제목: 로봇 조립하기- 클럽 리더를 위한 노트

embeds: "\*.png"

materials: [""]

...

**#수업소개:**

이 프로젝트에서 아이들은 자신의 로봇을 만드는 것으로 CSS positioning에 관해 배울 것입니다.

**#온라인 자료**

[trinket](https://trinket.io/)을 사용하여 온라인으로 HTML & CSS 쓰는 걸 추천합니다. 이 프로젝트는 다음 트링켓을 포함하고 있습니다:

+ ['로봇 조립하기' 시작점 -- <http://jumpto.cc/web-robot>

아이들은 또한 이 빈 트링켓을 사용하여 <http://jumpto.cc/trinket-blank> 직접 HTML & CSS를 쓰거나 다음 템플릿 트링켓을 쓸 수 있습니다 <http://jumpto.cc/trinket-template>.

또한 도전과제의 샘플 해법이 포함된 트링켓도 있습니다:

+ ['Build a Robot' Finished](https://trinket.io/html/00736c0e18)

**#오프라인 자료**

이 프로젝트는 원하신다면 [이곳에서 오프라인으로 완성할 수 있습니다](https://www.codeclubprojects.org/en-GB/resources/webdev-working-offline/). 프로젝트 소재는 이 프로젝트의 'Project Materials' 링크를 클릭하여 얻을 수 있습니다. 이 링크엔 'Project Resources' 란이 포함되 있고 아이들이 오프라인으로 프로젝트를 하는데 필요한 소재들이 들어 있습니다. 아이들이 각자 이 소재들을 갖고 있는지 확인하세요. 여기엔 다음 파일이 포함되어 있습니다:

+ Template/Template.html

+ Template/style.css

+ robot/Robot.html

+ Robot/style.css

+ Lots of .png images

또한 이 프로젝트의 도전과제들을 완성한 버전을 'Volunteer Resources' 란에서 찾을 수 있고 다음을 포함하고 있습니다:

+ RobotFinished.html

+ style.css

+ Lots of .png images

(위의 모든 소재들은 프로젝트로 받으시거나 자원자들의 `.zip`파일로 받을 수 있습니다.)

**#배움 목표**

+ HTML id attribute;

+ CSS 쓰기:

+ `#` id selector;

+ 위치;

+ 상단;

+ 왼쪽.

**#도전과제**

+ "나만의 로봇 디자인 하기" - CSS positioning을 응용하여 로봇 만들기;

+ "이미지 직접 더하기" - HTML id attribute와 associated CSS를 사용하여 이미지 더 더하기.

**#오프라인으로 프로젝트 완성시키기**

아이들이 이 프로젝트를 오프라인으로 한다면 사용할 이미지를 `Robot.html`과 같은 폴더에 넣어야 합니다.

사용할 파일의 이름을 `<img>` 태그로 넣으면 됩니다:

```

<img id="hat" src="hat.png" />

```

-------------------------------------------------수업내용-------------------------------

제목: 로봇 조립하기

난이도: HTML & CSS 2

언어: ko-KR

embeds: "\*.png"

materials: ["Club Leader Resources/\*.\*","Project Resources/\*.\*"]

stylesheet: web

...

**# 소개**

이 프로젝트에선 이미지의 위치를 지정해 로봇을 만드는 방법을 배울 것입니다.



**# 1 단계: 로봇 눈 만들기**

로봇에 눈을 달아 줘봅시다!

## 단계별 체크리스트

*+* 온라인으로 하고 있다면 이 트링켓을 열어주세요*:* <http://jumpto.cc/web-robot>. 또는 아래의 embedded 된 트링켓을 사용해도 됩니다.

<div class="trinket">

<iframe src="https://trinket.io/embed/html/b29b50e571" width="100%" height="600" frameborder="0" marginwidth="0" marginheight="0" allowfullscreen>

</iframe>

</div>

+ 이 프로젝트의 각 이미지는 각자의 이름(혹은 id)가 있습니다. 예를 들자면 얼굴과 눈 이미지의 HTML (코드 8번째 줄에서 시작하는‘face’, ‘eyes1’ 와 ‘eyes2') 은 이렇습니다:

```

<img id="face" ...>

<img id="eyes1" ...>

<img id="eyes2" ...>

```

+ 이미지의 id에 `#`을 사용하여 각각의 스타일을 줄 수 있습니다. 이걸로 각 이미지를 따로 스타일 할 수 있게 해줍니다.

`style.css` 파일을 클릭하세요. 로봇의 얼굴과 다른 이미지가 다른 크기 인걸 보셨 나요?

![screenshot](robot-id.png)

+ 이 CSS 코드를 넣어서 로봇의 눈을 스타일 하세요:

```

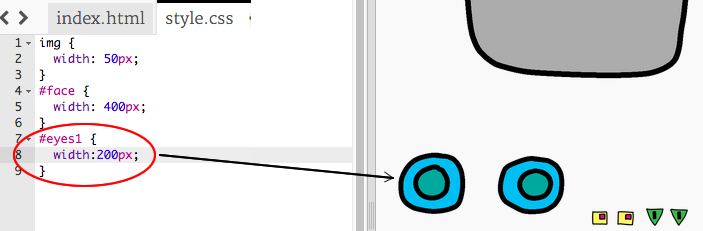
#eyes1 {

width: 200px;

}

```

CSS에선 `#eyes1`만 사용해서 `eyes1` 이미지만 스타일 하고있다는데 주의하세요. 원하신다면 `#eyes2` 나 `#eyes3` 를 쓸 수도 있습니다!



+ 각 이미지가 다음 이미지에 이어서 나오고 있다는 걸 보셨 나요? 이것은 \_\_relative\_\_ positioning이라고 불립니다. 브라우저가 정확히 어디에 로봇의 눈을 둘지 지정하고 싶다면 \_\_absolute\_\_ positioning 을 사용해야 합니다.

`eyes1` 이미지를 위해 다음 3줄의 코드를 CSS에 더해주세요:

```

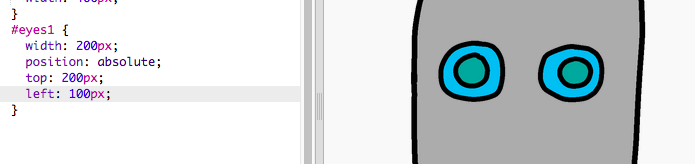
position: absolute;

top: 200px;

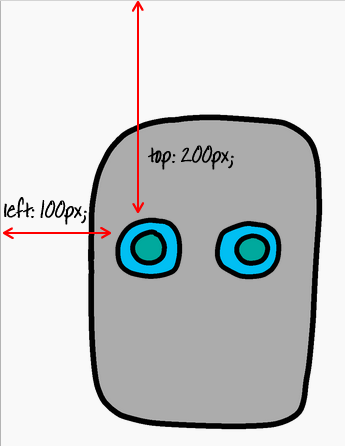
left: 100px;

```

로봇의 눈이 제대로 된 위치로 옮겨 간 것을 볼 수 있습니다.



이 CSS는 브라우저에게 웹페이지의 상단/왼쪽에서 얼마나 멀리 이미지를 표기할지 알려줍니다.



또한 `bottom` 을 `top` 대신 써서 바닥에서부터 얼마나 멀리서 이미지를 보여주게 하거나 `right` 을 `left` 대신 사용할 수도 있습니다.

**# 2 단계: 로봇에게 입을 달자**

로봇에게 입을 달아주죠!

## 단계별 체크리스트

+ 다음 CSS코드를 넣어 `mouth1` 이미지를 스타일 합시다:

```

#mouth1 {

width: 50px;

position: absolute;

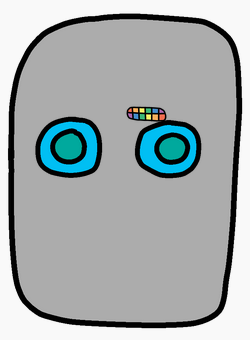
top: 200px;

left: 200px;

}

```

+ 로봇의 입이 꽤 작아 보이고 제대로 위치하고 있지 않군요.



CSS를 고쳐서 이 문제를 해결할 수 있나요?

## 프로젝트를 저장하세요

**##도전과제: 나만의 로봇을 디자인해보세요**

지금까지 배운 기술을 써서 로봇을 만드는 걸 완성시키세요. 완성된 로봇의 예들입니다:



## 프로젝트를 저장하세요

**##도전과제: 이미지 직접 더하기**

로봇에 사용할 추가 이미지들을 찾아서 로봇에 웹페이지에 넣을 수 있나요? 심지어 로봇의 얼굴을 자기 얼굴로 바꿔 넣을 수도 있습니다!

```

<img id="face" src="myFace.png">

```

## 프로젝트를 저장하세요