



반응형 웹



개요

- 반응형 웹 디자인
 - 웹 사이트의 내용을 그대로 유지하면서 다양한 화면 크기에 맞게 웹 사이트를 표시하는 방법
 - 다양한 화면 크기의 모바일 기기들이 계속 쏟아져 나오는데 그 때마다 그 크기에 맞춘 사이트를 별도로 제작하는 것은 비효율적
 - 화면 크기에 반응해 화면 요소들을 자동으로 바꾸어 사이트를 구현하는 것이 바로 반응형 웹 디자인

개요

- 반응형 웹 디자인의 장단점
 - 장점
 - 모든 스마트 기기에서 접속 가능
 - 가로 모드에 맞춘 레이아웃 변경 가능
 - 사이트 유지, 관리 용이
 - 단점
 - 반응형 웹 기술이 최신 웹 표준인 **CSS3**의 일부
 - 최신 모던 웹 브라우저에서만 지원됨

3

개요

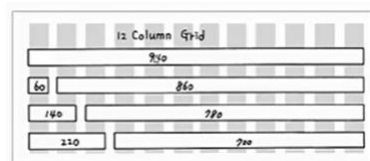
- 뷰포트 (viewport)
 - 실제 내용이 표시되는 영역
 - PC화면과 모바일 화면의 픽셀 표시 방법이 다르기 때문에 모바일 화면에서 의도한 대로 표시되지 않음
 - 뷰포트를 지정하면 기기 화면에 맞춰 확대/축소해서 내용 표시
- 뷰포트 지정하기
 - <head> 태그 안에서 <meta>태그를 이용해 뷰포트 지정
 - 형식 :
 - <meta name="viewport" content="width=device-width, initial-scale=1">
 - 일반적인 사용법 : 뷰포트의 너비를 스마트폰 화면 너비에 맞추고 초기 화면 배율을 1로 지정

속성	설명	사용 가능한 값	기본 값
width	뷰포트 너비	device-width 또는 표기	브라우저 기본 값
height	뷰포트 높이	device-height 또는 표기	브라우저 기본 값
user-scalable	확대/축소 가능 여부	yes 또는 no	yes
initial-scale	초기 확대/축소 값	1~10	1
minimum-scale	최소 확대/축소 값	0~10	0.25
maximum-scale	최대 확대/축소 값	0~10	1.6

4

개요

- 그리드 시스템 (grid system)
 - 화면을 여러 개의 칼럼으로 나누어 필요할 때마다 칼럼들을 묶어 배치하는 방법
 - 화면 너비 값에 따라 '960 그리드 시스템', '1200 그리드 시스템' 등으로 나눔
 - 칼럼 개수에 따라 '12칼럼 그리드 시스템', '16칼럼 그리드 시스템', '24칼럼 그리드 시스템' 등으로 나눔
 - 주로 960 픽셀 12칼럼의 그리드 시스템 사용
 - 고정 그리드 : 화면 너비를 일정하게 고정하고 레이아웃 만들기
 - 가변 그리드 : 화면 너비를 %같은 값으로 지정



5

개요

- 고정 그리드 레이아웃일 경우
 - 문서의 맨 바깥 부분을 #wrapper 요소로 묶고 너비를 960px로 지정
 - 해더와 본문, 사이드 바, 푸터를 배치, 이 때 너비는 px값
 - 화면 너비가 좁아질 경우 내용의 일부가 가려질 수 있음

```

<style>
#wrapper {
  width:960px;
  margin:0 auto;
}
header {
  width:960px;
  height:120px;
}
.content {
  float:left;
  width:600px;
}
.right-side {
  float:right;
  width:300px;
}
footer {
  clear:both;
  width:960px;
}
</style>

```

6



- 가변 그리드 레이아웃
 - 전체를 감싸는 요소의 너비를 %로 변환
 - 화면에 꼭 차게 하고 싶다면 100%
 - 전체를 감싸는 요소의 너비를 기준으로 각 요소의 너비를 계산

$$\text{요소의 너비} / \text{컨텐츠 전체를 감싸는 요소의 너비} * 100$$

```
<style>
#wrapper {
  width:96%;
  margin:0 auto;
}
header {
  width:100%;
  height:120px;
}
.content {
  float:left;
  width:62.5%;
  height:400px;
  padding:1.5625%;
  background-color:#ffd800;
}
```

```
.right-side {
  float:right;
  width:31.25%;
  height:400px;
  padding:1.5625%;
  background-color:#00ff90;
}
footer {
  clear:both;
  width:100%;
  height:120px;
  background-color:#c3590a;
}
</style>
```

7



가변 요소

- 가변 글꼴
 - em 단위
 - 부모 요소 폰트의 대문자 M 너비를 1em으로 지정, 1em = 16px

$$\text{글자크기}(em) = \frac{\text{글자크기}(px)}{16px}$$

```
<style>
.header-text{ font-size:2em; }
.content { font-size: 1.5em; }
.right-side { font-size: 1.5em; }
.footer { font-size: 1.5em; }
</style>
```

- rem 단위
 - em 단위는 부모 요소가 중첩될 경우 글자 크기가 계속 달라짐
 - Rem은 처음부터 기본 크기를 지정하고 그것을 기준으로 글자 크기 지정

```
<style>
body { font-size:16px; }
.header-text{ font-size:2rem; }
.fluid-text { font-size:1.5rem; }
</style>
```

8

가변 요소

- 가변 이미지
 - 브라우저 창의 너비가 변하더라도 이미지 너비 값은 변하지 않음
 - 브라우저 화면 너비를 줄일 경우 이미지 일부가 가려짐
 - 가변 이미지로 만들면 창의 너비에 따라 이미지 너비도 조절됨
- 1. CSS를 이용한 방법
 - 이미지를 감싸고 있는 부모 요소만큼만 커지거나 작아지도록 `max-width` 속성 값을 100%로 지정



9

가변 요소

- 2. `` 태그와 `srcset` 속성
화면 너비 값이나 픽셀 밀도에 따라 고해상도의 이미지 파일 지정 가능
- 3. `<picture>` 태그와 `<source>` 태그
화면 해상도 뿐만 아니라 화면 너비에 따라 다른 이미지 파일 표시

```
<picture>
  <source srcset="abc.jpg" media=(min_width:1024px)">
  <source srcset="test.jpg" media=(min_width:768px)">
  <source srcset="test2.jpg" media=(min_width:320px)">
  
</picture>
```

10

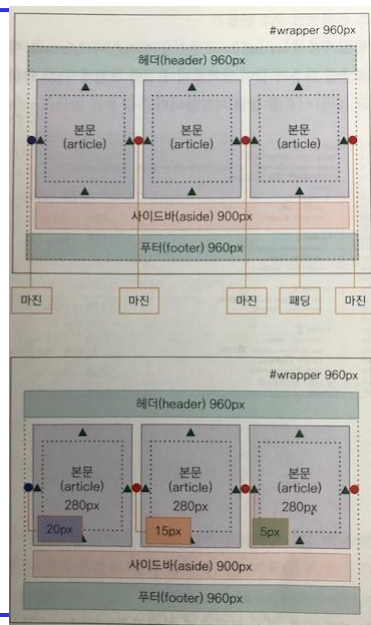
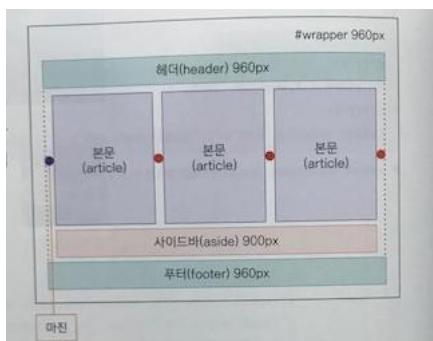
가변 요소

- 가변 비디오
 - CSS를 이용해 max-width 속성을 100%로 지정

11

가변 그리드 레이아웃 구성하기

- 구상하기





가변 그리드 레이아웃 구성하기

- 전체 구조 만들기

```
<style>
  #wrapper {
    width: 96%;
    margin: 0 auto;
  }
  header {
    background-color: #069;
    padding: 10px;
    overflow: hidden;
  }
  section {
    border: 5px solid #333;
    overflow: hidden;
  }
  footer {
    padding: 10px;
    background-color: #333;
    color: white;
    text-align: center;
  }
</style>
```

13



가변 그리드 레이아웃 구성하기

- 해더부분 코딩하기

```
<style>
...
header h1 {
  width: 40%;
  float: left;
  font-size: 2.5em;
  font-family: "돋움", "굴림";
  color: #fff;
}
header nav {
  width: 60%;
  float: right;
}
...
</style>
```

페이지의 제목 스타일

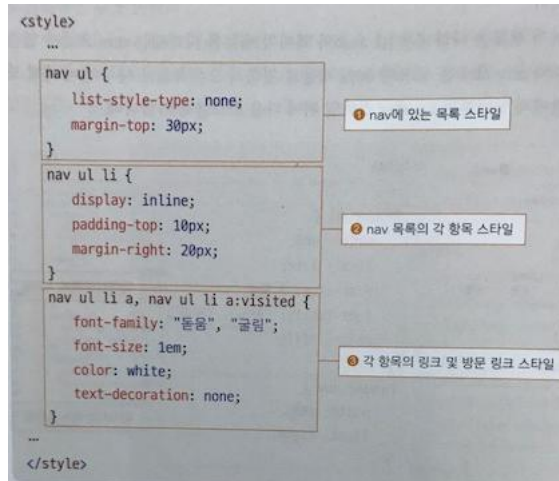
페이지의 메뉴 스타일

14



가변 그리드 레이아웃 구성하기

■ nav

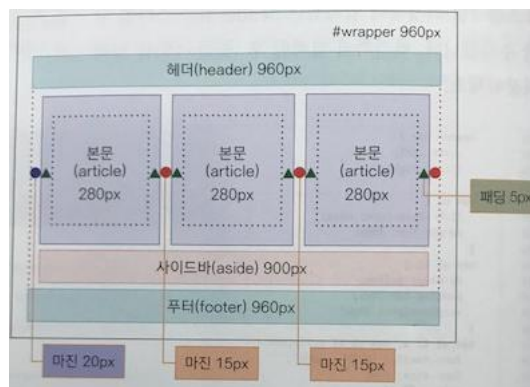


15



가변 그리드 레이아웃 구성하기

■ 콘텐츠 영역 너비 계산하기

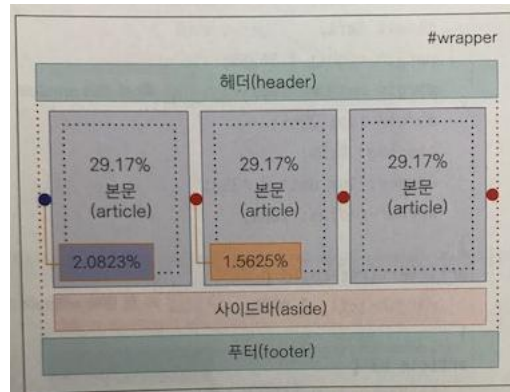


16



가변 그리드 레이아웃 구성하기

- 콘텐츠 영역 %로 계산 하기



17



가변 그리드 레이아웃 구성하기

```
<style>
...
article {
  width: 29.17%;
  height: auto;
  float: left;
  margin-right: 1.5625%;
  margin-bottom: 20px;
  margin-top: 20px;
  padding: 0.5%;
  border: 1px dotted #333;
  border-radius: 10px;
}
article:first-child {
  margin-left: 2.083%;
}
article h3 {
  background: url(images/note.png) no-repeat left center;
  padding-left: 30px;
}
dl {
  margin: 10px;
}
dt {
  font-weight: bold;
}
...
</style>
```

① 세 개의 article에 대한 스타일

② 첫 번째 article에 대한 스타일

③ 각 article의 소제목에 대한 스타일

④ article 안의 목록에 대한 스타일

18



가변 그리드 레이아웃 구성하기

```
<style>
... ..
aside {
  clear: both;
  width: 93.75%;
  margin: 10px auto;
  padding: 10px;
  background-color: #99cc33;
}
... ..
</style>
```