

補講課題2

次の画面を作成しなさい、

Web page title (header)

this section is nav.
福岡タワー
糸島
小倉城
柳田神社
宮地嶽神社

this section is main.
sample 1 title



福岡タワーは、福岡市早良区のシーサイドももち地区にある高さ234mの海浜タワーで、海浜タワーとしては日本一の高さ、福岡のシンボルです。地上123mの展望室からは 福岡市街や博多湾の360度パノラマが楽しめ、夜景も「夜景100選」に選ばれるほど絶景で、季節ごとのイルミネーションも人気、カフェやレストランも併設されています。1989年のアジア太平洋博覧会(よかトピア)のモニュメントとして建設され、「ミラーセイル」の愛称で親しまれています。VSCodeで長文の折り返しについては、Alt+Zで折りたたむ。

this section is footer.
copyright 2025 (c) CKGMC

ソースコード(index.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>
        /* 全てのarticle要素に対しての設定 */
        article{
            /* 非表示の設定 */
            display: none;
        }
        /* 最初の要素の設定；ここまで全てのarticle要素が非表示になった前提 */
        article:nth-child(1){
            /* 表示の設定 */
            display: block;
        }
    </style>
    <link rel="stylesheet" href="common.css">
</head>
<body>

    <header>
        <h1>Web page title ( header ) </h1>
    </header>
```

```
<nav>
    this section is nav.

    <ul>
        <!--
        リストについては、できるだけ揃えて記述する。
        特にそれぞれの色を変えたいときには、cssサフィックスnt-child()を使うと名前を省略できる
        あえて生を付けるときはid属性
        -->
        <li>福岡タワー</li>
        <li>糸島</li>
        <li>小倉城</li>
        <li>柳田神社</li>
        <li>宮地嶽神社</li>

    </ul>
</nav>

<main>
    this section is main.

    <article id="sample1">
        <h2>sample 1 title</h2>
        <!-- 画像とテキストはひとまとめて扱うために、divタグでグループ化 -->
        <div>
            <!-- 画像を入れるのはimgタグ -->
            <!-- 画像のパスがフォルダに入っているときは、「.」から打ち始めるとよい -->
            
            <!-- 画像とセットになる文章はここに入れる -->
            <p>
                福岡タワーは、福岡市早良区のシーサイドももち地区にある高さ234mの海浜タワーで、
                海浜タワーとしては日本一の高さ、福岡のシンボルです。地上123mの展望室からは
                福岡市街や博多湾の360度パノラマが楽しめ、夜景も「夜景100選」に選ばれるほど絶景で、
                季節ごとのイルミネーションも人気、カフェやレストランも併設されています。
                1989年のアジア太平洋博覧会(よかトピア)のモニュメントとして建設され、
                「ミラーセイル」の愛称で親しまれています。
                VSCodeで長文の折り返しについては、Alt+Zで折りたたむ。
            </p>
        </div>

    </article>
    <article id="sample2">
        <h2>sample 2 title</h2>
        <div>
            
            <p>
                </p>
        </div>

    </article>
    <article id="sample3">
        <h2>sample 3 title</h2>
        <div>
            
            <p>
                </p>
        </div>

    </article>

```

```

<article id="sample4">
  <h2>sample 4 title</h2>
  <div>
    
    <p>

    </p>
  </div>

</article>
<article id="sample5">
  <h2>sample 5 title</h2>
  <div>
    
    <p>

    </p>
  </div>

</article>

</main>

<footer>
  this section is footer.<br>
  <small>copyright 2025 (c) CKGMC </small>
</footer>
</body>
</html>

```

ソースコード (common.css)

```

/* 全体 (*) のCSS設定を最初に書く */
*{
  padding: 0;
  margin: 0;

}

/* 定数の定義 */
/* 数字は、様々な部分でそのままの数字を埋め込むと、意味が分からなくなる。
コンピュータで自動生成する以外には、できるだけ定数化する。
基本的に定数は、セクション「:root」の項目に、「定数名:値;」の形で記述する。 */
:root{
  /* 単位%はブラウザの幅なども含めた割合 */
  --nav-ratio: 30%;
  /* 単位vhは、画面の見える範囲についての割合（%相当）
  高さを固定値で作りたいときには、px,exなどの単位を使用する。 */
  --main-vh: 80vh;
  --header-vh: 10vh;

}

/* タグに対する設定は、大きなタグの出てくる順番に書くと整理がしやすい */
body{
  /* bodyタグの子要素はサイズに応じて折り返し配置を行う設定 */
  display: flex;
  /* 横並びで配置し、全体合計が100%に収まらないときは下段に移すという折り返しの設定 */
  flex-wrap: wrap;
}

```

```
header, footer{
  width: 100%;
  background-color: antiquewhite;
}

header{
  /* 定数の参照には、var()を使用する */
  /* 下記では、「height: 10vh;」と同じ意味になる。 */
  height: var(--header-vh);
}

nav{
  width: var(--nav-ratio);
  background-color: bisque;
  height: var(--main-vh) ;
}

nav ul{
  /* 番号なし要素のリストの頭には既定値で「・」が表示されるが、これをなくすこと */
  list-style-type: none;
}

main{
  /* CSSのサイズの計算にはcalc()を使用する */
  /* 右端のナビゲーションのサイズをかけながらページを作るときには、
   * 計算させると他方を変更しなくてよい。 */
  /* calc()の引数で計算を行うが、その項は空白で区切る方がトラブルが少ない。 */
  width: calc( 100% - var( --nav-ratio) );
  background-color: lightblue;
  height: var(--main-vh);
}

main img{
  width: 40%;
}

footer{
  /* footerをヘッダーの高さとコンテンツの高さを引き算した値で求める。
   * この時のブラウザのコンテンツの高さは100vh */
  height: calc( 100vh - var( --header-vh) - var( --main-vh) );
}
```