

# CSS 미디어 쿼리



# 목차

- 반응형 웹 페이지
- 미디어 쿼리

# 반응형 웹

브라우저의 크기에 따라 디자인이 변하는 웹



가로크기: 1008px



가로크기: 673px



가로크기: 401px

# 반응형 웹을 만들기 위해 필요한 속성들

- width의 값을 %로 표현하기
  - %로 설정했을 시, 부모요소의 크기를 기준으로 한다

```
header{
  width:100%;
  height: 30px;
  background-color: red;
}
section{
  width:80%;
  background-color:blue;
}
footer {
  width:50%;
  background-color:green;
}

</style>
</head>
<body>
<header>
  header
</header>
<section>
  section
</section>
<footer>
  footer
</footer>
```



# 반응형 웹을 만들기 위해 필요한 속성들

- **body의 width:100%** 로 하고 나머지 요소들의 width도 %로 설정하면 브라우저 크기에 따라 요소들의 크기를 변하게 할 수 있다

```
body {  
  width:100%;  
}  
header{  
  width:100%;  
  height: 30px;  
  background-color: red;  
}  
section{  
  width:80%;  
  background-color:blue;  
}  
footer {  
  width:50%;  
  background-color:green;  
}  
  
</style>  
</head>  
<body>  
<header>  
  header  
</header>  
<section>  
  section  
</section>  
<footer>  
  footer  
</footer>
```



# 미디어 쿼리

- width의 %설정은 브라우저 크기에 따라 기존 요소들의 폭을 조절할 수 있을 뿐 요소를 안보이게 하는 등 다른 속성들을 변경할 수는 없다
- **미디어쿼리**를 사용하면 브라우저 **크기에 따라** 요소들에게 **다른 속성**들을 적용할 수가 있다

```
@media (max-width:800px){  
  div {  
    background-color: red;  
  }  
}
```

max-width: 800px

800px **이하**인 브라우저 크기에 아래와 같은 속성을 적용

min-width: 400px

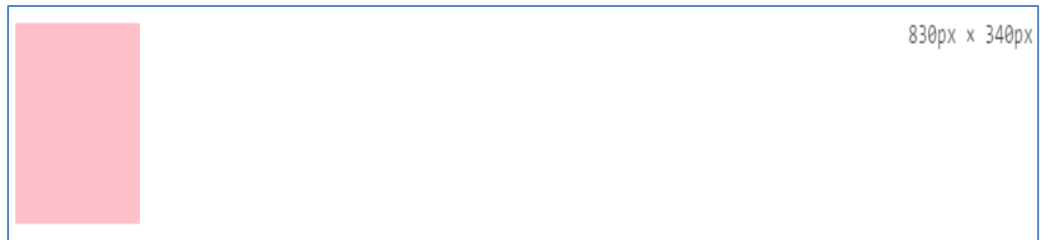
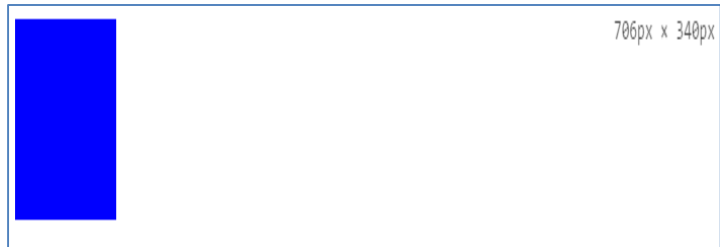
400px **이상**인 브라우저 크기에 아래와 같은 속성을 적용

(min-width: 400px) and (max-width: 800px)

**400px 이상이고 800px 이하**인 브라우저 크기에 아래와 같은 속성을 적용

# 미디어 쿼리

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>css mediaquery</title>
  <style type="text/css">
    div {
      width:100px;
      height: 100px;
      background-color: red;
    }
    @media( min-width: 700px ) {
      div {
        background-color: blue;
      }
    }
    @media( min-width: 800px ) {
      div {
        background-color: pink;
      }
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```



# viewport meta 요소

- viewport meta 요소
  - 웹 페이지를 실행하고 있는 디바이스를 인식할 게 하는 요소

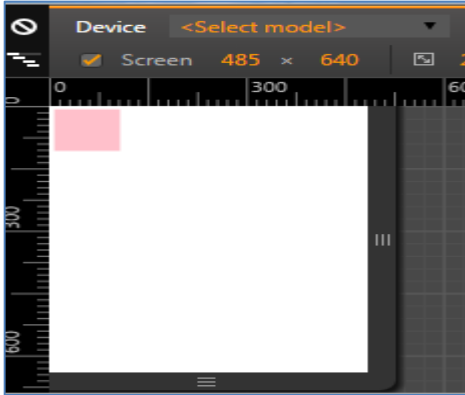
**<head>요소 안에 아래와 같은 viewport meta요소 추가**

**<meta name="viewport" content="width=device-width, initial-scale=1">**

name = "viewport"	디바이스를 인식하는 viewport 메타 태그
width=device-width	브라우저의 폭을 디바이스의 폭으로 설정
initial-scale=1	브라우저에서 표현되는 크기를 배수로 할 것인지 설정여부, 1이면 같은 크기로 사용

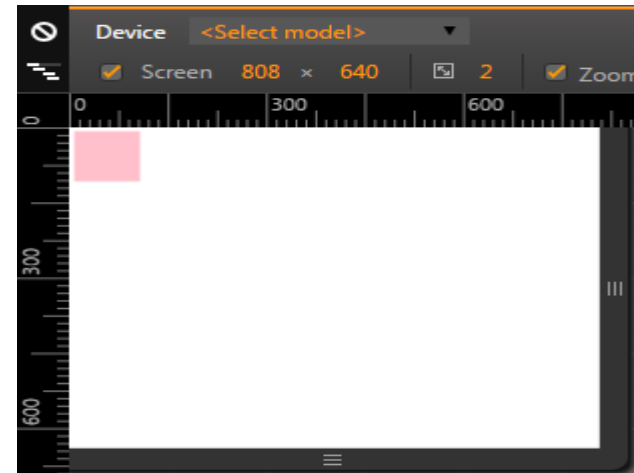
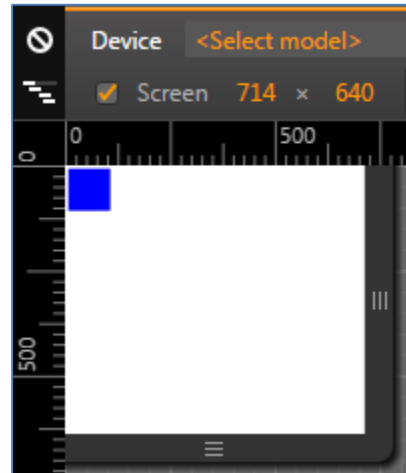
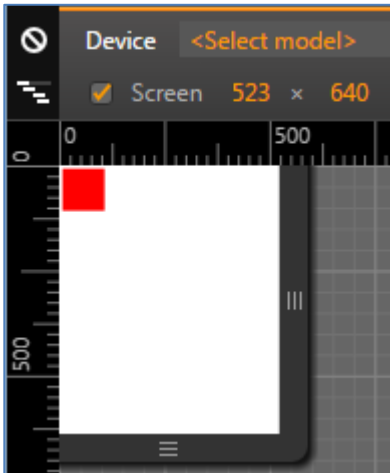


# viewport meta 요소



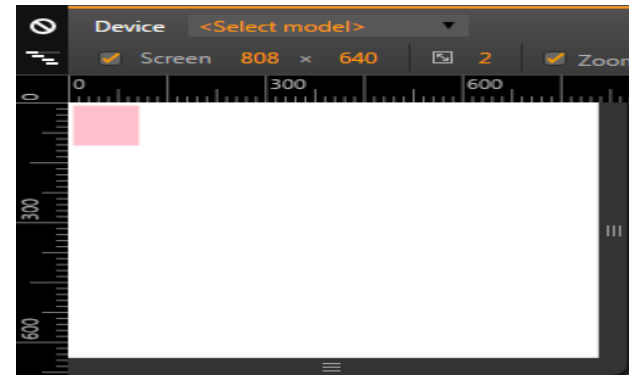
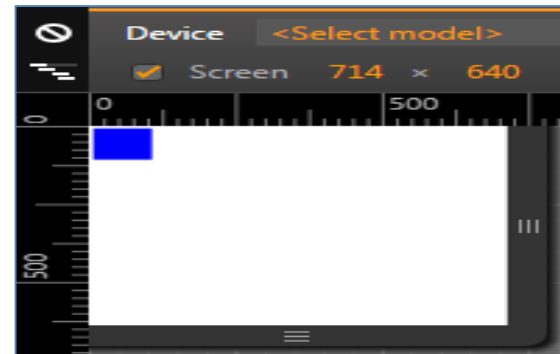
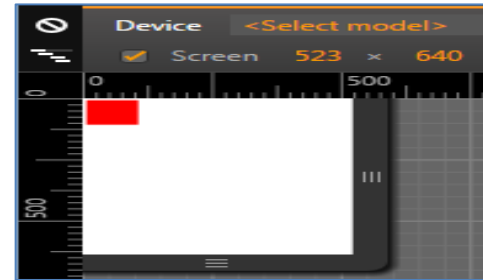
viewport meta요소를 사용하지 않으면 **모바일 또는 태블릿**에서 보고 있는지 여부를 **판단할 수 없다**

**viewport 메타 요소를 적용 후, 디바이스 크기에 따른 색 변화**



# viewport meta 요소

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>css mediaquery</title>
  <style type="text/css">
    div {
      width:100px;
      height: 100px;
      background-color: red;
    }
    @media( min-width: 700px ) {
      div {
        background-color: blue;
      }
    }
    @media( min-width: 800px ) {
      div {
        background-color: pink;
      }
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```



# 주로 설정되는 media-query 크기

디바이스 크기를 알려주는 사이트: <http://viewportsize.com>

디바이스가 너무 많기 때문에 3개 에서 5개 정도의 미디어 쿼리를 설정함

## 부트스트랩 기준

미디어	설정되는 크기
큰 데스크탑	1200px 이상
중간 크기의 데스크탑	992px 이상
태블릿	768px 이상
모바일	768 미만

\*부트스트랩: 트위터에서 제공하는 css, javascript 프레임워크

# 주로 설정되는 media-query 크기

미디어	설정되는 크기
큰 데스크탑	1200px 이상
중간 크기의 데스크탑	992px 이상
태블릿	768px 이상
모바일	768 미만

```
@media (min-width: 768px){  
  
}  
@media (min-width: 992px) {  
  
}  
@media (min-width: 1200px) {  
  
}
```

# 문제 1

- css\_mediaquery\_exam1.html문서를 아래와 같이 만드세요
  - 한글에는 나눔고딕을 적용하세요



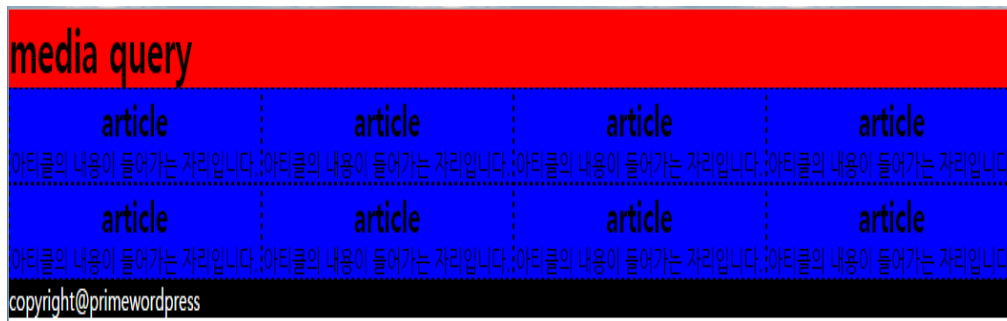
폭이 768px 미만일 때



폭이 768px 이상 일 때

# 문제 1

- css\_mediaquery\_exam1.html문서를 아래와 같이 만드세요



폭이 992px 이상 일때



폭이 1200px 이상일때