|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2021\_1\_WEB\_07\_과제 | 학번 : | 20165164 | 이름 : | 지현한 |

* 개념 확인

1. 함수 선언 시 사용하는 키워드는 무엇인가?
2. 아래의 함수를 동일한 기능을 수행하는 가장 간단한 형태의 화살표 함수로 정의하시오.

function arrow(a,b){

return a\*b;

}

;

**let arrow = (a,b) => a\*b;**

1. 변수 sum에 저장된 값은 무엇인가?

function defaultPara(a, b=20, c=30){

            return a+b+c;

  }

sum=defaultPara(3, 6);

**39**

1. 매개변수로 받은 값 중에서 마지막 값만 출력하는 함수를 정의하고 호출하는 코드에서 밑줄 친 부분을 채워 넣으시오

let hap = function(\_\_\_\_**...value**\_\_\_\_\_\_\_\_\_\_\_\_\_ ){

    document.write(value[\_\_**value.legnth-1**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]) ;

}

1. 자바 스크립트에서 함수를 정의하는 방법에 대하여 설명하시오

**기명함수: 함수 선언 시 function 키워드 사용**

**익명함수: 이름을 붙이지 않고 함수 선언**

**화살표 함수: 자바의 람다식과 유사한 함수**

**즉시 실행 함수: 함수를 선언함과 동시에 실행**

1. 콜백 함수에 대하여 설명하시오

**매개변수로 전달하는 함수**

1. 클로저에 대하여 설명하시오

**함수가 종료된 후에도 해당 지역변수를 사용할 수 있도록 남겨두는 현상**

1. 자바 스크립트 작성 위치에 대하여 설명하시오

**내부 자바 스크립트: <head>태그나 <body>태그 사이에 <script>태그로 작성**

**인라인 자바 스크립트: HTML태그 안에 속성값으로 정의**

**외부 자바 스크립트: <script> 태그의 src 속성 값으로 외부 자바 스크립트 파일명 작성**

1. 자바 스크립트 함수 특징에 대하여 설명하시오

**무명의 리터럴로 표현가능**

**변수나 자료구조에 저장가능**

**함수의 파라미터로 전달 가능**

**반환 값으로 사용 가능**

* 개념 활용

1. 다음과 같이 메뉴를 화면 하단에 고정시키는 css를 작성하시오. 단, 메뉴외에 출력되는 내용은 임의로 작성

<body>

    <h1>Lorem ipsum dolor sit amet</h1>

    <p>

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

    </p>

    <div class="menu">

       <ul>

            <li>JSP</li>

            <li>JSON</li>

            <li>jQuery</li>

            <li>HTML</li>

            <li>CSS3</li>

        </ul>

     </div>

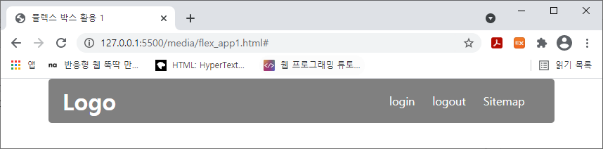
</body>

|  |
| --- |
| [소스]  <style>          ul{              position: fixed;              background: gray;              display: flex;              width: 600px;              font-size:20px;              justify-content: space-evenly;              position:fixed;              right: 10px;              bottom: 0;              border-radius: 5px;          }          li{              list-style: none;              padding:2px 10px;          }      </style> |
| [실행 결과] |

1. 플렉스 박스를 사용하여 다음과 같은 레이아웃을 구성할 수 있도록 css 를 작성하시오.

<body>

    <header>

        <h1><a href="#">Logo</a></h1>

        <nav>

            <ul>

                <li><a href="#">login</a></li>

                <li><a href="#">logout</a></li>

                <li><a href="#">Sitemap</a></li>

            </ul>

        </nav>

    </header>

</body>

|  |
| --- |
| [소스]  <style>          header {              display: flex;              background: gray;              padding: 0 20px;              margin: 0 50px;              border-radius: 5px;              justify-content: space-between;          }          h1 {              display: inline;              margin: 0;          }          li {              display: inline;              padding: 0 10px;          }          a {              text-decoration: none;              color: white;          }      </style> |
| [실행 결과] |

1. 플렉스 박스를 사용하여 다음과 같은 레이아웃을 구성하고 반응형 웹이 될 수 있도록 css 를 작성하시오

<body>

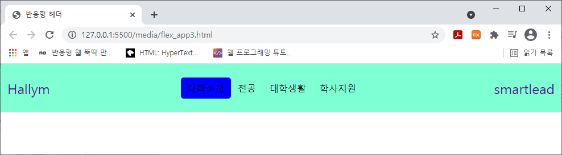
    <nav class="menu\_bar">

        <div class="menu\_left">

           <a href="">Hallym</a>

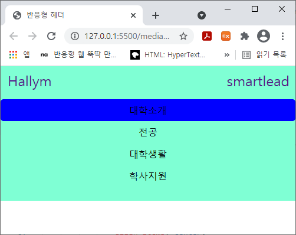
        </div>

        <ul class="menu\_item">

            <li>대학소개</li>

            <li>전공</li>

            <li>대학생활</li>

            <li>학사지원</li>

        </ul>

        <div class="menu\_right">

            <a href="">smartlead</a>

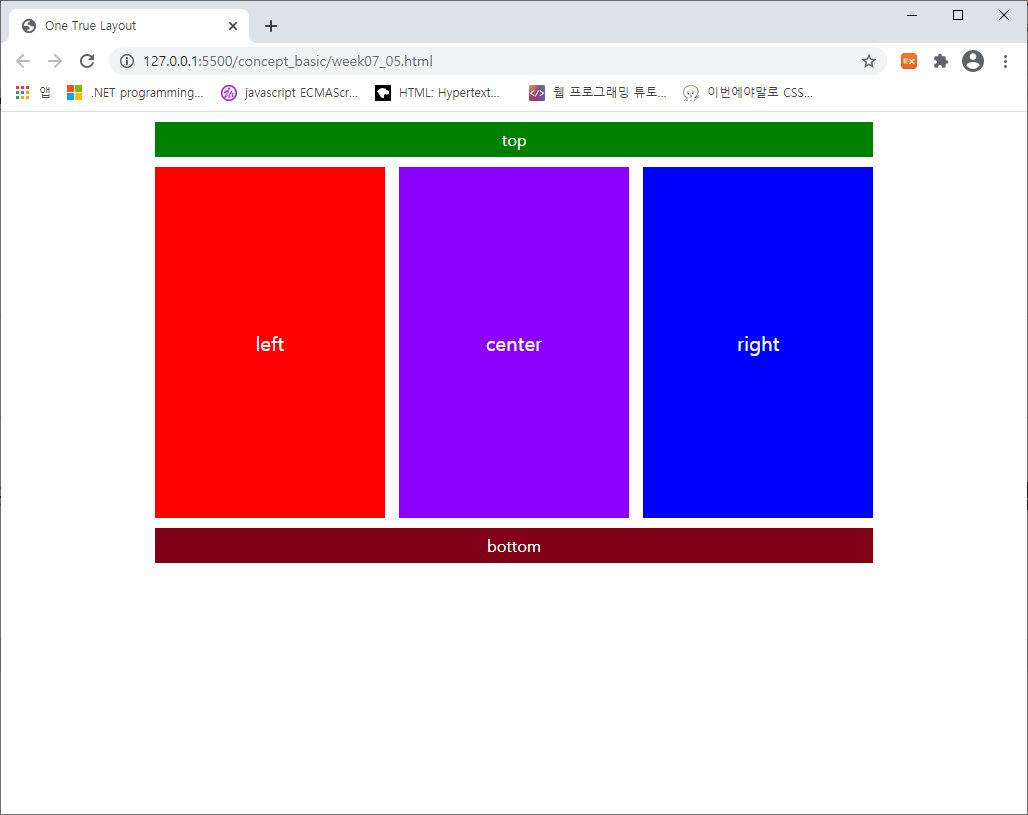
        </div>

    </nav>

</body>

|  |
| --- |
| [소스]  <style>          nav {              display: flex;              height: 50px;              background: mediumspringgreen;          }          div {              position: fixed;              margin: 10px;              padding: 10px;              top: 0;          }          .menu\_left {              left: 0;          }          .menu\_right {              right: 0;          }          ul {              display: flex;              width: 400px;              margin: 0 auto;              padding: 5px;              position: relative;              top: 10px;          }          li {              display: inline-block;              margin: 0 20px;          }          li:hover {              background: blue;          }          @media (max-width: 599px) {              ul {                  flex-direction: column;                  width: 600px;                  margin: 40px 0;                  padding: 0;              }              li {                  background: mediumspringgreen;                  margin: 0;                  padding: 5px 0;                  text-align: center;              }          }      </style> |
| [실행 결과] |
|  |

1. 플렉스 박스를 사용하여 다음과 같은 One True 레이아웃을 작성하시오.



<style>

body {

      width: 70vw;

      margin: 10px auto;

      color: white;

      text-align: center;

    }

</style>

<body>

  <div id="top">top</div>

  <div id="middle">

    <div class="item" id="left">left</div>

    <div class="item" id="center">center</div>

    <div class="item" id="right">right</div>

  </div>

  <div id="bottom">bottom</div>

</body>

|  |
| --- |
| [프로그램 소스]  <style>          body {              width: 70vw;              margin: 10px auto;              color: white;              text-align: center;          }          #top {              background: green;          }          #middle {              display: flex;              justify-content: space-between;              height: 300px;              margin: 10px 0;          }          .item {              width: 31%;              line-height: 300px;          }          #left {              background: red;          }          #center {              background: purple;          }          #right {              background: blue;          }          #bottom {              background: brown;          }      </style> |
| [실행 결과] |