

## □ 개념 확인

(1) 플렉스 박스 컨테이너에 적용하는 속성에 대하여 설명하시오.

- flex-direction은 flex 컨테이너의 주축 방향을 설정한다.
- flex-wrap은 컨테이너보다 아이템의 너비의 합계가 더 큰 경우의 표현 방식을 지정한다.
- flex-flow는 flex-direction과 flex-wrap 속성을 동시에 설정한다.
- justify-content는 컨테이너의 중심축을 기준으로 정렬하는 방식을 지정한다.
- align-items는 아이템을 수직 방향으로 정렬하는 장식을 지정한다.
- align-content는 아이템들의 행이 2줄 이상 되었을 때 컨테이너의 수직 축을 기준으로 아이템을 수직 정렬한다.

(2) 플렉스 박스 아이템에 적용하는 속성에 대하여 설명하시오.

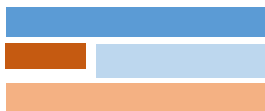
- flex-grow는 컨테이너에 여백이 있을 때 확대 인자를 지정해 아이템의 크기를 키운다.
- flex-shrink는 컨테이너안의 아이템의 크기가 넘칠 경우 너비에 대한 축소인자를 지정한다.
- flex-basis는 아이템의 공간 배분 전 너비의 기본 값을 지정한다.
- flex는 flex-grow, flex-shrink, flex-basis를 한번에 지정한다.
- align-self는 align-items보다 우선하여 개별 정렬한다.
- order은 아이템의 배치 순서를 지정한다.

(3) 미디어 쿼리를 적용하는 두가지 방법을 설명하시오.

@media 규칙을 사용하여 적용한다. => @media (<미디어 쿼리>) { <CSS 코드> }

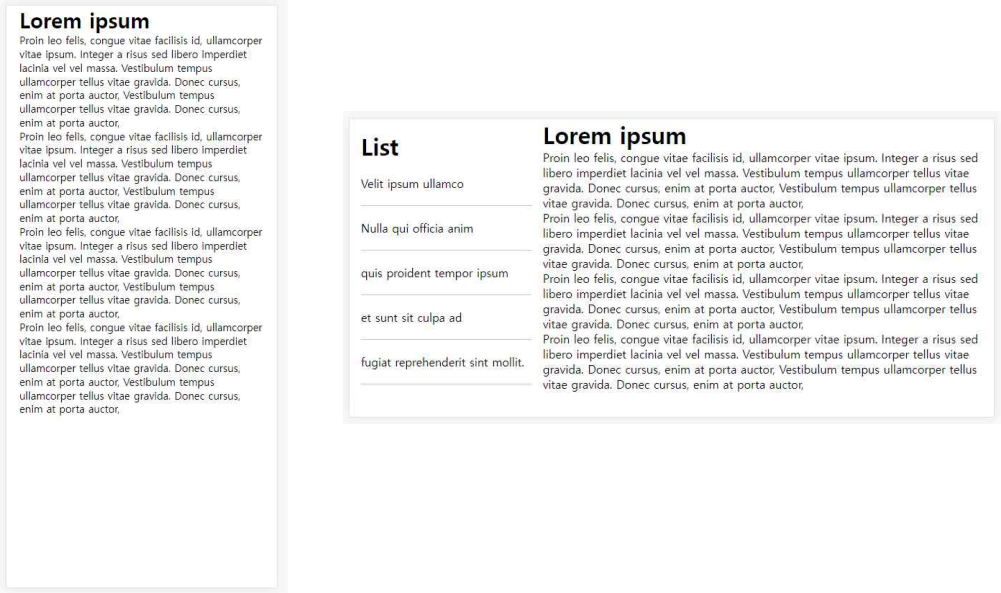
media속성을 사용하여 적용한다. => <link rel="stylesheet" href="<파일이름>" media="<미디어 쿼리>">

(4) 다음과 같은 One True 레이아웃을 구성하는 과정에 대하여 설명하시오.



- 3 개의 행이 있으니 div 로 공간을 3 개로 분할한 뒤, 중간 공간을 열 2 개로 분할한다.  
마지막으로 부모 태그에 고정된 너비를 입력하고, 부모 태그의 overflow 속성에 hidden 을,  
자손 태그엔 적당한 너비를 입력한 뒤 float 속성을 적용하는 레이아웃을 구성한다.

(5) 스마트폰 회전을 사용한 레이아웃을 만들어 보시오. 아이폰의 기본 메시지 앱은 다음과 같이 세로 상태에서는 메시지 목록과 내용이 별도의 페이지로 되어있다. 가로 상태에서는 메시지 목록과 내용을 한 화면에 표시할 수 있도록 css 코드를 작성하시오. 힌트)플렉스 박스와 미디어 쿼리 사용



다음 순서로 레이아웃을 구성하시오. 제시된 레이아웃 외에는 본인이 작성

초기화	<ol style="list-style-type: none"> <li>① 모든 태그(*)의 margin 속성과 padding 속성을 0으로 지정합니다.</li> <li>② li 태그의 list-style-type을 none으로 지정합니다.</li> </ol>
반응형 웹	<ol style="list-style-type: none"> <li>① 세로 상태에서는 div.left 태그가 보이지 않게 만듭니다.</li> <li>② 가로 상태에서는 div.left 태그가 너비의 30%를 차지하고, 왼쪽에 붙게 만듭니다.</li> <li>③ 가로 상태에서는 div.right 태그가 너비의 70%를 차지하고 오른쪽에 붙게 만듭니다.</li> </ol>
기본 콘텐츠 구성	<ol style="list-style-type: none"> <li>① div.content 태그의 padding을 16px로 지정합니다.</li> <li>② li 태그의 위아래 padding 속성을 24px로 지정합니다.</li> <li>③ li 태그의 아래쪽 테두리를 1px solid #c7c7c7로 지정합니다.</li> </ol>

<body>  
<div>

```

class="container">
  <div class="left">
    <div class="content">
      <h1>List</h1>
      <ul>
        <li>Velit ipsum ullamco</li>
        <li>Nulla qui officia anim</li>
        <li>quis proident tempor ipsum</li>
        <li>et sunt sit culpa ad</li>
        <li>fugiat reprehenderit sint mollit.</li>
      </ul>
    </div>
  </div>
  <div class="right">
    <div class="content">

```

```
<h1>Lorem ipsum</h1>
<p>Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet
    lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
    Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
</p>
<p>Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet
    lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
    Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
</p>
<p>Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet
    lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
    Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
</p>
<p>Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet
    lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
    Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,
</p>
</div>
</div>
</body>
```

---

[소스 – css만 제시 하세요]

```
*{
    margin: 0;
    padding: 0;
}

li{
    list-style-type: none;
    padding: 24px 0;
    border-bottom: 1px solid #c7c7c7;
}

@media screen and (orientation: portrait) {
    div.left{
        display: none;
    }
}

@media screen and (orientation: landscape) {
    div.left{
        position: absolute;
        width: 30%;
        left: 5px;
    }

    div.right{
        position: absolute;
        width: 70%;
        right: 5px;
    }
}
```

```
    }  
  }  
  
  div.container{  
    position: relative;  
  }  
  
  div.content{  
    padding: 16px;  
  }
```

```
  *{  
    margin: 0;  
    padding: 0;  
  }  
  
  li{  
    list-style-type: none;  
    padding: 24px 0;  
    border-bottom: 1px solid #c7c7c7;  
  }  
  
  @media screen and (orientation: portrait) {  
    div.left{  
      display: none;  
    }  
  }  
  
  @media screen and (orientation: landscape) {  
    div.left{  
      flex-basis: 30%;  
    }  
  
    div.right{  
      flex-basis: 70%;  
    }  
  }  
  
  div.container{  
    display: flex;  
    justify-content: space-between;  
  }  
  
  div.content{  
    padding: 16px;  
  }
```

Proin leu felis, congue vitae facilis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Proin leu felis, congue vitae facilis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Proin leu felis, congue vitae facilis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Proin leu felis, congue vitae facilis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,

Velit ipsum ullamco

Nulla qui officia anim

quis proident tempor ipsum

et sunt sit culpa ad

fugiat reprehenderit sint

Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Proin leo felis, congue vitae facilisis id, ullamcorper vitae ipsum. Integer a risus sed libero imperdiet lacinia vel vel massa. Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor, Vestibulum tempus ullamcorper tellus vitae gravida. Donec cursus, enim at porta auctor,

<body>

# <a href="#">Logo</a> </h1>



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

&lt;/header&gt;

&lt;/body&gt;



[소스 - CSS만 제시하시오]

```

*{
  margin: 0;
}

header{
  width: 80vw;
  height: 60px;
  display: flex;
  justify-content: space-between;
  align-items: center;
  background-color: gray;
  margin: 0 auto;
}

a{

```

```

text-decoration: none;
color: ghostwhite;
padding: 10px;
}

li{
  line-height: 60px;
padding: 10px;
display: inline-block;
list-style-type: none;
}

```

[실행 결과]



## □ 개념 활용 응용 프로그래밍

(1) 제시된 결과처럼 출력될 수 있도록 주어진 html에 대하여 css코드를 작성하시오. 단, 요소 선택 시 클래스 애트리뷰트가 있으면 클래스 선택자를, 아이디 애트리뷰트가 있으면 아이디 선택자를 사용한다.



<body>

<!-- [문제 3-1번] -->

<nav>

<ul class="menu">

<li>로그인</li>

<li>로그아웃</li>

<li>회원가입</li>

</ul>

</nav>

<!-- [문제 3-2번] -->

<section>

<!-- [문제 3-3번] -->

<div>

<table>

<tr><td>월</td>

<td>제목</td>

<td>비고</td></tr>

```

        <tr><td>3 월</td>
        <td>아기공룡둘리</td>
        <td>애니메이션</td></tr>
        <tr><td>4 월</td>
        <td>코다</td>
        <td>남우주연상</td></tr>
    </table>
    <ul id="check">
        <li>점검사항
            <ol>
                <li>창문 닫기</li>
                <li>냉난방 전원 끄기</li>
            </ol>
        </li>
    </ul>
</div>

```

```

<!-- [문제 3-4번] -->

```

```

<div class="footer">
    <em>춘천시 한림대학길 1 정보과학대학</em>
</div>
</section>
</body>
</html>

```

#### ■ 3-1번 :

1. 메뉴는 제시된 결과처럼 화면 위쪽에 고정되어 있으며, 고정 위치는 왼쪽 10px
2. 마우스 커서가 메뉴 영역 위에 있으면 전체 메뉴가 표시 된다
3. 메뉴 항목에 마우스가 위치하면 텍스트에 밑줄을 표시한다.
4. 메뉴 항목의 글자 색은 모두 다르게 한다.
5. 메뉴 항목의 너비는 100px, 안쪽 여백은 위/아래 2px, 왼쪽/오른쪽 10px
6. 플로팅 속성 사용 불가
7. 전체 margin은 auto로 한다.

#### ■ [문제 3-2번]

#### ■ [문제 3-3번]

1. 제시된 결과를 참고하여 출력과 관련된 속성만 작성
2. 표 너비와 높이는 300px, 90px
3. 점검사항의 경우 너비와 높이는 표와 동일하며 바깥쪽 여백 위/아래는 0px, 왼쪽/오른쪽 10px
4. 점검사항의 경우 안쪽여백은 위쪽만 10px, 배경색과 모서리 곡률은 임의로 정 할 것

#### ■ [문제 3-4번]

1. footer 요소는 부모를 기준으로 배치위치는 아래쪽 10px, 왼쪽은 뷰포트를 기준으로 15
2. 수직 가운데 정렬이 되도록 한다

3. 배경색과 너비와 높이는 임의로 설정하고 안쪽 여백은 왼쪽만 50px

[소스]

```
.menu{
  position: fixed;
  left: 10px;
  top: -35px;
  width: 400px;
  height: 50px;
  margin: auto;
  background-color: mintcream;
}

.menu:hover{
  top: 5px;
}

.menu li{
  display: inline-block;
  text-align: center;
  line-height: 50px;
  list-style: none;
  width: 100px;
  padding: 2px 10px;
  color: red;
}

.menu li:nth-child(2){
  color: blue;
}

.menu li:nth-child(3){
  color: green;
}

.menu li:hover{
  text-decoration: underline;
}

section{
  position: relative;
  top: 60px;
  margin: 0 30px;
  height: 90vh;
}

#check, table{
  right: 15vw;
  position: absolute;
  width: 300px;
  height: 90px;
  margin: 0 10px;
  padding: 10px 0 0;
}
```



```

border-collapse: collapse;
}

tr, td{
border-bottom: 2px solid blue;
}

#check{
left: 15vw;
border-radius: 10px;
background-color: aquamarine;
padding-left: 18px;
}

#check li{
list-style-position: inside;
}

.footer{
height: 40px;
width: 300px;
line-height: 40px;
position: absolute;
bottom: 10px;
left: 15vw;
padding-left: 50px ;
background-color: orange;
}

em{
text-align: center;
}

```

## [실행 결과]



(2) 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]

```
body{
  margin: 0;
}

a{
  text-decoration: none;
  color: blue;
}

.menu_bar{
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 0 12px;
  background-color: aquamarine;
}

.menu_item{
  display: flex;
  list-style: none;
  padding-left: 0;
}

.menu_item li{
  padding: 8px 8px;
}
```

```

.menu_item li:hover{
  background-color: blue;
  border-radius: 8px;
}

@media screen and (max-width : 750px){
  .menu_bar{
    flex-direction: column;
    align-items: flex-start;
  }

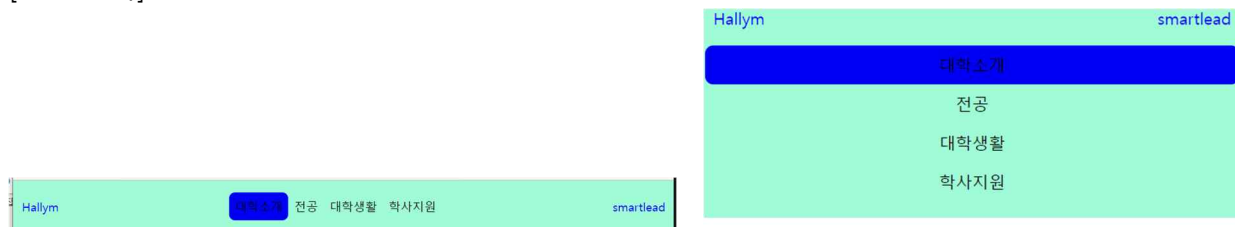
  .menu_item{
    width: 100%;
    flex-direction: column;
    align-items: center;
  }

  .menu_item li{
    width: 100%;
    text-align: center;
  }

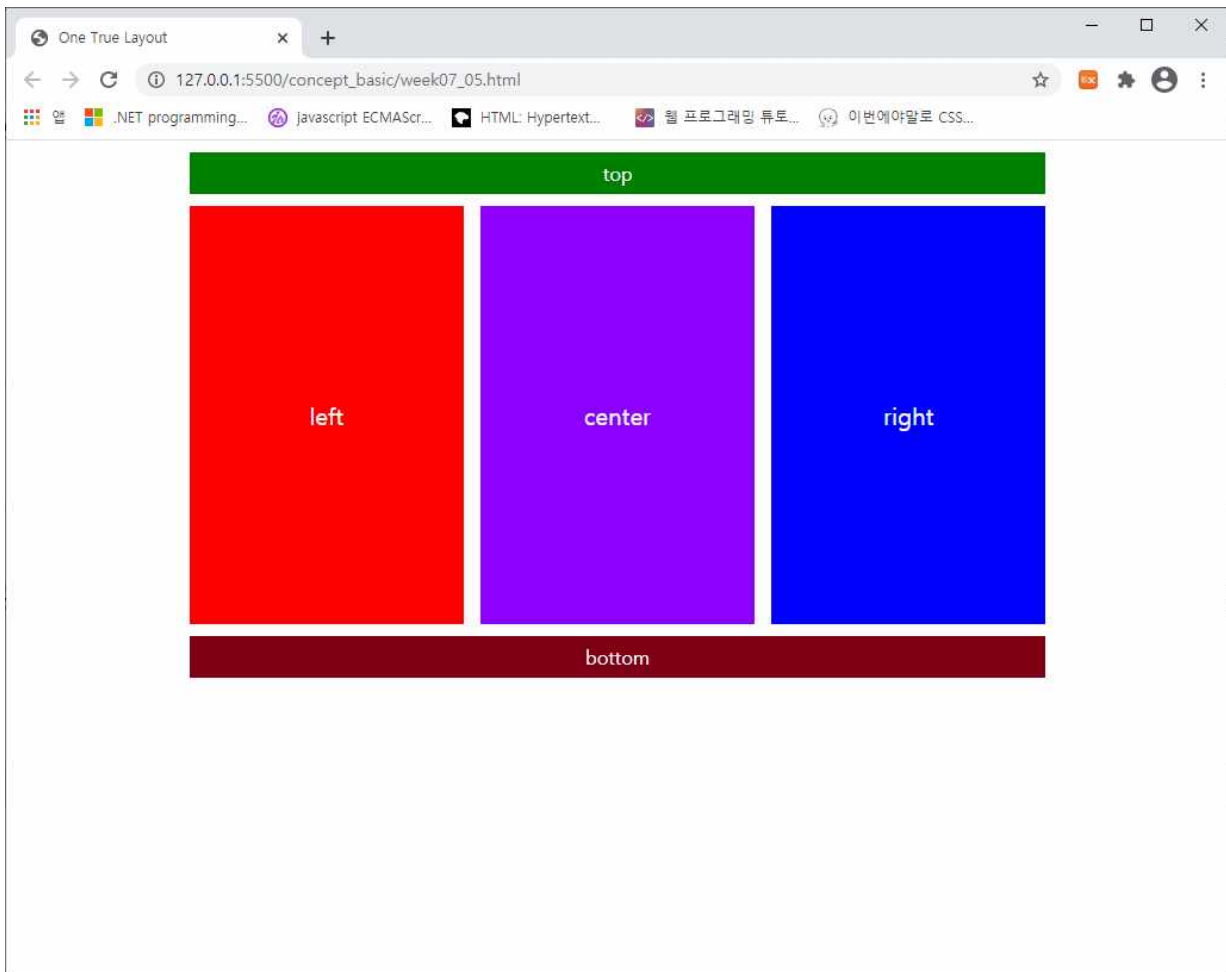
  .menu_right{
    position: absolute;
    right: 12px;
  }
}

```

[실행 결과]



(3) 플렉스 박스를 사용하여 다음과 같은 One True 레이아웃을 작성하시오.



```
<script>
  body {
    width: 70vw;
    margin: 10px auto;
    color: white;
    text-align: center;
  }
</script>
<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>
```

[소스]

```
body {
  width: 70vw;
  margin: 10px auto;
  color: white;
```

```
    text-align: center;
}

#top{
    background-color: green;
    padding: 10px;
}

#middle{
    overflow: hidden;
    display: flex;
    justify-content: space-between;
}

.item{
    margin: 10px;
    height: 40vh;
    line-height: 40vh;
}

#left{
    background-color: red;
    flex: auto;
    margin-left: 0;
}

#center{
    background-color: blueviolet;
    flex: auto;
}

#right{
    background-color: blue;
    flex: auto;
    margin-right: 0;
}

#bottom{
    background-color: brown;
    padding: 10px;
}
```

[실행 결과]

