CSS & JAVASCRIPTS

CSS

Kinds of CSS
- Inline CSS
- Internal CSS
- External CSS

Selector :
- Basic selectors
  - Id (#)
  - Class (.)
  - Element (*element_name*)
  - Universal (*)
- Relational selectors
  - Descendant selectors
  - Child combiner
  - Adjacent sibling
  - General sibling selector
  - Attribute selector (^,$,*)
- Pseudo selectors
  - Link
  - Visited
  - Hover
  - Active
  - First-child
  - last-child

Borders
- Solid
- Dotted
- Dashed
- Double
- Groove
- Inset
- Outset
- None
- Hidden

Positions
- Static
- Relative
- Fixed
- Absolute
- Sticky
- Inherit (optional)

CSS units
- Absolute units
  - Cm = 10 mm
  - Mm = 1/10 cm
  - In = 2.54cm
  - Pt = 1/72 in
  - Pc = ⅙ in (6pc = 1in)
  - Px = 1/96 in
-

CSS & JAVASCRIPTS

Relative units
- o %
- o Em
- o Ex (x)
- o Ch (0)
- o Vh
- o Vw

Flex
- Display
- Flex-direction
  - o Row
  - o Row-reverse
  - o Column
  - o column-reverse
- Flex-wrap
- Flex-flow
- Justify-content:
  - o Flex-start
  - o Flex-end
  - o Center
  - o Space-around
  - o Space-between
- Align-items
  - o Flex-start
  - o Flex-end
  - o Center
  - o Stretch

Sample code for CSS

style.css

```css
*{
    font-family: Arial, Helvetica, sans-serif;
}

.parent{
    display: flex;
    flex-flow: row wrap;
    justify-content: center;
}

.child-1{
    width: 25%;
    margin: 1%;
    padding: 1%;
    border: 1px solid #ddd;
    text-align: center;
    box-shadow: 0px 8px 8px -8px #000;
    height: 180px;
}

.child-2{
```

```css
    width: 65%;
    margin:1%;
    padding: 1%;
    border: 1px solid #ddd;
    box-shadow: 0px 8px 8px -8px #000;
}

.child-1 a{
    display: block;
    text-decoration: none;
    color: #4145bc;
}

.child-1,
.child-2 h2{
    color: #ad7f2a;
    text-shadow: 3px 3px 3px #ddd;
}

.child-1 em{
    color: #978f81;
}
```

Index.html

```html
<!DOCTYPE html>
<html lang="en-US">
    <head>
        <title> Sample Web document </title>
        <link rel="stylesheet" href="css/style.css"/>
    </head>
    <body>
        <section class="parent">
            <article class="child-1">
                <h2> HANUMAN KUMAR P </h2>
                <em> Multi skill trainer, APSSDC</em> <hr>
                <a href="mailto:hanumankumar.p@apssdc.in">
hanumankumar.p@apssdc.in </a>

                <a href="tel:+91 9878685846"> +91 9878685846 </a>
            </article>

            <article class="child-2">
                <h2> Summary </h2> <hr>
                <ul>
                    <li> Having 5+ years of experience as a software
developer, apart from this having experience in teaching also. </li>
                    <li>
                        Having good exposure on front-end (HTML, CSS,
Bootstrap, Materilize, JavaScript, React) and back-end technologies too
(PHP, Express, codeigniter and python),
                    </li>

                    <li>
                        Developed database connections for middle level
and high level projects by using mySql and noSql.
```

CSS & JAVASCRIPTS

```html
                </li>

                <li>
                    Having knowledge on API(s) such as REST, SOAP
and GraphQl
                </li>
            </ul>
        </article>
    </section>
</body>
</html>
```

Output:

Media queries

HANUMAN KUMAR P

Multi skill trainer, APSSDC

hanumankumar.p@apssdc.in
+91 9878685846

Summary

- Having 5+ years of experience as a software developer, apart from this having experience in teaching also.
- Having good exposure on front-end (HTML, CSS, Bootstrap, Materilize, JavaScript, React) and back-end technologies too (PHP, Express, codeigniter and python),
- Developed database connections for middle level and high level projects by using mySql and noSql.
- Having knowledge on API(s) such as REST, SOAP and GraphQl

```css
/* Small scale devices */
@media only screen and (min-width: 320px) and (max-width: 767px){
    .child-1,
    .child-2{
        width: 90%;
    }
}
```

Cursors:
- Pointer
- Progress
- Help
- Move
- Zoom-in
- Zoom-out

Multi-column-layout
- Column-count
- column-width
- Column-rule-style
- Column-rule-color
- column-rule-width
- Column-rule

CSS & JAVASCRIPTS

Header

Image, description, Button			

Footer

Text-transform
- Uppercase
- Capitalize
- Lowercase
- None

2d transform
- translate(*x-axis, y-axis*)
- rotate()
- Skew(*x-axis, y-axis*) = skewX, skewY
- Scale(*x-axis, y-axis*) = scaleX, scaleY
- matrix(*scaleX(),skewY(),skewX(),scaleY(),translateX(),translateY()*)

JavaScript - variables
- Number
- String
- Boolean
- Null
- Object
- Function
- Undefined
- Not defined

Conversion statements
- Number()
- parseInt()
- parseFloat()

Console statements
- console.log()
- console.warn()
- console.error()
- console.assert()
- console.count()

CSS & JAVASCRIPTS

- console.info()

Selection statements
- If
  - Simple if
  - If-else
  - Else-if ladder
  - Nested if
- Switch

Iteration statements
- While
- Do-while
- For

Pop Ups
- alert()
- prompt()
- confirm()

Function: (**CodeLink**)
- Subprogram design to perform the particular task

Syntax:
        Function <functionname>(parameter){
                <Our task>
                Return ()
                }

        functionname() // if we execute the function
//it's nothing but invoking

1. Function declaration
2. Function Expression
   a. **Var x=function(){
      <Statements>
      }
      x() //invoking**

3.   Arrow Function:(ES6)
a.   Syntax:
Var fname = ( ) =>{
      <statements>
}
4.   Higher Order function:(filter,map,Foreach, etc.,)
5.   Nested function
Function parent () {

CSS & JAVASCRIPTS

Function child(){

}
}

DOM:(Document Object Model):
- document.write()
- document.getElementsByTagName()
- document.getElementsByClassName()
- document.getElementById()
- textContent
- innerHTML

Array iteration
- for()
- map()
- for-in()
- for-of()
- forEach()

Array functions
- Array.isArray() => boolean values
  - True => the parameter is an array
  - False = > the parameter is not an array
- Slice()=> Accepts two parameters (startingPosition, endingPosition)
- find() => Returns a value which is match to a specific condition
- Filter() => Similar to find(). But we can get multiple values.
- includes()
- pop()
- push()
- shift()
- unshift()
- Delete
- splice()
- indexOf()
- sort()
- reverse()

String

Character array => "Hello world"
String functions
- Concat()
- indexOf()
- replace()
- search()
- substr()
- substring()
- split()
- startsWith()
- endsWith()
- toUpperCase()
- toLowerCase()

Date:

CSS & JAVASCRIPTS

- Date()
- getFullYear()
- getMonth()
- getDate()
- getHours()
- getMinutes()
- getSeconds()
- getMilliseconds()
- getTime()

Regular Expressions:
// or RegExp()
- test()
- Special Characters
  - /s
  - /S
  - /d
  - /D
  - /w
  - /W
- Set [ ] => 0-9, a-z, A-Z
- {count}

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta
charset="UTF-8">
    <meta
name="viewport"
content="width=dev
ice-width,
initial-
scale=1.0">
    <title>Document
</title>
    <link
rel="stylesheet"
href="css/style.cs
s" />
    <script
src="js/jQuery.js"
></script>
</head>
<body>
  <form>
      <input
type="number"
placeholder="Mobil
e number"
onkeyup="checkMobi
leNumber()"
id="mobile"/> <br>
<br>
```

```javascript
function checkMobileNumber(){
    var
mobileNumber=document.getElementById("mobile").val
ue;
    var reg_mobile=/^[6-9]{1}[0-9]{9}$/;
    if(reg_mobile.test(mobileNumber)){
        document.getElementById("message").clas
sList.remove("errorMessage")
        document.getElementById("message").clas
sList.add("successMessage")
        document.getElementById("message").text
Content="Valid mobile Number"
    } else {
        document.getElementById("message").clas
sList.remove("successMessage")
        document.getElementById("message").clas
sList.add("errorMessage");
        document.getElementById("message").text
Content="Invalid mobile Number";
    }

  }
```

CSS & JAVASCRIPTS

```
    <span
id="message"></spa
n>
    </form>
</body>
</html>
```

Window:
- Alert
- History
- Length, frames
- innerHeight
- innerWidth,
- localStorage
- outerWidth
- OuterHeight
- close()

Screen
- Height
- availHeight
- Width
- availWidth
- Colordepth

Location properties
- Href
- Host
- Hostname
- Protocol
- Pathname

Location functions
- reload()
- replace()
- assign()

History
- back()
- forward()

Timer functions
- setTimeout()
- setInterval(0
- clearInterval()

Why do we need to use react?
- Virtual DOM
- JSX
- Component
- Routing

CSS & JAVASCRIPTS

- Performance
-