

Hema Coding School

Wednesday, January 31, 2024

Day 1 || Introduction of Html || Element || Structure of HTML || Full stack web development

HTML Topics

Day 1 :

1. Introduction of Html:

Who has invented HTML?

- Sir Tim Berners-Lee

<https://home.web.cern.ch/science/computing/birth-web>

Growth of HTML:

Version	Year	Tags
HTML 1.0	1993	H1 to h6, <p>, , list, img, etc
HTML 2.0	1995	Frames, Form elements, meta, etc
HTML 3.2.0	1997	Tables, acronym, etc
HTML 4.0	1997	CSS, header, footer, label etc.
XHTML 1.0	2000	Self closing tags, xml etc
HTML 5.0	2008	Doctype, semantic, media etc

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

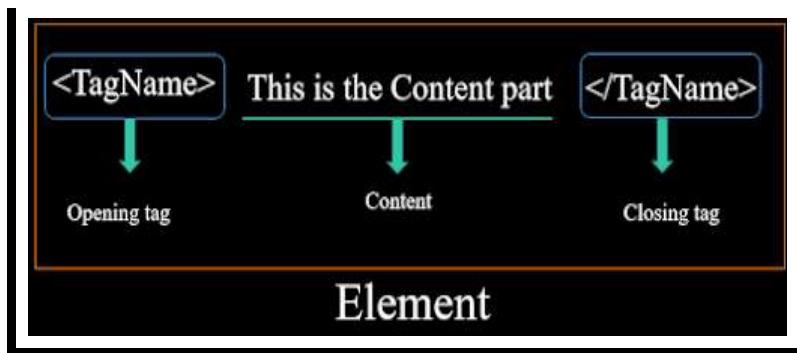
[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

2. Element:



Nest
ed
tags
:

```
<p> Welcome
<h4> Hema
<h1>Coding School</h1>
```

```
</h4>
</p>
```

3. Structure of HTML :

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1>Hema coding school</h1>

</body>
</html>
```

Task:

1. Complete VS code installation

Interview Questions:

1. What is the purpose of the <!DOCTYPE html> declaration in an HTML document?
2. Describe the purpose of the <meta charset="UTF-8"> tag in an HTML document?
3. Explain the importance of the opening and closing tags in HTML?

at January 31, 2024

Hema Coding School

Tuesday, December 5, 2023

Day 2 || Text Formatting tags in Html ||
List tags || Link tags|| Image tags || Full
stack web development in telugu

HTML Topics

Day 2 :

- Text Formatting
- List
- Link
- Image
- Table
- Form

Text Formatting :

Heading tags:		Px = Pixel em = 16px
<h1>	main title or headline of a page or article</h1>	32 px / 2 em
<h2>	section titles or major headings within the content</h2>	24 px / 1.5 em
<h3>	used as subheadings within <h2> sections</h3>	18.72 px / 1.17 em
<h4>	further levels of subsections or subheadings</h4>	16 px / 1 em
<h5>	further levels of subsections/subheadings </h5>	13.28 px / 0.83 em
<h6>	further levels of subsections or subheadings </h6>	12 px / 0.75 em

Search This Blog

Search

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

<hr>: Horizontal rule (line).

<p>: Paragraph.

****: Bold text.

<i>: Italicized text.

<u>: Underlined text.

**
**: Line break.

list, Link
and Image :

List:

****: Unordered list.

****: List item.

****: Ordered list.

Link:

<a>: Anchor (used for hyperlinks).

href stands for “ Hypertext Reference ”

Image:

Self closing tag:

< tagName / >

****:Image.

Table:

table: Creates a table

<table >

tr: Table row

<tr>

<th>Header </th>

</tr>

<tr>

<td>Data </td>

</tr>

</table>

Form:

form: Creates an HTML form.

<form >

Label: Naming of the field

<label >Name:</label>

input: Input field

<input type="text" >

button: Button.

<label >Age:</label>
<input type="text" >

```
<input type="text" >
<input type="submit" >
</form>
```

Task:

1. By using text formatting, list, link and image tags create below document?

The Magical Bottle

Once upon a time, in a faraway land, there was a mysterious magical bottle. Legends spoke of its ~~mysterious powers~~ and the wonders it held within. People from all around the kingdom heard of the bottle's ~~mysterious~~ abilities. It was said that the bottle had the ability to ~~absorb~~ or ~~release~~ anything placed inside it. Stories even mentioned ~~time-traveling~~ and hidden within.

Discovery of the Bottle

As Elena reached the base of the forest, she stumbled upon a hidden ~~plain~~ ~~case~~. Inside this case, she discovered the long-lost ~~magical bottle~~ resting on a pedestal.



The Wonders Unveiled

With a mere thought, Elena realized that [the plain case](#) could transport her to any period she desired. She marvelled at the opportunity to witness history firsthand.

The Journey Continues

- Visiting the first city of Athens.
- Attending Shakespearean theater performances.
- Joining the ~~magical~~ ~~most~~ grand ~~magical~~ tableaus.

Elena's adventure had just begun. Armed with the magical bottle, she set off on a [time-travelling quest](#), exploring ancient civilisations, witnessing legendary events, and learning the stories of the past.

2. Create below table using table tags ?

Title	Author	Genre	Publication Year
The Great Gatsby	F. Scott Fitzgerald	Classic	1925
To Kill a Mockingbird	Harper Lee	Drama	1960
1984	George Orwell	Dystopian	1949

3. Create feedback form using form tags?

Name:

Email:

Rating: Excellent Good Fair

Comments:

Interview Questions:

How can you create a hyperlink in HTML?

What is the difference between an ordered list (``) and an unordered list (``) in HTML?

Describe the purpose of the `<sub>` and `<sup>` tags in HTML

at December 05, 2023

Day 3 || Introduction of CSS || Types of Styles || CSS selectors and properties

CSS Topics

Day 3 :

- Introduction of CSS
- Types of Styles
- CSS selectors and properties

Types of Styles:

Inline Styles:

```
<p style = " color :orange; " >Text</p>
```

The diagram illustrates the structure of inline styles. It shows a code snippet: <p style = " color :orange; " >Text</p>. A vertical green arrow points upwards from the word 'style' in the attribute to the word 'color' in the value, with the word 'value' written above it. Another vertical green arrow points downwards from the attribute to the colon, with the word 'property' written below it.

Internal Styles:

```
<head>      <style>
              Selector {
                  Property: Value;
              }
              </style>
            </head>

            <body>
            <p type of selector : selector name>text</p>
            </body>
```

Search This Blog

[Home](#)
[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)
[March 2024 \(6\)](#)
[January 2024 \(9\)](#)
[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)
[View my complete profile](#)

External Styles: <head>

```
<link rel="stylesheet" type="text/css" href="style.css">  
</head>
```

Style.css

```
h1 {  
    color: blue;  
}
```

C
S
S

selectors and properties:

ID Selector (#id):

```
#header {  
    color : green;  
}
```

```
<p id = "header " >Text</p>
```

Class Selector (.classname):

```
.highlight {  
    color: yellow;  
}
```

```
<p class = " highlight" >Text</p>
```

Type Selector (Element Selector):

```
h1 {  
    color: blue;  
}
```

```
<h1>text</h1>
```

```
Child Selector (parent > child): ul > li {
    list-style-type: square;
}
```

```
Attribute Selector ([attribute=value]) input[type="text"] {
    border: 1px solid #ccc;
}
```

```
Pseudo-classes (:pseudo-class) a:hover {
    color: red;
}
```

Task:

Please add color in below image (this is continuation of Day2 task)

The Magical Bottle

Once upon a time, in a faraway land, there was a mysterious magical bottle. Legends spoke of its ~~mysterious powers~~ and the wonders it held within.

People from all around the kingdom heard of the bottle's ~~mysterious powers~~. It was said that the bottle had the ability to ~~magically~~ anything placed inside it. Before even ~~mentioning~~ their ~~desires~~ and ~~wishes~~ within.

Discovery of the Bottle

As Elma reached the heart of the forest, she stumbled upon a hidden glowing ~~area~~. Hurriedly, she discovered the long-lost magical bottle resting on a pedestal.

[Illustration of the magical bottle](#)

Elma ~~curiously~~ examined the bottle. Its surface was adorned with intricate patterns, and a faint glow emanated from within. She couldn't resist the temptation to open it.

The Wonders Unveiled

As the bottle's cap was lifted, a burst of ~~golden light~~ filled the room. To her amazement, Elma found herself surrounded by floating books containing stories from different eras.

The Journey Continues

Elma's adventure had just begun. Armed with the magical bottle, she set off on a ~~true~~ ~~adventure~~, exploring ancient civilizations, witnessing legendary events, and learning the stories of the past.

- Visiting the lost city of ~~Atlantis~~
- Attending Shakespearean theater performances
- Witnessing the ~~renaissance~~'s great scientific breakthroughs

Interview Questions:

- When would you prefer using inline styles over other styling methods?
- How do you define internal styles within an HTML document?
- How do you link an external CSS file to an HTML document?
- Differentiate between element, class, and ID selectors.

Hema Coding School

Friday, December 8, 2023

Day 4 || Styling text in CSS || Box Model ||
Block-level Elements || In-line || Elements
Float

CSS Topics

Day 4 :

- Styling text
- Box Model
- Block-level Elements
- In-line Elements
- Float

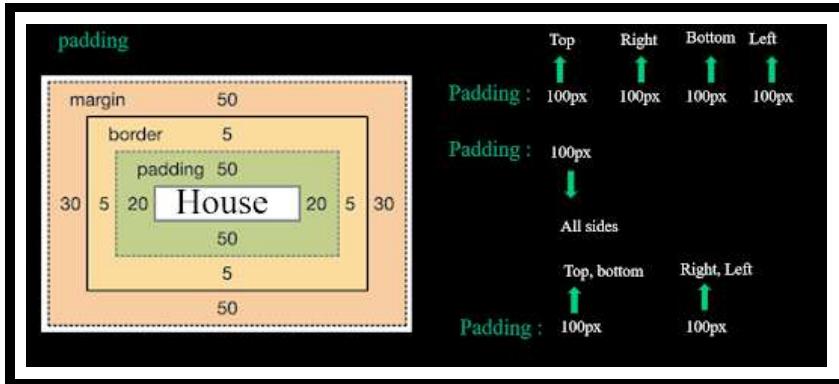
Styling text :

Font Properties `font-family, font-size, font-weight ...etc`

Text Alignment: `text-align`

Text Decoration: `background-color, color`

Box Model:



Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

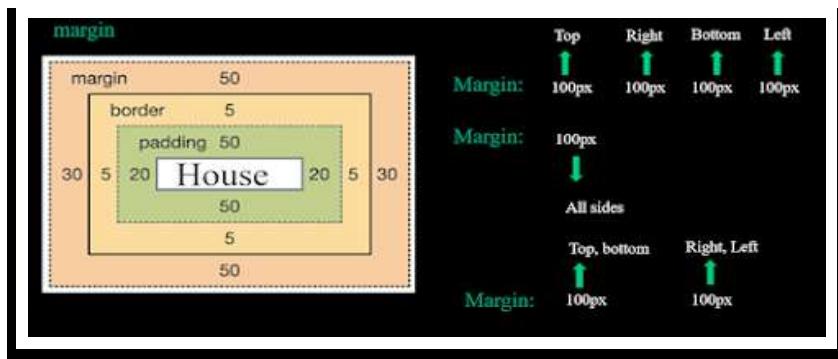
[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
<style>
p{
    font-family: Arial, Helvetica, sans-serif;
    font-size: 30px;
    font-weight: 600;
    text-align: center;
    background-color: orange;
    color: white;
    padding: 50px;
    border: 10px solid black;
}
</style>
</head>
<body>
    <p>Hema coding School</p>
</body>
</html>
```

Block-level Elements:

Common block-level elements include `<div>`, `<p>`, `<h1>` to `<h6>` (headings), `` (unordered list), `` (ordered list), `` (list item), `<table>`, `<form>`, and more.

In-line Elements:

Common inline elements include ``, `<a>` (anchor), `` (strong/bold), `` (image), `
` (line break), `<code>` (code), and more.

Float:

```
selector {
    float: right or left ;
}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
<style>
p {
    display: inline;
```

```

}
span {
  display: block;
}
img {
  float: right;
}
</style>
</head>
<body>
<p style="background-color: blue">Welcome to Hema coding School</p>
<p style="background-color: blue">Hema coding School</p>

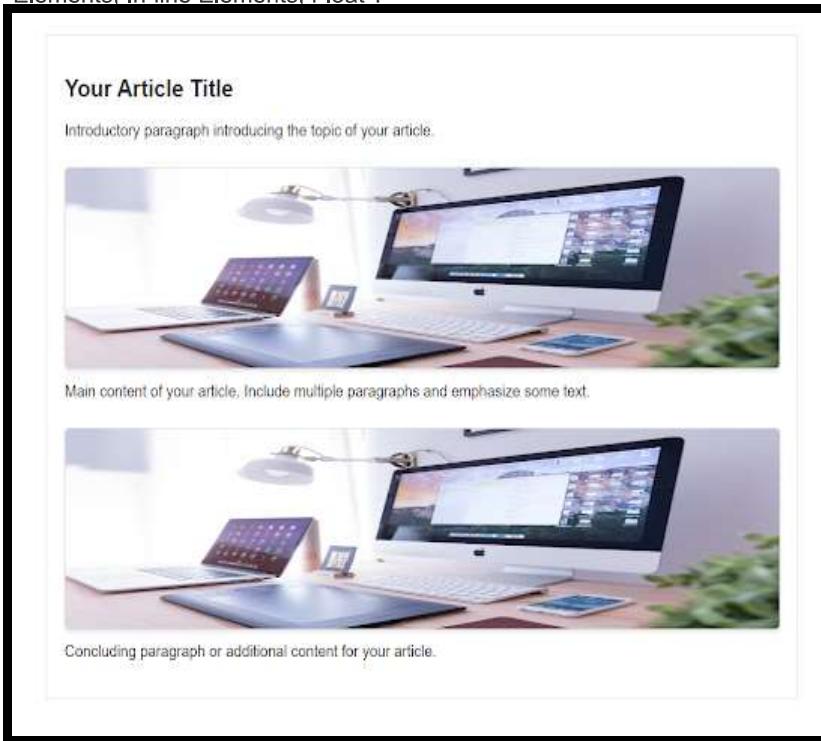
<span style="background-color: red">CSS</span>


</body>
</html>

```

Task:

Create article page as like below using Styling text, Box Model, Block-level Elements, In-line Elements, Float ?



Interview Questions:

Hema Coding School

Sunday, December 10, 2023

Day 5 || Media Queries in CSS

CSS Topics

Day 5 :

- Added CSS to HTML Page
- Media Queries

HTML Page with CSS:

newPage.css:

```
body {
  background-color: #d1cece;
  text-align: center;
  font-family: "Arial", sans-serif;
}
h1 {
  color: #0066cc;
}
table {
  margin: auto;
  margin-top: 10px;
  border-collapse: collapse;
  width: 70%;
}
th,
td {
  padding: 10px;
}
marquee {
  color: red;
}
form {
  float: right;
  background-color: white;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 2px 2px 5px 0px #888888;
}
label {
  display: block;
  text-align: left;
}
input {
  width: 100%;
  padding: 8px;
  border-radius: 5px;
  border: 1px solid #ccc;
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

Hema Coding School

[View my complete profile](#)

```
}

input[type="submit"] {
  background-color: #0066cc;
  cursor: pointer;
  color: #fff;
}

input[type="submit"]:hover {
  background-color: #004080;
}

.container {
  width: 1000px;
  margin: auto;
}

img {
  border-radius: 10px;
  box-shadow: 2px 2px 5px 0px #888888;
}

@media screen and (min-width: 0px) and (max-width: 767px) {
  .container {
    width: auto;
    margin: auto;
  }

  img {
    max-width: 100%;
    margin-bottom: 20px;
  }

  form {
    float: none;
  }
}

@media screen and (min-width: 768px) and (max-width: 1023px) {
  .container {
    width: auto;
    margin: auto;
  }

  img {
    width: auto;
    margin-bottom: 20px;
  }

  form {
    float: none;
    margin: 0px 100px;
  }
}

@media screen and (min-width: 1024px) and (max-width: 1365px) {
  .container {
    margin: auto;
  }

  img {
    border-radius: 10px;
    box-shadow: 2px 2px 5px #888888;
    height: auto;
  }

  form {
    float: right;
    background-color: white;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 2px 2px 5px #888888;
  }
}

@media screen and (min-width: 1366px) {
  .container {
```

```

        margin: auto;
    }
    img {
        border-radius: 10px;
        box-shadow: 2px 2px 5px #888888;
        height: auto;
    }

    form {
        float: right;
        background-color: white;
        padding: 20px;
        border-radius: 10px;
        box-shadow: 2px 2px 5px #888888;
    }
}

```

newPage.html

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>Document</title>
        <link rel="stylesheet" type="text/css" href="./newPage.css" />
    </head>
    <body>
        <h1>Hema Coding School</h1>
        <p>
            Welcome to Hema Coding School channel. We can teach
            <a href=".//textFormatting.html">full stack web development.</a>
        </p>
        <marquee>
            We will cover in this course HTML, CSS, JavaScript, React, Node.js, and
            MongoDB</marquee>
        </>
        <div class="container">
            
            <form>
                <label>Name:</label>
                <input type="text" />
                <label>Email:</label>
                <input type="text" />
                <label>Age:</label>
                <input type="text" />
                <input type="submit" />
            </form>
        </div>
        <table border="1">
            <tr>
                <th>HTML</th>
                <th>CSS</th>
                <th>JavaScript 3</th>
            </tr>
            <tr>
                <td>Basics of html</td>
                <td>Basics of CSS</td>
                <td>Basics of JavaScript</td>
            </tr>
            <tr>
                <td>HTML 5</td>
                <td>CSS 3</td>

```

```

<td>ES 6</td>
</tr>
</table>
</body>
</html>

```

Media Queries:

Mobile Devices (up to 767px): @media screen and (min-width: 768px) and (max-width: 767px) {}

Tablet Devices (768px to 1023px): @media screen and (min-width: 768px) and (max-width: 1023px) {}

Laptop Devices (1024px to 1365px): @media screen and (min-width: 1024px) and (max-width: 1365px) {}

Desktop Devices (1366px and above): @media screen (min-width: 1366px) {}

Interview Questions:

1. How do you write a basic media query for screens with a maximum width of 768 pixels?
2. What are some common media features used in media queries?
3. What is the purpose of media queries in creating responsive web designs?

at December 10, 2023

No comments:**Post a Comment**

SIGN IN WITH GOOGLE

[Newer Post](#)[Home](#)[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Top 10 | JavaScript Coding Interview Question | Beginner Level

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console

[Day 1 || Introduction of Html || Element || Structure of HTML || Full stack web development](#)

HTML Topics Day 1 : 1. Introduction of Html: Who has invented HTML? - Sir Tim Berners-Lee https://home.web...

Hema Coding School

Monday, December 11, 2023

Day 6 || Semantic HTML elements || Form Elements

CSS Topics

Day 6 :

- Semantic HTML elements
- Form Elements

Semantic HTML elements:

Header:	<code><header> </header></code>	Aside:	<code><aside> </aside></code>
Nav:	<code><nav> </nav></code>	Footer:	<code><footer> </footer></code>
Main:	<code><main> </main></code>		
Article:	<code><article> </article></code>		
Section:	<code><section> </section></code>		

Home.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <link rel="stylesheet" type="text/css" href="./Home.css" />
  </head>
  <body>
    <header>
      <h1>Hema Coding School</h1>
    </header>
    <nav>
      <a href="./Home.html">Home</a>
      <a href="./Courses.html">Courses</a>
      <a href="/About.html">About Us</a>
      <a href="./Contact.html">Contact</a>
    </nav>
  </body>
</html>
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

<main>
  <section>
    <h2>Welcome to Hema Coding School!</h2>
    <p>Empowering your journey in the world of coding and programming.</p>
  </section>
  <section>
    <h2>Featured Courses</h2>
    <p>
      Explore our diverse range of coding courses designed to meet your
      learning needs.
    </p>
  </section>
  <article>
    <h2>Featured Article</h2>
    <p>This is a featured article providing valuable insights into the coding
      industry.</p>
  </article>
  <aside>
    <h2>Announcements</h2>
    <p>
      Stay updated with the latest news and announcements from Hema Coding
      School.
    </p>
  </aside>
</main>
<footer>
  <p>&copy; 2024 Hema Coding School. All rights reserved.</p>
</footer>
</body>
</html>

```

Home.css

```

body {
  font-family: Arial, sans-serif;
}

header {
  background-color: #333;
  color: #fff;
  text-align: center;
  padding: 16px;
}

nav {
  background-color: #444;
  padding: 8px;
}

nav a {
  color: #fff;
  margin: 0 16px;
  padding: 8px;
  text-decoration: none;
}

footer {
  background-color: #333;
  color: #fff;
  text-align: center;
  padding: 16px;
}

main {
  padding: 16px;
}

section {
  margin-bottom: 16px;
}

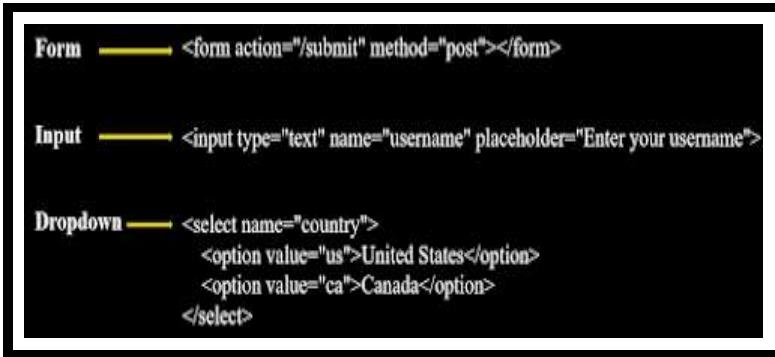
```

```

}
aside {
  padding: 16px;
}
article{
  padding: 16px;
}

```

Form Elements:



Contact.html

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <link rel="stylesheet" type="text/css" href=".//Home.css" />
    <link rel="stylesheet" type="text/css" href=".//Contact.css" />
  </head>
  <body>
    <header>
      <h1>Hema Coding School</h1>
    </header>
    <nav>
      <a href=".//Home.html">Home</a>
      <a href=".//Courses.html">Courses</a>
      <a href=".//About.html">About Us</a>
      <a href=".//Contact.html">Contact</a>
    </nav>
    <main>
      <form action="/submit">
        <label>Username:</label>
        <input type="text" name="username" placeholder="Enter your username" />

        <label>Password:</label>
        <input
          type="password"
          name="password"
          placeholder="Enter your password"
        />
        <label>Email:</label>
        <input type="email" name="useremail" placeholder="Enter your email" />

        <label>Country:</label>
        <select>
          <option>India</option>
          <option>Canada</option>
        </select>
      </form>
    </main>
  </body>
</html>

```

```

        <option>United Kingdom</option>
    </select>

    <label>Number:</label>
    <input type="number" min="1" max="10" />
    <label>Birthdate:</label>
    <input type="date" name="birthdate" />

    <label>Upload File:</label>
    <input type="file" name="file-upload" />

    <button type="submit">Submit</button>
</form>
</main>
</body>
</html>

```

Contact.css

```

label {
    display: block;
    margin-bottom: 8px;
    font-weight: bold;
    color: #d52929;
}

form {
    max-width: 400px;
    margin: 20px auto;
    padding: 20px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    border-radius: 8px;
}
input,
select {
    width: 90%;
    padding: 10px;
    margin-bottom: 16px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
button {
    background-color: #007bff;
    color: #fff;
    padding: 10px 20px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}
button:hover {
    background-color: #0056b3;
}

```

Interview Question:

1. What are semantic HTML tags, and why are they important?
2. How does the <nav> tag differ from a regular <div>?
3. Explain the purpose of the required attribute in form elements ?

Hema Coding School

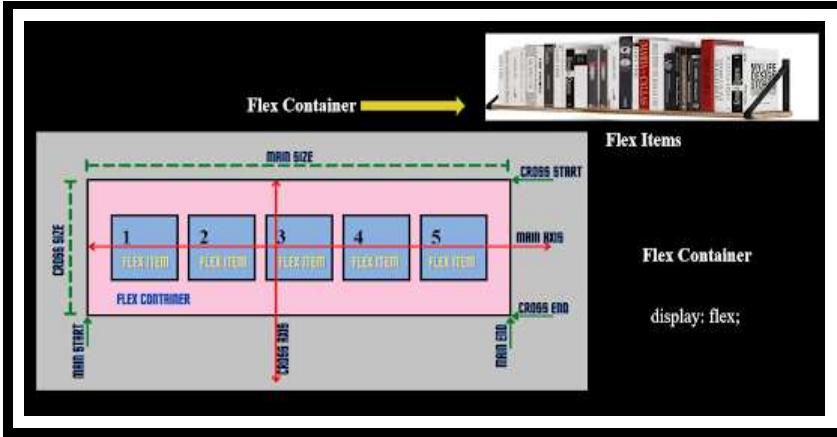
Tuesday, December 12, 2023

Day 7 || Flex box in CSS || Flexible Box Layout in CSS

CSS Topics

Day 7 :

Flexible Box Layout:



Flex.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <style>
      .container {
        display: flex;
        border: 1px solid black;
        height: 500px;
        /* flex-direction: row; */
        /* flex-direction: column; */
        /* flex-direction: row-reverse; */
        /* flex-direction: column-reverse; */
        /* justify-content: flex-start; */
        /* justify-content: flex-end; */
        /* justify-content: center; */
        /* justify-content: space-around; */
        /* justify-content: space-between; */
        align-items: flex-start;
        /* align-items: flex-end; */
        /* align-items: center; */
        /* align-items: baseline; */
        /* align-items: stretch; */
      }
    </style>
  </head>
  <body>
    <div class="container"></div>
  </body>
</html>
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

        /* flex-wrap: wrap; */
        /* flex-wrap: wrap-reverse; */
    }
    .item {
        background-color: aqua;
        margin: 5px;
        padding: 40px;
    }
    .item3 {
        align-self: center;
    }

</style>
</head>
<body>
    <div class="container">
        <div class="item item1">Book1</div>
        <div class="item item2">Book2</div>
        <div class="item item3">Book3</div>
        <div class="item item4">Book4</div>
        <div class="item item5">Book5</div>
    </div>
</body>
</html>

```

Courses.html

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>Document</title>
        <link rel="stylesheet" type="text/css" href="./Home.css" />
        <link rel="stylesheet" type="text/css" href="./Courses.css" />
    </head>
    <body>
        <header>
            <h1>Hema Coding School</h1>
        </header>
        <nav>
            <a href="./Home.html">Home</a>
            <a href="./Courses.html">Courses</a>
            <a href="/About.html">About Us</a>
            <a href="./Contact.html">Contact</a>
        </nav>
        <main class="courses-container">
            <section class="course-card">
                <h2>HTML</h2>
                <p>
                    Learn the fundamentals of HTML for building the structure of web pages.
                </p>
            </section>
            <section class="course-card">
                <h2>CSS</h2>
                <p>
                    Explore the styling and layout capabilities of CSS to enhance your web designs.
                </p>
            </section>
            <section class="course-card">
                <h2>JavaScript</h2>
                <p>
                    Master the scripting language for dynamic and interactive web development.
                </p>
            </section>
            <section class="course-card">

```

```
<h2>React</h2>
<p>
  Build modern, efficient, and scalable user interfaces with the React
  library.
</p>
</section>
<section class="course-card">
  <h2>Node.js</h2>
  <p>
    Learn server-side JavaScript with Node.js for building scalable
    network applications.
  </p>
</section>
<section class="course-card">
  <h2>MongoDB</h2>
  <p>
    Explore the NoSQL database for building scalable and flexible
    applications.
  </p>
</section>
</main>
<footer>
  <p>&copy; 2024 Hema Coding School. All rights reserved.</p>
</footer>
</body>
</html>
```

Courses.css

```
h2{
  color: blue;
}

.courses-container {
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;

}

.course-card {
  background-color: #f0f0f0;
  border: 1px solid #ccc;
  border-radius: 8px;
  margin: 20px;
  padding: 15px;
  text-align: center;
  flex: 30%;

}
```

Interview Questions:

1. What is Flexbox, and what problem does it solve in web development?
2. Explain the difference between the main axis and the cross axis in Flexbox.
3. What is the purpose of the align-self property in Flexbox?

Hema Coding School

Thursday, December 14, 2023

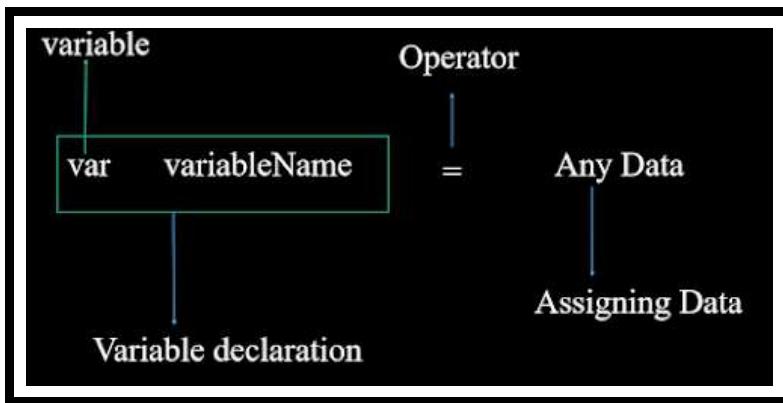
Day 8 || Introduction of JavaScript || Variable || Data types || Operators

JavaScript Topics

Day 8 :

- Introduction
- Variable
- Data Types
- Operators

Variable :



```
var bag = 'Hello world';
console.log(bag)
```

Data Types:

Search This Blog

[Home](#)

[Report Abuse](#)

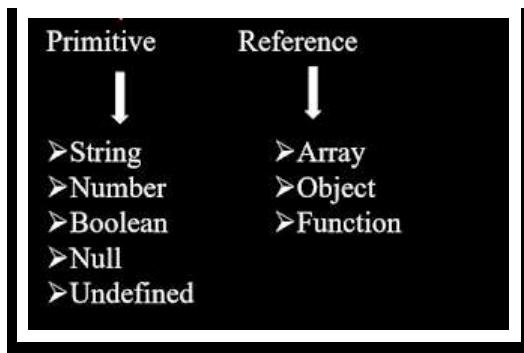
Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)



```

var name = "Mahesh";
var age = 25;
var isMarried = false;
var love = null;
var address;
console.log(name)
console.log(age)
console.log(isMarried)
console.log(love)
console.log(address)
  
```

Operators:

Arithmetic Operators:	Comparison Operators:
Addition +	Equal to ==
Subtraction -	Not equal to !=
Multiplication *	Greater than >
Division /	Less than <
	Greater than or equal to >=
	Less than or equal to <=

```

// Product: iPhone
// var iPhonePrice = 150000;
// var discountPercentage = 25;
// var discountAmount =(iPhonePrice * discountPercentage)/100;
// console.log("Discount Amount: " + discountAmount + " Rupees");
var a = 5;
var b = 6;
// console.log(a == b);
// console.log(a === b);
// console.log(a != b)
// console.log(a < b)
console.log(a > b);
  
```

Task:

- Logical Operators**
- Conditional (Ternary) Operator**
- Bitwise Operators**
- Unary Operators**
- String Operators**
- Type Conversion Operators**

Interview Questions:

1. What is the difference between null and undefined in JavaScript?
2. What is the significance of the typeof operator?
3. Explain the difference between == and === in JavaScript?

at December 14, 2023

No comments:

Post a Comment

SIGN IN WITH GOOGLE

Older Post

Newer Post

Home

Subscribe to: [Post Comments \(Atom\)](#)

[Top 10 | JavaScript Coding Interview Question | Beginner Level](#)

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console

Hema Coding School

Friday, December 15, 2023

Day 9 || if Statement || else if Statement || Switch in JavaScript

JavaScript Topics

Day 9 :

Conditional statements:

1. if Statement
2. else Statement
3. else if Statement
4. Switch

Conditional statements:

1. if Statement and 2. else Statement:

```
var userAge = 10;
if (userAge >= 18) {
  console.log("Welcome! You can open the website.");
} else {
  console.log("Sorry, you are not old enough to open the website.");
}
// Ticket booking
var totalSeatsInTheater = 100;
var bookedSeatsInTheater = 70;
var availableSeatsInTheater = totalSeatsInTheater - bookedSeatsInTheater;
if (availableSeatsInTheater > 0) {
  console.log("We have " + availableSeatsInTheater + " seats available.");
  var numOfTicketsBooking = 2;
  if (numOfTicketsBooking <= availableSeatsInTheater) {
    console.log("Booking " + numOfTicketsBooking + " tickets...");
  } else {
    console.log("Sorry, there are not enough seats available.");
  }
} else {
  console.log("Sorry, the theater is fully booked.");
}
```

3. else if Statement:

```
var examScore = 45;
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

if (examScore >= 90) {
    grade = "A";
} else if (examScore >= 80) {
    grade = "B";
} else if (examScore >= 70) {
    grade = "C";
} else if (examScore >= 50) {
    grade = "D";
} else {
    grade = "F";
}
console.log("Your grade is:" + grade);

```

4. Switch

```

var teaOrder = "jasmineTea";

switch (teaOrder) {
    case "Chai":
        message = "Chai ordered";
        break;
    case "jasmineTea":
        message = "jasmineTea ordered";
        break;
    default:
        message = "Not Available"
}
console.log(message);

```

Task 1: Check if a Number is Even or Odd

Write a JavaScript program that takes a number as input and displays whether it's even or odd.

Task 2: Determine the Greater Number

Write a JavaScript program that takes two numbers as input and displays which one is greater (or if they are equal).

Task 3: Check if a Number is Positive, Negative, or Zero

Write a JavaScript program that takes a number as input and displays whether it's positive, negative, or zero.

Interview Questions:

1. What is the difference between switch and if...else if...else when it comes to handling multiple conditions?
2. What is the purpose of the break statement in a switch statement?
3. How do you use the else if statement in JavaScript? Provide a code example.

at December 15, 2023

Hema Coding School

Saturday, December 16, 2023

Day 10 || Array in JavaScript || Object in JavaScript

JavaScript Topics

Day 10 :

- Array
- Object

Array:

```
var sweetBox = ["Mango Crunch", "Oreo Crunch", "Kesar Crunch", "Orange Apricot Cru   Hema Coding School
sweetBox.push("Rose Kaju Crunch")
sweetBox.unshift("Cherries Crunch")
sweetBox.pop()
sweetBox.shift()
array.splice(index, elements to be removed, elements to be added)
sweetBox.splice(0,1,"Lemon Crunch", "Kiwi Crunch")
console.log(sweetBox)

var sweetBox = ["Mango Crunch", "Oreo Crunch", "Kesar Crunch", "Orange Apricot Crunch",123,143,133, true, false];
```

Object:

```
var student = {
  name:"Hema",
  age:21,
  isStudent:true,
  studentAge:23
}

student.course = "FSWD"
student.age = 22
delete student.isStudent;

student["course"] = "FSWD";
student["age"] = 23;
delete student["isStudent"];

var studentAge = "age";
console.log(student.studentAge)
console.log(student[studentAge])
```

Tasks:

Array Task: Shopping List

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

Create a JavaScript program that simulates a simple shopping list. Use an array to represent the items in the list.

Your program should perform :

Create an Array:

Create an array called shoppingList with some initial items.

Display Shopping List:

Write a function that takes the shopping list array as a parameter and displays the list of items.

Add Item:

Write a function that takes the shopping list array and a new item as parameters. Add the new item to the shopping list.

Remove Item:

Write a function that takes the shopping list array and an item to remove as parameters. Remove the specified item from the shopping list.

Object Task: Student Information

Create a JavaScript program that manages information about students. Each student should have the following properties:

Name

Age

Grade Level

Subjects (an array of subjects the student is studying)

Your program should perform :

Create an Object:

Create an object called student with properties for a sample student.

Display Student Information:

Write a function that takes a student object as a parameter and displays its information (name, age, grade level, and subjects).

Update Property:

Write a function that takes a student object, a property name, and a new value as parameters. Update the specified property of the student with the new value.

Add Subject:

Write a function that takes a student object and a new subject as parameters. Add the new subject to the student's list of subjects.

Interview Questions:

Hema Coding School

Sunday, December 17, 2023

Day 11 || Function in JavaScript || Object Methods in JavaScript

JavaScript Topics

Day 11 :

- Function
- Object Method

Function:

```
function addition(x,y){
    return x + y;
}
console.log(addition(2,5))

var addition = function(x,y){
    return x + y
}
console.log(addition(5,5))

var juiceMixer = function(fruit1,fruit2){
    return "Delicious " + fruit1 + " and " + fruit2 + " juice is ready"
}

console.log(juiceMixer("mango","strawberry"))
console.log(juiceMixer("apple","orange"))
console.log(juiceMixer("blueberry","orange"))
```

Object Method:

```
var magicalToolBox = {
    compartments:{
        screws:["wood screws", "sheet metal screws", "machine screws"],
        wires: ["red wires", "black wires", "green wires"],
        gears:["small gears - 12 teeth", "medium gears - 24 teeth", "large gears - 36 teeth"]
    },
    openToolBox:function(){
        return "The magical toolbox opens";
    },
    closeToolBox:function(){
        return "The magical toolbox closes"
    }
}
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```
}
```

```
console.log(magicalToolBox.openToolBox())
```

```
console.log(magicalToolBox.closeToolBox())
```

```
console.log(magicalToolBox.compartments.gears)
```

```
console.log(Object.keys(magicalToolBox))
```

```
console.log(Object.values(magicalToolBox))
```

```
console.log(Object.entries(magicalToolBox))
```

Tasks:

Function Task1: Calculate Area of a Rectangle

Create a simple program to calculate the area of a rectangle using a function.

Define a function named calculateArea that takes two parameters, length and width.

Inside the function, calculate the area of the rectangle using the formula area = length * width.

Return the calculated area from the function.

Call the function with specific values for length and width, and print the result to the console.

Object Method Task2: Employee Information Management

Create a program to manage employee information using objects.

Define an object named employee that represents an employee in a company. The employee object should have properties such as name, age, designation, and salary.

Implement a function within the object to update the employee's salary based on a percentage increase.

Create multiple instances of the employee object to represent different employees in the company.

Use the function to update the salary for at least one employee, and then print the updated information for that employee.

Interview Questions:

1. Explain the difference between function declarations and function expressions
2. What is the purpose of the Object.keys(), Object.values(), and Object.entries() methods?
3. What is the purpose of the hasOwnProperty() method?

at December 17, 2023

No comments:

Post a Comment

Hema Coding School

Monday, December 18, 2023

Day 12 || For loop in JavaScript || While Loop in JavaScript || Loop through an Array and Object

JavaScript Topics

Day 12 :

Looping Statement:

- For Loop
- While Loop
- Loop through an Array and Object

For Loop:

```
for (Initialization; condition; Increment)
{
    Execution of code
}
```

```
for(var i = 0; i<=100; i++){
    console.log(i)
}

for(var number = 1; number<=5; number++){
    console.log(number)
}
```

While Loop:

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

Initialization;
while (condition)
{
    Execution of code
    Increment;
}

```

```

Var count = 0;
while (count <= 5)
{
    console.log("Iteration " + count);
    count++;
}

```

```

var count = 0;
while(count<=5){
    console.log(count);
    count++
}

var number = 0;
var sum = 0;
while(number<=10){
    console.log("No: " + number)
    console.log("Sum: " + sum)
    sum = sum + number;
    console.log("sum + number: " + sum)
    number++;
    console.log("inc No: " + number)
}
console.log(sum)

```

Loop through an Array and Object:

Array:

```

var naturalNumber = [10, 20, 30, 40, 50];
for(var number=0; number<naturalNumber.length; number++){
    console.log(naturalNumber[number] + 2)
}

```

Object:

```

var magicalLibrary = {
    book1: {
        name: "Fantasy",
        author: "Eleanor Moon",
        keyPlayer: "Aria",
    },
    book2: {
        name: "Mystery",
        author: "Maxwell Nightingale",
        keyPlayer: "Detective Drake",
    },
    book3: {
        name: "Adventure",
        author: "Isabella Silverwing",
        keyPlayer: "Captain Hawk",
    },
    book4: {
        name: "Adventure",
        author: "Isabella Silverwing",
        keyPlayer: "Captain Hawk",
    }
}

```

```
}

for (var book in magicalLibrary) {
  console.log(magicalLibrary[book].name)
  console.log(magicalLibrary[book].keyPlayer)
}
```

Tasks:

Task 1: Multiplication Table (Using for Loop)

Write a program that takes a number as input and uses a for loop to generate the multiplication table for that number up to 10.

Task 2: Factorial Calculator (Using while Loop)

Write a program that prompts the user to enter a positive integer and uses a while loop to calculate its factorial.

Task 3: Find Even Numbers in an Array

Objective:

Write a program that iterates through an array of numbers and prints only the even numbers.

Instructions:

Create an array of numbers

Use a for loop to iterate through the array.

Print only the even numbers to the console.

Task 4: Movie Database

Objective:

Create a program that represents a simple movie database using an object. The object should contain information about several movies.

Instructions:

Create an object named movieDatabase with at least three movie entries.

Each movie entry should have the following properties:

Title

Director

Release Year

Genre

Rating (out of 10)

Use a for...in loop to iterate through the movieDatabase object.

For each movie, print its title, director, and rating to the console.

Interview Questions:

1. What is the difference between a for loop and a for...in loop?
2. Can you use multiple variables in the initialization, condition, and iteration expressions of a for loop?
3. How does a while loop differ from a for loop?
4. Can a while loop have multiple conditions in its condition expression?



Hema Coding School

Tuesday, December 19, 2023

Day 13 || DOM manipulation in JavaScript || Document Object Model in JavaScript

JavaScript Topics

Day 13 :

- DOM (Document Object Model) manipulation

DOM (Document Object Model) manipulation:

Accessing Elements:	Changing Element Properties
Document.getElementById	textContent
Document.getElementsByClassName	innerHTML
Document.getElementsByTagName	style.color
Document.querySelector	Creating Elements:
Document.querySelectorAll	document.createElement
	appendChild

About.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <link rel="stylesheet" type="text/css" href="./Home.css" />
  </head>
  <body>
    <header>
      <h1>Hema Coding School</h1>
    </header>
    <nav>
      <a href=".//Home.html">Home</a>
      <a href=".//Courses.html">Courses</a>
      <a href=".//About.html">About Us</a>
      <a href=".//Contact.html">Contact</a>
    </nav>
    <main>
      <section>
        <h2 id="about"></h2>
        <p class="aboutParagraph"></p>
      </section>
      <section>
        <h3></h3>
        <p>Our Full Stack Web Development course is designed to equip you with
          the skills needed to become a proficient web developer capable of
        </p>
      </section>
    </main>
  </body>
</html>
```

Search This Blog

Search

Home

Report Abuse

Blog Archive

April 2024 (1)

March 2024 (6)

January 2024 (9)

December 2023 (20)

About Me

Hema Coding School

[View my complete profile](#)

```

        handling both front-end and back-end development.</p>
    <p class="highlight"></p>
    <ul id="courseHighlightsList">
        <li>Comprehensive coverage of HTML, CSS, and JavaScript for building
            interactive and responsive user interfaces.</li>
    </ul>
</section>
</main>
<footer>
    <p>© 2024 Hema Coding School. All rights reserved.</p>
</footer>
<script>
    var subHeading = document.getElementById("about");
    subHeading.textContent = "About : Hema Coding School";

    var paragraphElement = document.getElementsByClassName("aboutParagraph");
    paragraphElement[0].textContent =
        "Hema Coding School is dedicated to providing high-quality coding education. Our experienced instructors are committed to

        var h3Element= document.getElementsByTagName('h3');
        h3Element[0].textContent="Full Stack Web Development Course";

    var highlightElement = document.querySelector('.highlight');
    highlightElement.innerHTML = "<b>Course Highlights:</b>";

    var paragraphStyle = document.querySelectorAll('p');
    for(var i=0; i< paragraphStyle.length; i++){

        paragraphStyle[i].style.color = "red";
    }

    var highlightsList = document.getElementById("courseHighlightsList");
    var listItemsContent = [ "Introduction to Responsive Web Design",
        "Advanced JavaScript Concepts",
        "Server-Side Rendering with Node.js",
        "Database Design and Management",
        "Git and Version Control",
        "Real-world Project Development"
    ];
    // var newListItem = document.createElement('li');
    // newListItem.textContent="Introduction to Responsive Web Design";
    // highlightsList.appendChild(newListItem)

    for(var i =0; i< listItemsContent.length; i++){
        var newListItem = document.createElement('li');
        newListItem.textContent=listItemsContent[i];
        highlightsList.appendChild(newListItem)
    }

</script>
</body>
</html>

```

Task:

Create an HTML file with the following structure:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>DOM Manipulation Task</title>
    <style>
        /* Add any additional styling if needed */
    </style>
</head>
<body>

    <div id="mainContainer">
        <h1>Welcome to DOM Manipulation Task</h1>
        <p id="taskDescription">This is your DOM manipulation task. Follow the instructions below to complete it.</p>

        <div id="taskContainer">
            <!-- Your task content will go here -->
    
```

```
</div>

<button id="submitButton">Submit Task</button>
</div>

<script src="dom-manipulation-task.js"></script>
</body>
</html>
```

1. Create a JavaScript file named dom-manipulation-task.js.

2. Inside the JavaScript file, complete the following tasks:

a. Append three new paragraphs to the taskContainer with the following content:

"Task 1: Use createElement and appendChild to create and add this paragraph."

"Task 2: Change the text content of this paragraph using textContent property."

"Task 3: Add a class 'highlight' to this paragraph using classList."

b. Change the background color of the mainContainer to a light gray color.

c. Attach a click event listener to the submitButton button. When the button is clicked, display an alert message saying "Task completed!".

d. Bonus: Add additional styling to make your page visually appealing.

3. Open the HTML file in a web browser and verify that your DOM manipulation tasks are working as expected.

Interview Questions:

1. How does the DOM differ from HTML?

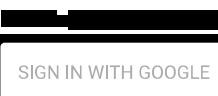
2. How do you select elements in the DOM?

3. What is the difference between innerHTML and textContent?

at December 19, 2023

No comments:

Post a Comment



Newer Post

Home

Older Post

Subscribe to: [Post Comments \(Atom\)](#)

Hema Coding School

Wednesday, December 20, 2023

Day 14 || let and const in JavaScript || var vs let vs const in JavaScript || Template literals in JavaScript

JavaScript Topics

Day 14 :

- let and const
- Template literals

let and const:

```
//Declaration
// 1. Redeclaration
// 2. Reassignment
// 3. scope
// 4. Hoisting

var varSibling = "I might cause neglect and no boundaries";
var varSibling = "I am causing neglect (re-declared)";
varSibling = "I am causing neglect second time (re-assigned/updated)";
console.log(varSibling);

let letSibling = "I can commit to change and having boundaries";
letSibling = "I am re-assigning new value";
let letSibling = "My commitment is one time declaration";
console.log(letSibling);

const constSibling = "I remain constant";
constSibling = "I want to change, but I cannot"
const constSibling = "I can't re-declare";
console.log(constSibling);

if (true) {
  var varScope = "I am var and global-scoped ";
  let letScope = "I am let and block-scoped";
  const constScope = "I am const and block-scoped";
  console.log(letScope);
  console.log(constScope)
}

console.log(varScope);
console.log(letScope);
console.log(constScope);
console.log(hoistedVar);
var hoistedVar = "I am hoisted var means declared, but not assigned value";

console.log(hoistedLet)
let hoistedLet = "I am hoisted let means not initialized";

console.log(hoistedConst)
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)
[March 2024 \(6\)](#)
[January 2024 \(9\)](#)
[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```
const hoistedConst = "I am hoisted const means not initialized";
```

Template literals:

```
let name = "Hema";
let age = 21;
let info = "Student name is:" + name + " and " + age;
let infoTemp = `Student name is: ${name} and ${age}`;
console.log(info);
console.log(infoTemp)
```

Interview Questions:

1. What is the difference between var, let, and const when declaring variables?
2. How does let differ from var in terms of block scoping?
3. What are template literals, and how do they differ from traditional string concatenation?

at December 20, 2023

No comments:

Post a Comment



[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Top 10 | JavaScript Coding Interview Question | Beginner Level

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console

 Day 1 || Introduction of Html || Element || Structure of HTML || Full stack web development
HTML Topics Day 1 : 1. Introduction of Html: Who has invented HTML? - Sir Tim Berners-Lee <https://home.web...>

 Day 2 || Text Formatting tags in Html || List tags || Link tags|| Image tags || Full stack web development in telugu
HTML Topics Day 2 : Text Formatting List Link Image Table Form Text Formatting : List, Link and Image : Table: Form: Task: 1. By using t...

 Day 2 || Real-time Bank Project Using MERN
Hema Coding Bank Application Day 2: Images: App.js : import './App.css' ; import Home from "./Home" ; function App ()...

Hema Coding School

Thursday, December 21, 2023

Day 15 || Spread in JavaScript || Rest in JavaScript || Destructuring in JavaScript

JavaScript Topics

Day 15 :

- Spread
- Rest
- De-structuring

Spread:

```
// 1. Array
let array1 = [1, 2, 3, 4];
let array2 = [...array1, 5, 6, 7];
let array3 = [...array2, 8, 9, 10];
// console.log(array2);
// console.log(array3)

// 2. Object

let object1 = { a: 1, b: 2 };
let object2 = { ...object1, c: 3, d: 4 };
let object3 = { ...object2, e: 5, f: 6 };
// console.log(object2);
// console.log(object3);

// 3. function

function sum(a, b, c, d) {
  return a + b + c + d;
}
let numbers = [1, 2, 3, 4];
// console.log(sum(...numbers))
```

Rest:

```
function addingNumbers(...numbers) {
  // console.log(numbers)
  let total = 0;
  for(let i=0; i<numbers.length; i++){
    total = total + numbers[i]
  }
  return total
}
console.log(addingNumbers(...array3));
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

Hema Coding School

[View my complete profile](#)

De-structuring:

```
// 1. Object

let Student = {
  name: "Hema",
  role: "Developer",
  username: "Hema_H",
  email: "hema@example.com",
};

// console.log(Student.name)
// console.log(Student.role)

let { name, role: task, ...otherDetails } = Student;
// console.log(name)
// console.log(role)
// console.log(task);
// console.log(otherDetails)

// 2. Array

let number = [1,2,3,4,5,6,7,8,9]
// console.log(number[1])
// console.log(number[2])

let [first, second, third, ...remainingNumbers] = number;
console.log(first)
console.log(third)
console.log(remainingNumbers)
```

Tasks:**Spread:****1. Copying Arrays with Modifications:**

Given an array of numbers, create a function that makes a copy of the array and then adds a new number to the end of the copied array using the spread operator.

2. Combining Objects with Defaults:

Write a function that accepts an object and merges it with a default object, providing default values for missing keys using the spread operator.

Rest:**3. Calculating Average:**

Write a function that calculates the average of a variable number of arguments passed using the rest parameter.

De-structuring:**4. Extract Specific Values:**

Write a function that accepts an array containing a person's name, age, and country, and uses destructuring to extract these values into separate variables.

5. Alias Object Properties:



Hema Coding School

Thursday, December 21, 2023

Day 16 || Arrow function in JavaScript

JavaScript Topics

Day 16:

- Arrow Function

ArrowFunction:

```
// 1. Syntax

function traditionalFunction(x, y) {
  return x + y;
}
// console.log(traditionalFunction(2, 3));

let arrowFunction = (x, y) => {
  return x + y;
};

// console.log(arrowFunction(2,3))

// 2. this binding

let user = {
  name: "Hema",
  sayNamaste: function () {
    console.log("Namaste " + this.name);
  },
  sayHello: () => {
    console.log("Hello " + this.name);
  },
};

// user.sayNamaste()
// user.sayHello()

// 3. Arguments Object
function printArguments() {
  for (i = 0; i < arguments.length; i++) {
    console.log(arguments[i]);
  }
}
// printArguments(1, 2, 3, 4, 5);

let printArgumentsA = (...arguments) => {
  for (i = 0; i < arguments.length; i++) {
    console.log(arguments[i]);
  }
};
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```
// printArgumentsA(1, 2, 3, 4, 5);

// 4. Return

function multiply(x,y){
    return x*y;
}
console.log(multiply(2,3))

let multiplyA =(x,y)=> x * y;
console.log(multiplyA(4,5))
```

Task:**Task 1: Arrow Function with Parameters**

Write an arrow function called `calculateArea` that takes the `radius` of a circle as a parameter and returns the area of the circle. Use the formula: $\text{Area} = \pi \times \text{radius}^2$. Assume that π (pi) is approximately 3.14.

Task 2: Arrow Function with a Default Parameter

Write an arrow function called `power` that takes two parameters: a base (`x`) and an exponent (`y`). If no exponent is provided, default it to 2. The function should return the result of raising the base to the exponent.

Interview Questions:

1. How do arrow functions differ from regular functions in terms of syntax and behavior?
2. Explain the concept of lexical scoping and how it applies to arrow functions.
3. What is the main advantage of using arrow functions over traditional functions?

at December 21, 2023

No comments:**Post a Comment**[Newer Post](#)[Home](#)[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Top 10 | JavaScript Coding Interview Question | Beginner Level

JavaScript Coding Interview Q. No. 01/10: `console . log ("1" + "4" + "1") // 141 console`



Hema Coding School

Friday, December 22, 2023

Day 17 || Call Back Function || Array Methods in JavaScript || map || filter || reduce || map vs filter vs reduce in JavaScript

JavaScript Topics

Day 17:

- Call Back Function
- Array Methods
 - Map(),
 - Filter(),
 - Reduce()

Call Back Function:

```
function name(x) {
  console.log("Hema");
  role();
}

function role() {
  console.log("Student");
}

name(role);

let name = (x)=> {
  console.log("Hema");
  role();
}

let role = ()=> {
  console.log("Student");
}

name(role);
```

Array Methods:

```
let originalArray = [1, 2, 3, 4, 5];

let newArray = originalArray.map((currValue, ind, arr) => {
  // console.log(currValue * 2);
  return currValue % 2 === 0
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

});  
  

console.log(newArray);  
  

let originalArray = [1, 2, 3, 4, 5];
let multiply = {
  number: 2,
};  
  

let newArray = originalArray.map(function(currValue, ind, arr) {
//   console.log(currValue * this.number);
  return currValue * this.number;
}, multiply);  
  

console.log(newArray);
let originalArray = [1, 2, 3, 4, 5];  
  

let filterArray = originalArray.filter((currValue, ind, arr) => {
  return currValue % 2 === 0;
});  
  

console.log(filterArray);  
  

let originalArray = [6, 1, 2, 3, 4, 9, 5, 6, 8, 7];  
  

let reduceArray = originalArray.reduce((acc, currValue, ind, arr) => {
  return acc + currValue;
});  
  

console.log(reduceArray);  
  

let originalArray = [1, 2, 3, 4, 5];  
  

let newArray = originalArray.map((currValue, ind, arr) => currValue * 3);
console.log(newArray);  
  

let evenNumber = newArray.filter((currValue) => currValue % 2 === 0);
console.log(evenNumber);  
  

let singleValue = evenNumber.reduce((acc, currValue) => acc + currValue);
console.log(singleValue);
let newArray = originalArray
  .map((currValue, ind, arr) => currValue * 3)
  .filter((currValue) => currValue % 2 === 0)
  .reduce((acc, currValue) => acc + currValue);
console.log(newArray);

```

Tasks:

Task 1: Call Back

Write a function calculate that takes two numbers and a callback function as arguments. The callback function should perform a mathematical operation (addition, subtraction, multiplication, or division) on the two numbers.

Task 2: Map + Filter

Given an array of words, create a new array that contains the lengths of words that have more than three characters.

```
const words = ['apple', 'banana', 'pear', 'grape', 'kiwi'];
```

Task 3: Map + Reduce

Given an array of numbers, create a new array that contains the square of each number and then calculate the sum of the squared values.

Hema Coding School

Saturday, December 23, 2023

Day 18 || Module in Node.js || Export in JavaScript || Import in JavaScript

JavaScript Topics

Day 18:

- Module
 - Export
 - Import
- Page Creation

export.mjs

```
// =====Node.js=====
let greeting = "Hello";
let name = "Hema";

let sayHello = () => {
  return `${greeting} ${name}, How are you?`;
};

module.exports = { sayHello };

//===== ES6 =====
export function sayGreeting() {
  return "Hello Hema, what's going on";
}
```

import.mjs

```
// =====Node.js=====

let myCall = require('./export.js');
console.log(myCall.sayHello())

//===== ES6 =====

import { sayGreeting } from './export.mjs';
console.log(sayGreeting());
```

StudentInfo.html:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

Search This Blog

Search

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

<title>Document</title>
<link rel="stylesheet" type="text/css" href="./Home.css" />
<link rel="stylesheet" type="text/css" href="./StudentInfo.css" />
</head>
<body>
<header>
    <h1>Hema Coding School</h1>
</header>
<nav>
    <a href=".//Home.html">Home</a>
    <a href=".//Courses.html">Courses</a>
    <a href=".//About.html">About Us</a>
    <a href=".//Contact.html">Contact</a>
    <a href=".//StudentInfo.html">Student Information</a>
</nav>
<main>
    <section>
        <marquee>Welcome to Hema Coding School</marquee>
    </section>
    <section>
        <h2>Student Information</h2>
        <table id="studentInfoTable">
            <thead>
                <tr>
                    <th>S. No</th>
                    <th>Name</th>
                    <th>Attendance</th>
                    <th>Feedback</th>
                </tr>
            </thead>
            <tbody id="studentInfoBody">
                </tbody>
        </table>
    </section>
</main>
<footer>
    <p>&copy; 2024 Hema Coding School. All rights reserved.</p>
</footer>
<script src=".//StudentInfo.js"></script>
</body>
</html>

```

StudentInfo.css:

```

table {
    width: 100%;
    margin-top: 1em;
}
table, th, td{
    border: 1px solid #ddd;
}
th{
    background-color: #efdcfc;
}
th, td{
    padding: 0.8em;
    text-align: center;
}

```

StudentInfo.js:

```

const students = [
    { id: 1, name: "Hema", attendance: 80, feedback: "Excellent" },
    { id: 2, name: "Mahesh", attendance: 75, feedback: "Good" },
]

```

```

{ id: 3, name: "Maruthi", attendance: 90, feedback: "Very Nice" },
};

let displayStudents = () => {
  let studentInfoBody = document.getElementById("studentInfoBody");
  students.map((student) => {
    console.log(student.name);

    let row = document.createElement('tr');
    row.innerHTML = `<td>${student.id}</td><td>${student.name}</td><td>${student.attendance}%</td><td>${student.feedback}</td>`;
    studentInfoBody.appendChild(row)
  });
};

let addStudent = (id, name, attendance, feedback)=>{
  let studentData = students.push({id, name, attendance, feedback})
}

addStudent(4, "Riya", 95, "Out Standing");
addStudent(5, "Rathan", 85, "Very Good");
addStudent(6, "Girija", 95, "Outstanding");

displayStudents()

```

Interview Questions:

1. What is the purpose of the export statement in JavaScript?
2. How do you use the export statement to export a variable or function?
3. Explain the difference between named exports and default exports?

at December 23, 2023

No comments:

Post a Comment



[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Top 10 | JavaScript Coding Interview Question | Beginner Level

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console



Hema Coding School

Monday, December 25, 2023

Day 19 || Introduction of React ||
Integration of React into an HTML
structure || Including React and React
DOM from CDN Links || JSX

React.js Topics

Day 19:

- Introduction of React.js
- Integration of React into an HTML structure
 - Including React from CDN Links
 - Including React DOM from CDN Links
- JSX

Including React from CDN Links :

CDN Links:

```
<script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>

React.createElement(
  type,           // The type of the HTML element or React component
  [props],         // Optional: The properties or attributes for the element/component
  [...children] // Optional: The children of the element/component
)
```

Ex: `React.createElement("h1", {id:"1", style:{color:"red", padding:"20px"}}, "Hello React")`

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

Including React DOM from CDN Links :

CDN Links:

```
<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>

ReactDOM.render(
  React.createElement( // The React element to be rendered
    "h1", // The DOM element where the rendered element should be placed
    "Welcome to Hema coding school", // Content of the element
    () => { // Optional: A function to be executed after the rendering is complete
      console.log("Rendered successfully!")
    }
);
```

JSX:

- JSX is a syntax extension for JavaScript
- JSX = HTML/XML + JavaScript

```
React.createElement("h1", {id:"1", style:{color:"red"}}, "Welcome to Hema coding school")
```

Babel is compiler will convert JSX syntax into native
react for browser understands

In JSX: <h1 id="1", style:{{color:"red"}}>Welcome to Hema coding school</h1>

ReactWithCDNLinks.html:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <script
      crossorigin
      src="https://unpkg.com/react@18/umd/react.development.js"
    ></script>
    <script
      crossorigin
      src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"
    ></script>
    <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
  </head>
  <body id="root">
    <script type="text/babel">
      // let h1Tag = React.createElement(
      //   "h1",
      //   { id: 1, style: { color: "red" } },
      //   "Welcome to Hema coding school",
      // );
```

```

// [
//   React.createElement(
//     "p",
//     null,
//     "Your presence is very important to us"
//   ),
//   React.createElement("div", null, "However, guests are limited"),
// ]
// );
function reactReuse() {
  let h2TagJSX = (
    <h1 id="1" style={{ color: "blue" }}>
      Welcome to Hema coding school
      <p>Your presence is very important to us</p>
      <div>However, guests are limited</div>
    </h1>
  );
  return h2TagJSX;
}
function reactCall() {
  let functionCall = (
    <div>
      {reactReuse()}
      {reactReuse()}
      {reactReuse()}
      {reactReuse()}
    </div>
  );
  ReactDOM.render(functionCall, document.getElementById("root"), () => {
    alert("Rendered successfully");
  });
}
reactCall();
</script>
</body>
</html>

```

Interview Questions:

1. What is the role of React.createElement in integrating React into an HTML structure?
2. Why might you choose to include React and React DOM from CDN links instead of installing them locally?
3. Explain how JSX gets transpiled into JavaScript and why this step is necessary.
4. Walk through the process of rendering a React element into the DOM using ReactDOM.render

at December 25, 2023

No comments:

Hema Coding School

Friday, December 29, 2023

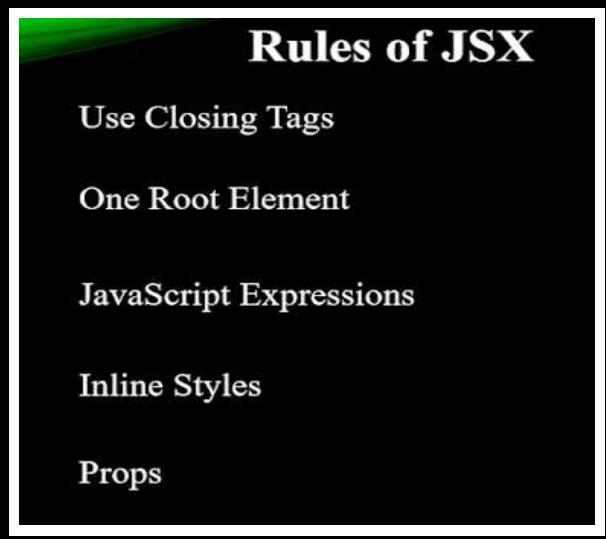
Day 20 || Rules of JSX || Components and Props in React js

React.js Topics

Day 20:

- Rules of JSX
- Components and Props

Rules of JSX:



ReactWithCDNLinks.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <script
      crossorigin
      src="https://unpkg.com/react@18/umd/react.development.js"
    ></script>
    <script
      crossorigin
      src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"
    ></script>
    <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
  </head>
  <body id="root">
    <script type="text/babel">
      function reactReuse(name, age, img) {
        return (
          <div
            style={{
              width: "250px",

```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

        textAlign: "center",
        boxShadow: "0 0 10px black",
        padding: "10px",
    )}
>
<img src={img} alt="TATA" width="100%" height="300px" />
<h2>{name}</h2>
<p>{age}</p>
</div>
);
}
function reactCall() {
    return (
        <div style={{ display: "flex", justifyContent: "space-between" }}>
            {reactReuse(
                "Rathan Tata",
                86,
                "https://assets.gqindia.com/photos/645e034efc79052643f24e8e/16:9/w_1920,c_limit/Ratan-Tata.jpg"
            )}
            {reactReuse(
                "Sundar Pichai",
                51,
                "https://sugermint.com/wp-content/uploads/2020/04/Biography-of-Sundar-Pichai.jpg"
            )}
            {reactReuse(
                "Nadella Satya",
                56,
                "https://akm-img-a-in.tosshub.com/indiatoday/images/story/202306/rtx6otlw-sixteen_nine.jpg?VersionId=G9LYYNj7sPjP7Bv"
            )}
            {reactReuse(
                "Jaya Varma Sinha",
                60,
                "https://sambadenglish.com/wp-content/uploads/2023/08/Jaya-Verma-Sinha-e1693478762805.jpg"
            )}
        </div>
    );
}
ReactDOM.render(reactCall(), document.getElementById("root"));
reactCall();
</script>
</body>
</html>

```

Components and Props:

ComponentAndProps.html:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <script
        crossorigin
        src="https://unpkg.com/react@18/umd/react.development.js"
    ></script>
    <script
        crossorigin
        src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"
    ></script>
    <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
</head>
<body id="root">
    <script type="text/babel">
        function ReactReuse(props) {
            return (
                <div
                    style={{
                        width: "250px",
                        textAlign: "center",
                        boxShadow: "0 0 10px black",
                        padding: "10px",
                    }}
                >
                    <img src={props.img} alt="TATA" width="100%" height="300px" />
                
```

```

        <h2>{props.name}</h2>
        <p>{props.age}</p>
    </div>
);
}
function ReactCall() {
    return (
        <div style={{ display: "flex", justifyContent: "space-between" }}>
            <ReactReuse
                name="Rathan TATA"
                age={86}
                img="https://assets.gqindia.com/photos/645e034efc79052643f24e8e/16:9/w_1920,c_limit/Ratan-Tata.jpg"
            />

            <ReactReuse
                name="Sundar Pichai"
                age={51}
                img="https://sugermint.com/wp-content/uploads/2020/04/Biography-of-Sundar-Pichai.jpg"
            />
            <ReactReuse
                name="Nadella Satya"
                age={56}
                img="https://akm-img-a-in.tosshub.com/indiatoday/images/story/202306/rtx6otlw-sixteen_nine.jpg?VersionId=G9LYYNj7sP"
            />
            <ReactReuse
                name="Jaya Varma Sinha"
                age={60}
                img="https://sambadenglish.com/wp-content/uploads/2023/08/Jaya-Verma-Sinha-e1693478762805.jpg"
            />
        </div>
    );
}
ReactDOM.render(<ReactCall/>, document.getElementById("root"));

</script>
</body>
</html>

```

Interview Questions:

1. How do you embed JavaScript expressions in JSX?
2. What is a React component?
3. How do you access props in a functional component?

at December 29, 2023

No comments:

Post a Comment



SIGN IN WITH GOOGLE





Hema Coding School

Saturday, December 30, 2023

Day 21 || Setting up development environment || CRA || Create React App || npx create-react-app || Folder structure || Entry point in react.js App

React.js Topics

Day 21:

- Setting up development environment
 - CRA : Create React App
 - Folder structure
 - Entry point

CRA:

How to create React app using CRA?

- npx create-react-app <app Name>

How to run React Application ?

- npm start

How to stop React Application ?

- Ctrl + c

Folder structure:

Node_Modules

Public

- Index.html

Src

- App.css
- App.js
- Index.css
- Index.js

Package.json

Package-lock.json

Entry point:

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

npm start => package.json (configuration and dependencies) =>
 Webpack and Babel bundle (transform and bundle code) => index.js
 (entry point) => index.html (HTML template with injected bundled
 code).

Interview Questions:

1. How does CRA handle Webpack and Babel configuration?
2. What is the purpose of the npm start command in a Create React App project?
3. what is difference between npm and npx

at December 30, 2023

No comments:

Post a Comment



SIGN IN WITH GOOGLE

[Newer Post](#)

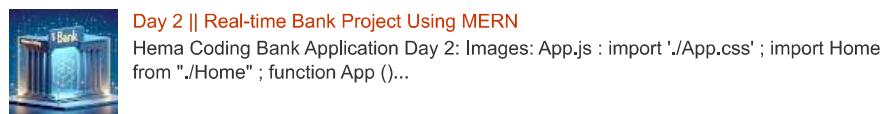
[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Top 10 | JavaScript Coding Interview Question | Beginner Level

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console





Hema Coding School

Thursday, January 4, 2024

Day 22 || Functional Component in React || Styling in React || Props in React

React.js Topics

Day 22:

- Functional Component
- Styling in React
- Props

Functional Component:

HouseComponent.js

```
import "./HouseComponent.css";
const HouseComponent = () => {
  return (
    <div className="villageContainer">
      <h1>Color Full Village</h1>
    </div>
  );
};

export default HouseComponent;
```

Styling in React:

HouseComponent.css

```
.villageContainer {
  color: rgb(240, 231, 231);
  text-align: center;
  background: url("https://w0.peakpx.com/wallpaper/190/244/HD-wallpaper-beautiful-landscape-nature-green-field-sky-vi");
  height: 100vh;
  background-size: cover;
  padding: 100px;
  margin: 10px;
  border-radius: 5px;
}
```

Props:

DeliveryService.js

```
import "./DeliveryService.css";
const DeliveryService = ({ item, destination, img }) => {
  return (
    <div className="delivery-service-container">
      <h1>Delivery Details</h1>
      <div>
        <h2>
          {item} items sending to {destination}
        </h2>
      </div>
    </div>
  );
};

export default DeliveryService;
```

Search This Blog

Search

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

        </h2>
    </div>
    <div className="delivery-service-image">
        <img src={img} alt={item} />
    </div>
</div>
);
};

export default DeliveryService;

```

DeliveryService.css

```

.delivery-service-container{
    text-align: center;
    border: 1px solid blue;
    margin: 10px;
    border-radius: 5px;
}
.delivery-service-container img{
    width: 60%;
    border-radius: 5px;
}

```

MarketComponent.js

```

import DeliveryService from "./DeliveryService.js";
const MarketComponent = () => {
    return (
        <div>
            <DeliveryService
                item="Groceries"
                destination="Red House"
                img="https://images.unsplash.com/photo-1604719312566-8912e9227c6a?auto=format&fit=crop&q=80&w=1974&ixlib=rb-4.0.0"
            />
            <DeliveryService
                item="Package"
                destination="Blue House"
                img="https://images.unsplash.com/photo-1556229040-2a7bc8a00a3e?auto=format&fit=crop&q=80&w=1930&ixlib=rb-4.0.0"
            />
            <DeliveryService
                item="Mail"
                destination=" Green House"
                img="https://images.unsplash.com/photo-1528329140527-75853b1e1650?auto=format&fit=crop&q=80&w=1935&ixlib=rb-4.0.0"
            />
        </div>
    );
}
export default MarketComponent;

```

App.js

```

import HouseComponent from "./Components/HouseComponent";
import MarketComponent from "./Components/MarketComponent";
function App() {
    return (
        <div>
            <HouseComponent />
            <MarketComponent />
        </div>
    );
}

export default App;

```

Index.js

```

import React from "react";
import ReactDOM from "react-dom/client";
import "./index.css";
import App from "./App";

```

Hema Coding School

Friday, January 5, 2024

Day 23 ||Event handling in React ||Form in React

React.js Topics

Day 23:

- Event Handling
- Form and form validation

Form and form validation:

ArtCompetitionForm.js

```
import './ArtCompetitionForm.css';
const ArtCompetitionForm = () => {
  const handleSubmit = (event) => {
    const formData = {
      name: event.target.name.value,
      age: event.target.age.value,
      artworkDescription: event.target.artworkDescription.value,
    };
    console.log(formData)
    // alert("Clicked");
    event.preventDefault();
    event.target.name.value="";
    event.target.age.value="";
    event.target.artworkDescription.value ="";
  };
  //  const handleChange = (event)=>{
  //    console.log(event.target.value, "ID")
  //  }
  return (
    <div className="art-competition-form-container">
      <h1>Art Competition Form</h1>
      <form onSubmit={handleSubmit}>
        <label>Name: </label>
        <input type="text" name="name" required/>
        <label>Age: </label>
        <input type="number" name="age" required/>
        <label>Artwork Description: </label>
        <textarea name="artworkDescription" required/>
        <button type="submit">Submit</button>
      </form>
    </div>
  );
};

export default ArtCompetitionForm;
```

ArtCompetitionForm.css

```
.art-competition-form-container {
  background: url("https://images.unsplash.com/photo-1510935813936-763eb6fbc613?q=80&w=1778&auto=format&fit=crop&ixlib=rb-4.0.3&i");
  background-size:100% 100%;
  height: 100vh;
```

[Search This Blog](#)

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

        display: flex;
        justify-content: center;
        flex-direction: column;
        align-items: center;
    }
.art-competition-form-container h1, label{
    display: block;
    margin: 5px;
}
.art-competition-form-container input, textarea{
    padding: 10px;
    width: 100%;
    border-radius: 5px;
    border: none;
}
.art-competition-form-container button[type="submit"]){
    width: 100%;
    background-color: blue;
    padding: 8px;
    color: white;
    border-radius: 5px;
    border: none;
    margin: 5px;
}
.art-competition-form-container button[type="submit"]:hover{
    background-color: green;
    cursor: pointer;
}

```

Interview Questions:

1. Explain the concept of event handling in React?
2. What is the purpose of the onChange event in React? Provide an example.
3. What is the purpose of the e.preventDefault() method in a React form?

at [January 05, 2024](#)

No comments:

Post a Comment



[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

[Top 10 | JavaScript Coding Interview Question | Beginner Level](#)

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console





Hema Coding School

Saturday, January 6, 2024

Day 24 || useState in React || useEffect in React || React hooks

React.js Topics

Day 24:

- React Hooks
 - useState()
 - useEffect()

useState():

Playground.js

```
import { useState } from "react";
import "./Playground.css";
const Playground = () => {
  const [gameScore, setGameScore] = useState(0);
  const [artistName, setArtistName] = useState("Mahesh")
  const handleIncrement = () => {
    setGameScore(gameScore + 1);
  };
  const handleDecrement = () => {
    if (gameScore > 0) {
      setGameScore(gameScore - 1);
    }
  };
  const handleChangeName = (e)=>{
    setArtistName(e.target.value)
  }
  return (
    <div className="playground-container">
      <h1>Playground : Art Competition</h1>
      <h3>Artist Name: <span> {artistName}</span></h3>
      <h3>Game Score: <span>{gameScore}</span></h3>
      <div>
        <input type="text" onChange={handleChangeName}/>
        <button onClick={handleIncrement}>Increase Score</button>
        <button onClick={handleDecrement}>Decrease Score</button>
      </div>
    </div>
  );
}
export default Playground;
```

Playground.css

```
.playground-container {
  background: url("https://images.unsplash.com/photo-1604921827342-b4bc94df162c?q=80&w=1933&auto=format&fit=crop&ixlib");
  background-size: 100% 100%;
  height: 100vh;
  text-align: center;
  border-radius: 5px;
  margin: 10px;
```

Search This Blog

Search

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

    }
    .playground-container span{
      color: red;
    }
    .playground-container input{
      padding: 10px;
      border-radius: 5px;
      border: none;
    }
    .playground-container button {
      padding: 10px;
      background-color: rgb(104, 119, 219);
      color: white;
      border-radius: 5px;
      border: none;
      margin: 5px;
    }
  }

```

useEffect():**DataFetchingApi.js**

```

import { useEffect, useState } from "react";
const DataFetchingApi = () => {
  const [data, setData] = useState([]);
  useEffect(() => {
    fetch("https://jsonplaceholder.typicode.com/todos")
      .then((response) => {
        return response.json();
      })
      .then((json) => {
        console.log(json);
        return setData(json);
      });
  }, []);
}

return (
  <div>
    Data Fetching Api
    <div>{data.map((item)=>{
      console.log(item.title);
      return(<li>{item.title}</li>)
    })}</div>
  </div>
);
};

export default DataFetchingApi;

```

Interview Questions:

1. What is a React Hook?
2. What is the purpose of the second element returned by useState?
3. How do you update state in a functional component?
4. Explain the purpose of the dependency array in useEffect?
5. What are common use cases for useEffect?

at [January 06, 2024](#)

No comments:



Hema Coding School

Sunday, January 7, 2024

Day 25 || React Router || Router in React
|| Routes in React || Route in React || Link
in React

React.js Topics

Day 25:

- React Router
 - Router
 - Routes
 - Route
 - Link

React Router:

```
Router: npm install react-router-dom

import { BrowserRouter as Router, Routes, Route, Link } from 'react-router-dom';

<Router>
  <Link to="/" />
  <Routes>
    <Route path="/" element={Component Name} />
  </Routes>
</Router>
```

App.js:

```
import HouseComponent from "./Components/HouseComponent";
import MarketComponent from "./Components/MarketComponent";
import ArtCompetitionForm from "./Components/ArtCompetition";
import Playground from "./Components/Playground";
import {
  BrowserRouter as Router,
  Routes,
  Route,
  Link,
} from "react-router-dom";
import './App.css'
function App() {
  return (
    <Router>
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```

<div className="app-container">
  <div className="sidebar">
    <Link to="/" className="sidebar-btn">Home</Link>
    <Link to="/deliveryService" className="sidebar-btn">Delivery Service</Link>
    <Link to="/artCompetition" className="sidebar-btn">Art Competition</Link>
    <Link to="/playground" className="sidebar-btn">Play Ground</Link>
  </div>
  <Routes>
    <Route path="/" element={<HouseComponent />}/>
    <Route path="/deliveryService" element={<MarketComponent />}/>
    <Route path="/artCompetition" element={<ArtCompetitionForm />} />
    <Route path="/playground" element={<Playground />} />
  </Routes>
</Router>
);
}

export default App;

```

App.css

```

.app-container {
  display: flex;
}

.sidebar{
  background-color: #202020;
  padding: 20px;
  flex: 0 0 240px;
}

.sidebar-btn{
  color: white;
  display: block;
  padding: 10px;
  text-decoration: none;
}

.sidebar-btn:hover{
  background-color:gray;
  border-radius: 5px;
  color:red
}

```

Interview Questions:

1. What is React Router, and why is it used in React applications?
2. Explain the significance of the Link component in React Router. How is it different from an anchor (<a>) tag?
3. What is the purpose of the exact prop in the Route component?



Hema Coding School

Friday, January 12, 2024

Day 26 || Introduction to Node.js || Operating System Module || File System Module || HTTP Module in node.js

Node.js Topics

Day 26:

- Introduction to Node.js
- Operating System Module
- File System Module
- HTTP Module
 - Create Server

App.js

```
function sayHello(){
  return "Welcome to Hema coding School"
}
// module.exports= sayHello;
School = {
  name:"Hema",
  role:"Student"
}
module.exports = {sayHello, School}
```

Index.js

```
const myCall = require('./App.js');
console.log(myCall.sayHello())
console.log(myCall.School)

const os = require("os")
console.log(os.freemem())
console.log(os.totalmem())

const fs = require('fs');
// (file, data, [optional], callback)
fs.writeFile("output", "Welcome to hema coding School", "utf-8", (error)=>{
  if(error){
    console.log(error)
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

[April 2024 \(1\)](#)

[March 2024 \(6\)](#)

[January 2024 \(9\)](#)

[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```
        }
    else{
        console.log("file has created successfully")
    }
})
// path, [optional], callback
fs.readFile("./output","utf-8",(error,data)=>{
    if(error){
        console.log(error)
    }
    else{
        console.log(data)
    }
})

const http = require('http');
const server = http.createServer((req,res)=>{
    res.writeHead(200,{"Content-Type": "text/plain"})
    res.end("Welocme hema coding school")
})
// PORT, HOSTNAME, BACKLOG, CALLBACK
const PORT = 3000;
const HOSTNAME = "127.0.0.1";
const BACKLOG = 551;
const CALLBACK = ()=>{
    console.log(`Server has started http://${HOSTNAME}:${PORT}`)
}
server.listen(PORT, HOSTNAME, BACKLOG, CALLBACK);
```

Interview Questions:

1. Explain the key features of Node.js?
2. What is the purpose of the Operating System Module in Node.js?
3. What is the role of the fs.createReadStream and fs.createWriteStream methods?
4. Explain the process of creating an HTTP server in Node.js?

at [January 12, 2024](#)

No comments:

Post a Comment



Hema Coding School

Sunday, January 14, 2024

Day 27 || Express js || Express Framework || Get method in Express || Post Method in Express

Node.js Topics

Day 27:

- Express Framework
 - Get
 - Post

Express.js

```
const express = require('express');
const callingData = require('./Data.js')
const app = express()
const bodyParser = require('body-parser')
app.use(bodyParser.json())
app.use(bodyParser.urlencoded({extended:false}))
app.get('/',(req,res)=>{
res.send("Hello, Hema coding school");
})
app.get('/about',(req,res)=>{
  res.send("About , Hema coding school")
})
app.get('/item',(req,res)=>{
  res.send(`<h1>Hema Coding School</h1>
<h2>${callingData[0].id}</h2>
<h2>${callingData[0].name}</h2>
<h2>${callingData[1].id}</h2>
<h2>${callingData[1].name}</h2>
<h2>${callingData[2].id}</h2>
<h2>${callingData[2].name}</h2>
`)
})
app.post('/api',(req,res)=>{
  newData={
    id:req.body.id,
    name: req.body.name
  }
  callingData.push(newData)
  res.json(callingData)
})
const PORT = 3000;
const HOSTNAME = "127.0.0.1";
const BACKLOG = 551;
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```
app.listen(PORT, HOSTNAME, BACKLOG, ()=>{
  console.log(`Server is running at http://${HOSTNAME}:${PORT}`)
})
```

Data.js

```
const items = [
  {
    id: 1,
    name: "Hema",
  },
  {
    id: 2,
    name: "Mahesh",
  }
];

module.exports = items
```

Interview Questions:

1. How does routing work in Express.js?
2. What is middleware in Express and how is it used in the context of a GET request?
3. How do you handle a POST request in Express?

at [January 14, 2024](#)

No comments:

Post a Comment



[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

[Top 10 | JavaScript Coding Interview Question | Beginner Level](#)

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console

Hema Coding School

Monday, January 15, 2024

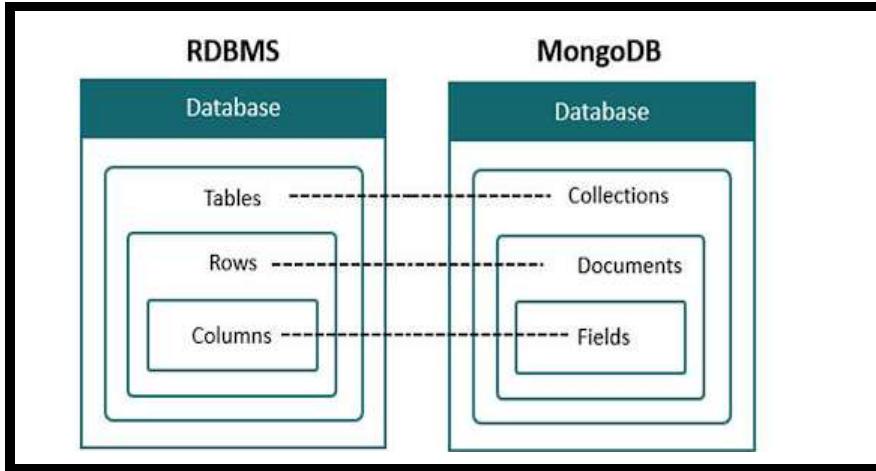
Day 28 || Introduction to MongoDB || Setting up MongoDB || CRUD operations with MongoDB

MongoDB Topics

Day 28:

- Introduction to MongoDB
- Setting up MongoDB
- CRUD operations with MongoDB

Introduction to MongoDB:



Setting up MongoDB:

Search This Blog

[Home](#)

[Report Abuse](#)

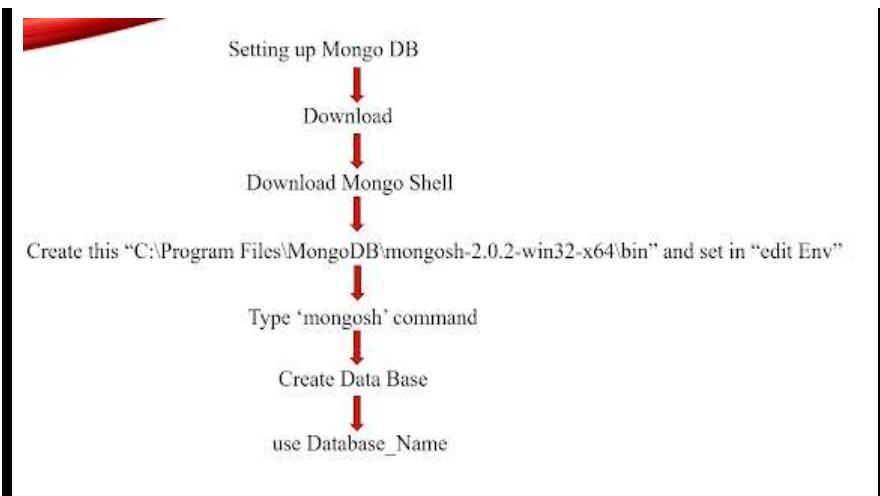
Blog Archive

[April 2024 \(1\)](#)
[March 2024 \(6\)](#)
[January 2024 \(9\)](#)
[December 2023 \(20\)](#)

About Me

[Hema Coding School](#)

[View my complete profile](#)



CRUD operations with MongoDB:

C → Create db.collectionName.insertOne({})

R → Read db.collectionName. find()

U → Update db.collectionName. updateOne({})

D → Delete db.collectionName. deleteOne({})

To read:

show databases

show collections

Create collection:

use HemaDatabase

db.createCollection('HemaCodingSchool')

db.HemaCodingSchool.insertOne({ name: 'Mahesh', role: 'Student' })

Create:

```

db.HemaCodingSchool.insertOne(
    { name: 'Mahesh', role: 'Student' }
)
  
```

Read:

db.HemaCodingSchool.find({})

Update:

```

db.HemaCodingSchool.updateOne(
    { name: 'Mahesh'}, {$set:{ role: 'Employee'}}
)
  
```

```
)-----
```

Delete:

```
db.HemaCodingSchool.deleteOne({name: 'Mahesh'})-----
```

Create Many:

```
db.HemaCodingSchool.insertMany([
  { name: 'Hema', role: 'Organizer' },
  { name: 'Mahesh', role: 'Employee' },
  { name: 'Maruthi', role: 'Student' }
])-----
```

Update Many:

```
db.HemaCodingSchool.updateMany(
  { name: { $in: ['Hema', 'Mahesh', 'Maruthi'] } },
  { $set: { role: 'NewRole' } }
)-----
```

Delete Many:

```
db.HemaCodingSchool.deleteMany(
  { name: { $in: ['Hema', 'Mahesh', 'Maruthi'] } }
)-----
```

Interview Questions:

1. What is a Collection in MongoDB?
 2. How do you insert a document in MongoDB?
 3. Explain the \$set operator in MongoDB?
-

at [January 15, 2024](#)

No comments:**Post a Comment**A large rectangular area of the page has been redacted with black ink, obscuring several lines of text that would normally be displayed as comments.

SIGN IN WITH GOOGLE



Hema Coding School

Wednesday, January 17, 2024

Day 29 || Mongoose ODM for Node.js || Connect || Schema || Model || Connection of Front-End + Back-End + Data Base || Connection of React.js + Node.js + MongoDB

MongoDB Topics

Day 29:

- Mongoose ODM for Node.js
 - Connect
 - Schema
 - Model
- Connection of Front-End + Back-End + Data Base
 - React.js + Node.js + MongoDB

Mongoose ODM for Node.js :

Mongoose.js:

```
const mongoose = require('mongoose');
mongoose.connect("mongodb://localhost:27017/myDataBase",{
  useNewUrlParser: true,
  useUnifiedTopology: true,
})

const userSchema = new mongoose.Schema({
  username: String,
  email: String,
  password: String
})

const User = mongoose.model('User',userSchema)

const newUser = new User({
  username:"Hema",
  email:"hema@gmail.com",
  password: "123"
})

newUser.save().then((user)=>{
  console.log(user)
}).catch((err)=>{
  console.log(err)
})
```

Search This Blog

[Home](#)

[Report Abuse](#)

Blog Archive

April 2024 (1)
March 2024 (6)
January 2024 (9)
December 2023 (20)

About Me

[Hema Coding School](#)

[View my complete profile](#)

```
User.updateOne({username:"Hema"},{password:"hema@123"}).then((result)=>{
  console.log(result)
})
```

Connection of Front-End + Back-End + Data Base:

server.js:

```
const mongoose = require('mongoose');
const express = require('express');
const cors = require('cors')
const app = express();
const bodyParser = require("body-parser");
app.use(bodyParser.json())
app.use(cors())
app.post("/submitFormData", (req,res)=>{
  const formData = req.body;
  const artFormData = new ArtForm({
    name:formData.name,
    age:formData.age,
    artworkDescription:formData.artworkDescription
  })
  artFormData.save().then((result)=>{
    console.log(result)
    res.status(200).json({ message: "Form submitted successfully!" });
  })
})

mongoose.connect("mongodb://localhost:27017/myDataBase",{
  useNewUrlParser: true,
  useUnifiedTopology: true,
})

const ArtSchema = new mongoose.Schema({
  name:String,
  age:String,
  artworkDescription:String
})

const ArtForm = mongoose.model("ArtForm", ArtSchema)

// const newArtForm = ArtForm({
//   name:"Mahesh",
//   age:"23",
//   artworkDescription:"Nice work"
// })

// newArtForm.save()
const PORT = 3001

app.listen(PORT, ()=>{
  console.log(`Server is running on port ${PORT}`)
})
```

ArtCompetitionForm.js:

```
fetch("http://localhost:3001/submitFormData",{
  method:"POST",
  headers:{
    "Content-Type" : "application/json"
  },
  body:JSON.stringify(formData)
```

```

        }).then((result)=>{
            console.log(result,"Form submitted successfully!")
        }).catch((err)=>{
            console.log(err)
        })
    )

```

Interview Questions:

1. How do you connect to a MongoDB database using Mongoose?
2. How do you create a Mongoose Schema?
3. What is a Mongoose Model?
4. How do you establish a connection between Node.js and MongoDB in a MERN stack application?
5. How can you handle CORS issues when connecting a React.js frontend to a Node.js backend?

at [January 17, 2024](#)

No comments:

Post a Comment



[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Top 10 | JavaScript Coding Interview Question | Beginner Level

JavaScript Coding Interview Q. No. 01/10: console . log ("1" + "4" + "1") // 141 console



[Day 2 || Real-time Bank Project Using MERN](#)