Name: Asif Mohtadi Ahmed
Id: 161-15-7000

1. Write short notes on HTPPS vs HTTP

HTTPS:

HTTPS stands for Hypertext Transfer Protocol Secure. It is the protocol where encrypted HTTP data is transferred over a secure connection. By using secure connection such as Transport Layer Security or Secure Sockets Layer, the privacy and integrity of data are maintained and authentication of websites is also validated.

HTTP:

Full form of HTTP is Hypertext Transfer Protocol. HTTP offers set of rules and standards which govern how any information can be transmitted on the World Wide Web. HTTP provides standard rules for web browsers & servers to communicate.HTTP is an application layer network protocol which is built on top of TCP. HTTP uses Hypertext structured text which establishes the logical link between nodes containing text. It is also known as "stateless protocol" as each command is executed separately, without using reference of previous run command.

1. Why is RFC required?

An Internet standard is published in a document known as a Request for Comments, or RFC. When a document is accepted for publication, it is assigned an RFC number by the IETF. The RFC is then published. After it is published, an RFC is never changed.

An RFC (Request for Comments) is a pure technical document published by the Internet Engineering Task Force (IETF). Request for Comments (RFCs) are mainly used to develop a "standard" network protocol, a function of a network protocol or any feature which is related with network communication.

2. CSS doesn’t require any standard - Justify.

Before writing any code, it is generally a good idea to have some ground rules about writing code. CSS developers should respect coding standard—it is essential for maintainability and scalability of the project.

Choose which naming convention to use throughout the project. Establishing a naming convention early on can help developers produce better and more organized code. It can also help everyone involved in the

project understand the component structure and relationship between components and elements by reading the HTML code alone.

Decide how to handle indentation, selector types, shorthand properties, units in CSS. For example, avoid using pixel units if the coding standard proposes the usage of rem units. Everyone has a distinct writing/coding style, but as a professional, you have to be able to adopt, and more importantly, understand every concept.

3. JavaScript is essentially a programming language - provide your view.

JavaScript is essentially a programming language that simplifies the building and designing of websites, while also ensuring they're tooled up to be as engaging and attractive as possible. It shouldn't be confused with Java, however, which is a completely separate programming language.
JavaScript was developed at Netscape. It was originally called Live Script, but that name wasn't confusing enough. The -Script suffix suggests that it is not a real programming language, that a scripting language is less than a programming language. But it is really a matter of specialization.

As its name implies, JavaScript is a scripting language. Traditional languages such as C++ are compiled before they're run into executable binary form, with the compiler checking for any errors in the entire program before the process is complete.

4. Design the following layout using HTML and CSS:

```
<!DOCTYPE html>
<html lang="en">

<head>

    <title>Document</title>

    <style>

        body{
            background-color: white;
            display: flex;
            justify-content: center;
        }

        .head{
            display: block;
```

```
    width: 1000px;
    height: 900px;
    background-color: white;
    padding: 10px;

}

#text{
    color: white;
    padding: 10px;
    font-size: 25px;

}
#text1{
    color: white;
    padding: 10px;
    font-size: 35px;
}


.first{
    background-color:gainsboro;
    display: flex;
    align-items: center;
    height: 60px;
    width: 250px;
    margin: 8px;
}
.second{
    background-color:gainsboro;
    display: flex;
    height: 40px;
    margin: 8px;
    align-items: center;
}
.third{
    background-color:gainsboro;
    display: flex;
    height: 100px;
    align-items: center;
    margin: 8px;
}
.four{
    display: flex;
    height: 600px;
    flex-direction: row;


}
```

```
        .section-1{
            background-color:gainsboro;
            display: flex;
            margin: 5px;
            width: 200px;

        }
        .section-2{
            background-color:gainsboro;
            display: flex;
            margin: 5px;
            width: 800px;
        }

    </style>

</head>

<body>

    <header class="head">

        <div class="first">
            <h3 id="text1">Logo</h3>
        </div>

        <div class="second">
            <h3 id="text">Menu</h3>
        </div>

        <div class="third">
            <h3 id="text">Header panel</h3>
        </div>

        <div class="four">

            <div class="section-1">
                <h3 id="text">Submenu</h3>
            </div>

            <div class="section-2">
                <h3 id="text" >Content</h3>
            </div>
        </div>

    </header>

</body>

</html>
```

OutPut: