags?

The <meta> tag provides metadata about the HTML document. 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.

A meta tag can have following attributes :

Examples-Define keywords for search engines:

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

Define a description of your web page:

```
<meta name="description" content="Free Web tutorials on HTML and CSS">
```

7. Explain box model.

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, borders, padding, and the actual content.

The **content area**, bounded by the content edge, contains the "real" content of the element, such as text, an image, or a video player. Its dimensions are the content width (or content-box width) and the content height (or content-box height). It often has a background color or background image.

The **padding area**, bounded by the padding edge, extends the content area to include the element's padding. Its dimensions are the padding-box width and the padding-box height.

The **border area**, bounded by the border edge, extends the padding area to include the element's borders. Its dimensions are the border-box width and the border-box height.

The **margin area**, bounded by the margin edge, extends the border area to include an empty area used to separate the element from its neighbors. Its dimensions are the margin-box width and the margin-box height.

8. What are the different types of CSS Selectors?

In CSS, selectors are patterns used to select the element(s) you want to style.

The **universal selector** works like a wild card character, selecting all elements on a page. Every HTML page is built on content placed within HTML tags. Each set of tags represents an element on the page.

```
* {  
  color: green;}
```

Element type selector or "type selector," this selector must match one or more HTML elements of the same name. Thus, a selector of nav would match all HTML nav elements, and a selector of would match all HTML unordered lists, or elements.

An **ID selector** is declared using a hash, or pound symbol (#) preceding a string of characters. The string of characters is defined by the developer. This selector matches any HTML element that has an ID attribute with the same value as that of the selector, but minus the hash symbol.

The **class selector** is the most useful of all CSS selectors. It's declared with a dot preceding a string of one or more characters. Just as is the case with an ID selector, this string of characters is defined by the developer. The class selector also matches all elements on the page that have their class attribute set to the same value as the class, minus the dot.

```
.box {  
    padding: 20px;  
}
```

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.

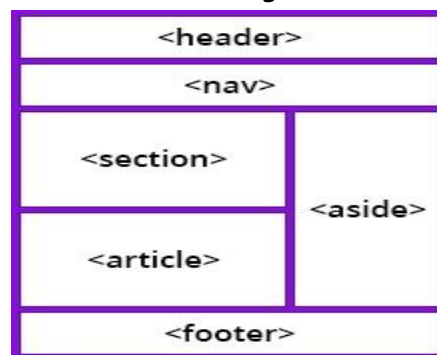
In HTML 4.01, the `<!DOCTYPE>` declaration refers to a DTD, because HTML 4.01 was based on SGML. The DTD specifies the rules for the markup language, so that the browsers render the content correctly.

HTML5 is not based on SGML, and therefore does not require a reference to a DTD.

Usage Example- `<!DOCTYPE html>`

10. Explain 5 HTML5 semantic tags.

A semantic element clearly describes its meaning to both the browser and the developer.



`<article>`

The `<article>` element is one of the major HTML5 semantic tags. It is used to define the article content on your website. It is usually used for big parts of the text.

`<aside>`

The `<aside>` semantic element defines the content which will be set to the side. It is occasionally used for creating sidebars but can also be used for less important content sharing.

<details>

This is one of the HTML5 semantic tags that defines the details on your website. The details can either be visible to the audience or hidden.

<figcaption>

The <figcaption> tag generates a title for <figure> element. It always goes together with <figure> element.

<footer>

The HTML5 element <footer> describes the footnote for your website or part of the content.

11. Create HTML for web-page.jpg

File uploaded

12. Create HTML for form.png

File uploaded