

Generative Design II

Processing

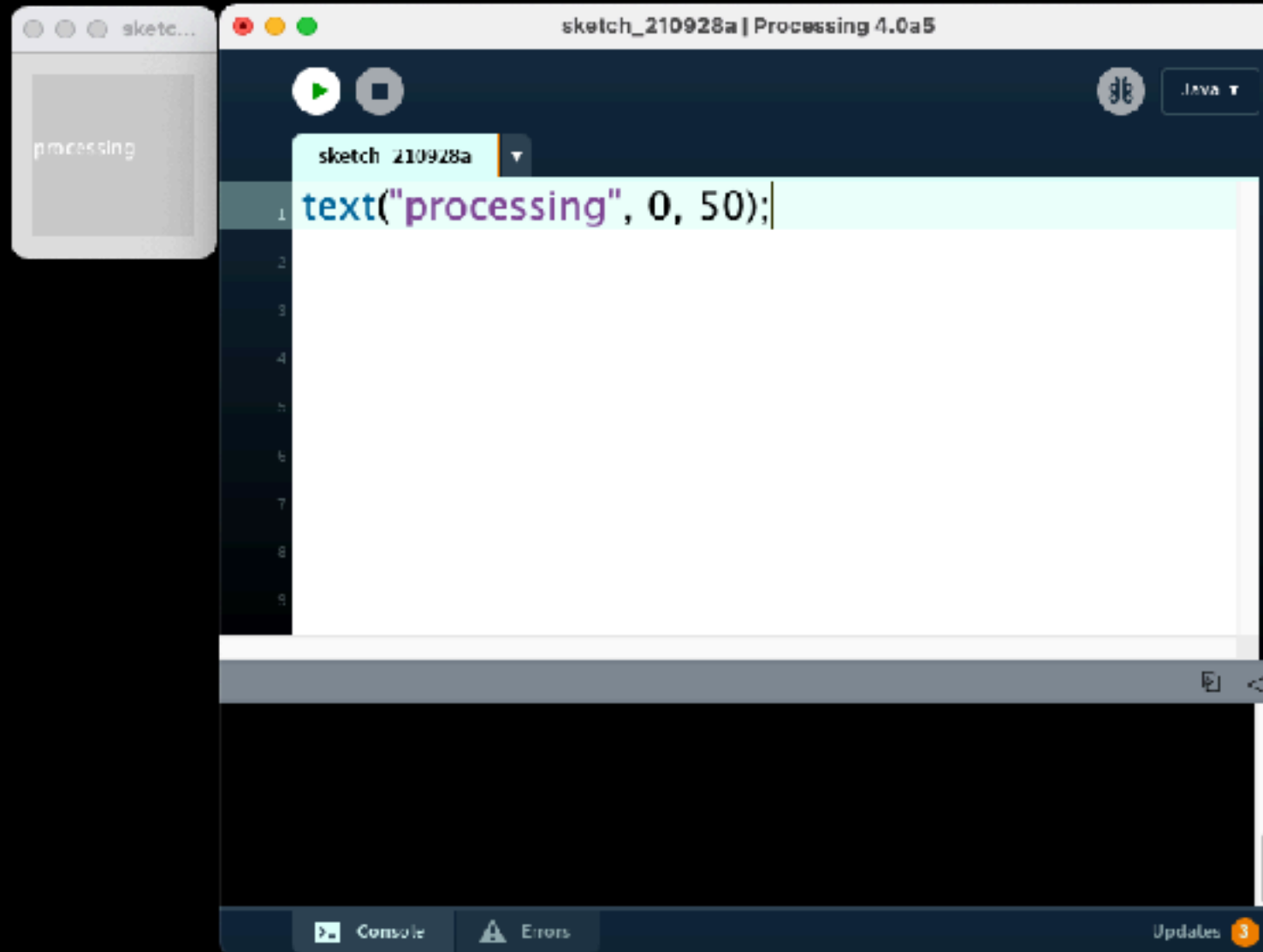
<https://processing.org/>

p5js

<https://p5js.org/>

Processing

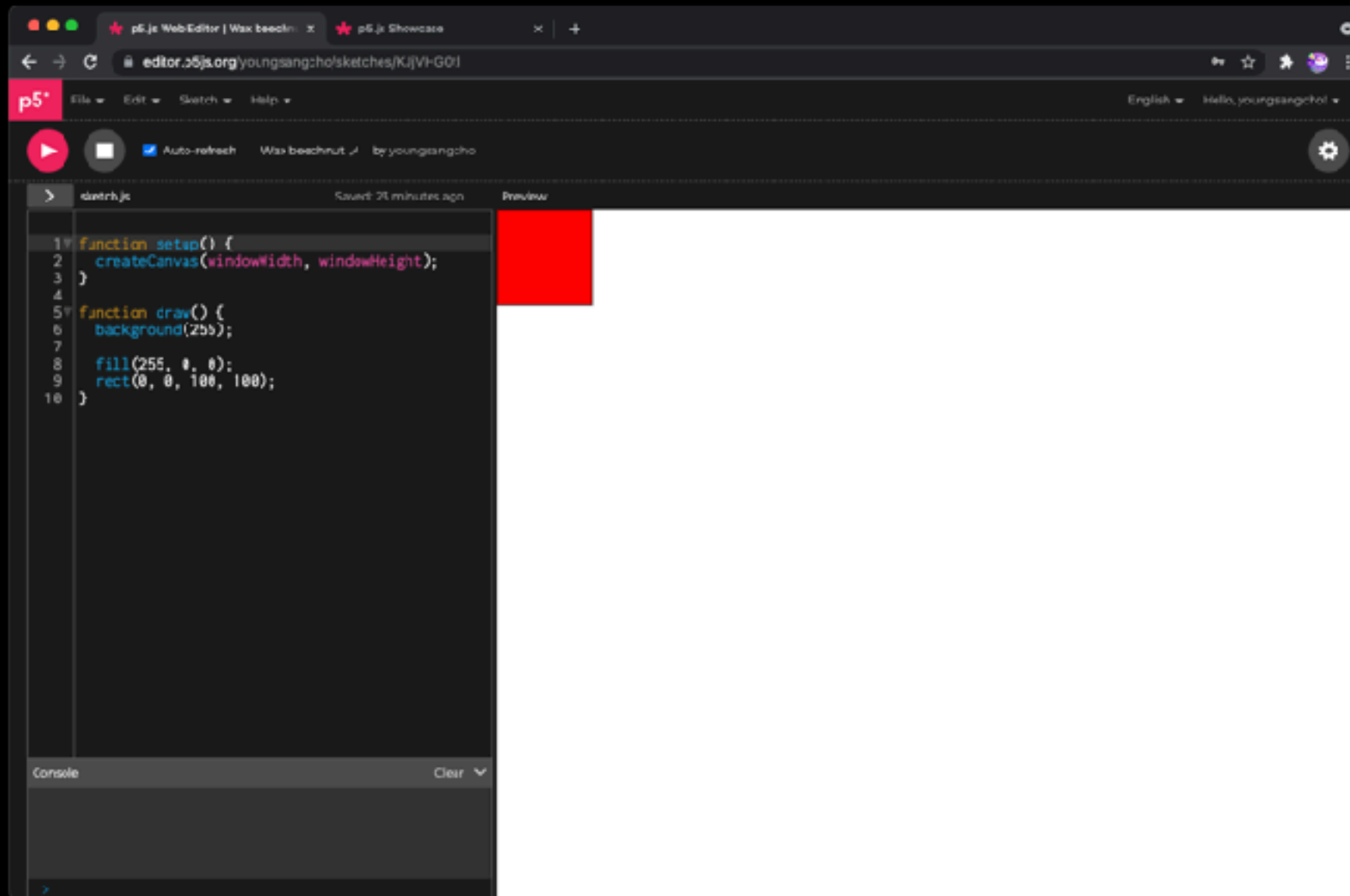
Windows, Mac 등 컴퓨터에서 구동되는 프로그램을 만들 수 있음.
Java



p5js

웹브라우저(크롬, 엣지 등) 안에서 구동되는 프로그램(페이지)을 만들 수 있음.

JavaScript
접근성



p5js Showcase

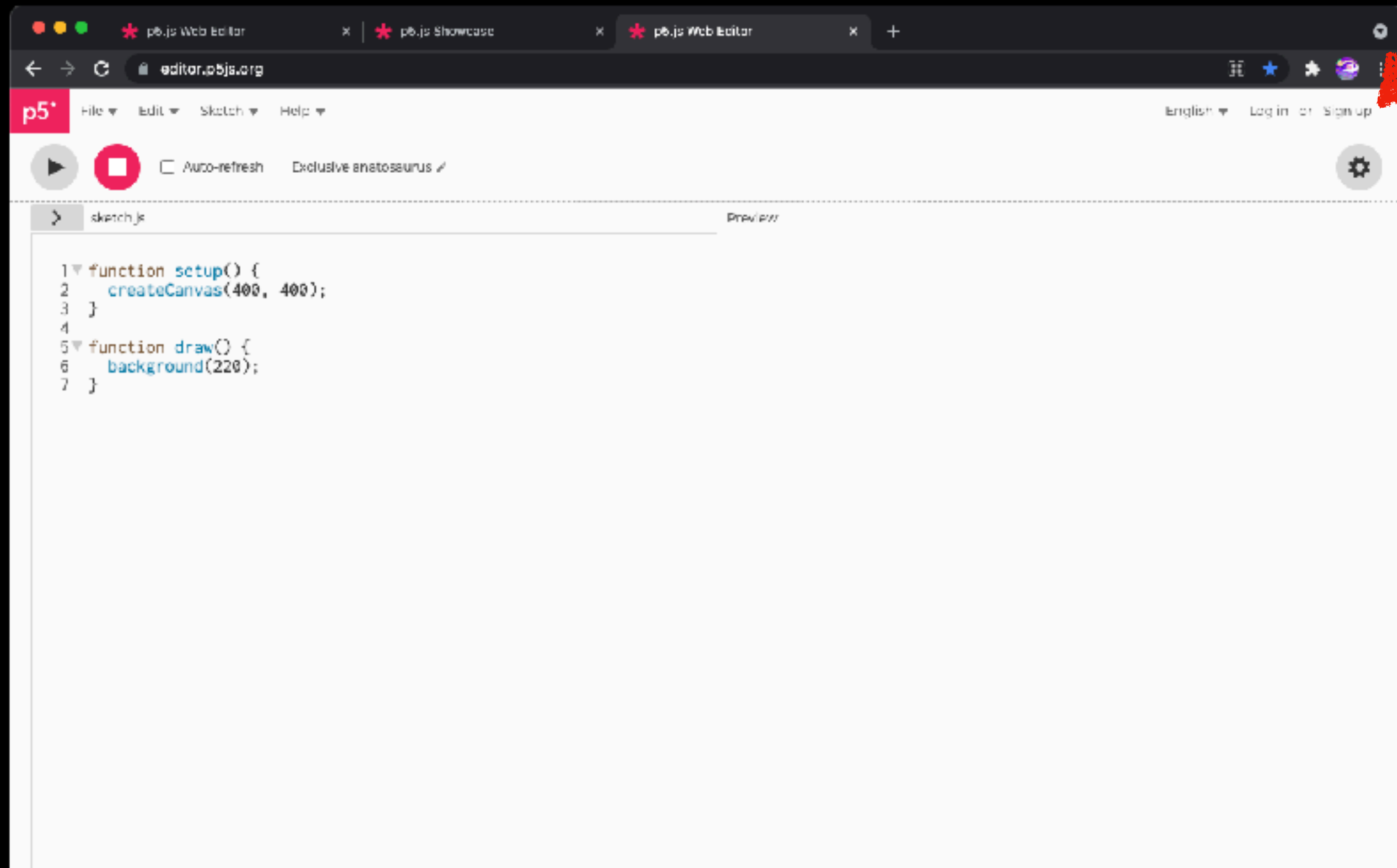
<https://showcase.p5js.org/#/2021-All/>

p5js editor 계정 만들기

<https://p5js.org>

왼쪽 메뉴 editor

오른쪽 위 Sign up



Processing



p5js

<https://github.com/processing/p5.js/wiki/Processing-transition>

Processing

```
void setup() {  
}
```

```
void draw() {  
}
```

p5js

```
function setup() {  
}
```

```
function draw() {  
}
```


Processing

```
void setup() {  
  size(500, 500);  
}
```

p5js

```
function setup() {  
  createCanvas(500, 500);  
}
```

Processing

```
int a = 0;  
float b = 0;  
String str = "hello world";  
PImage img;
```

p5js

```
let a = 0;  
let b = 0;  
let str = "hello world";  
let img;
```

```
var a = 0;  
var b = 0;  
var str = "hello world";  
var img;
```

Processing

```
void draw() {  
    if (mousePressed == true) {  
        // do something  
    }  
  
    if (keyPressed == true) {  
        // do something  
    }  
}  
  
void mousePressed() {  
}
```

p5js

```
function draw() {  
    if (mouseIsPressed == true) {  
        // do something  
    }  
  
    if (keyIsPressed == true) {  
        // do something  
    }  
}  
  
function mousePressed() {  
}
```

Processing

```
void draw() {  
  pushMatrix();  
  // ...  
  popMatrix();  
}
```

p5js

```
function draw() {  
  push();  
  // ...  
  pop();  
}
```

Processing

```
void draw() {  
  println(frameCount);  
}
```

p5js

```
function draw() {  
  console.log(frameCount);  
  print(frameCount);  
}
```

Processing

```
PImage img;

void setup() {
  size(500, 500);
  img = loadImage("dog.jpg");
}

void draw() {
  image(img, 0, 0, 100, 100);
}
```

p5js

```
let img;
let img2;

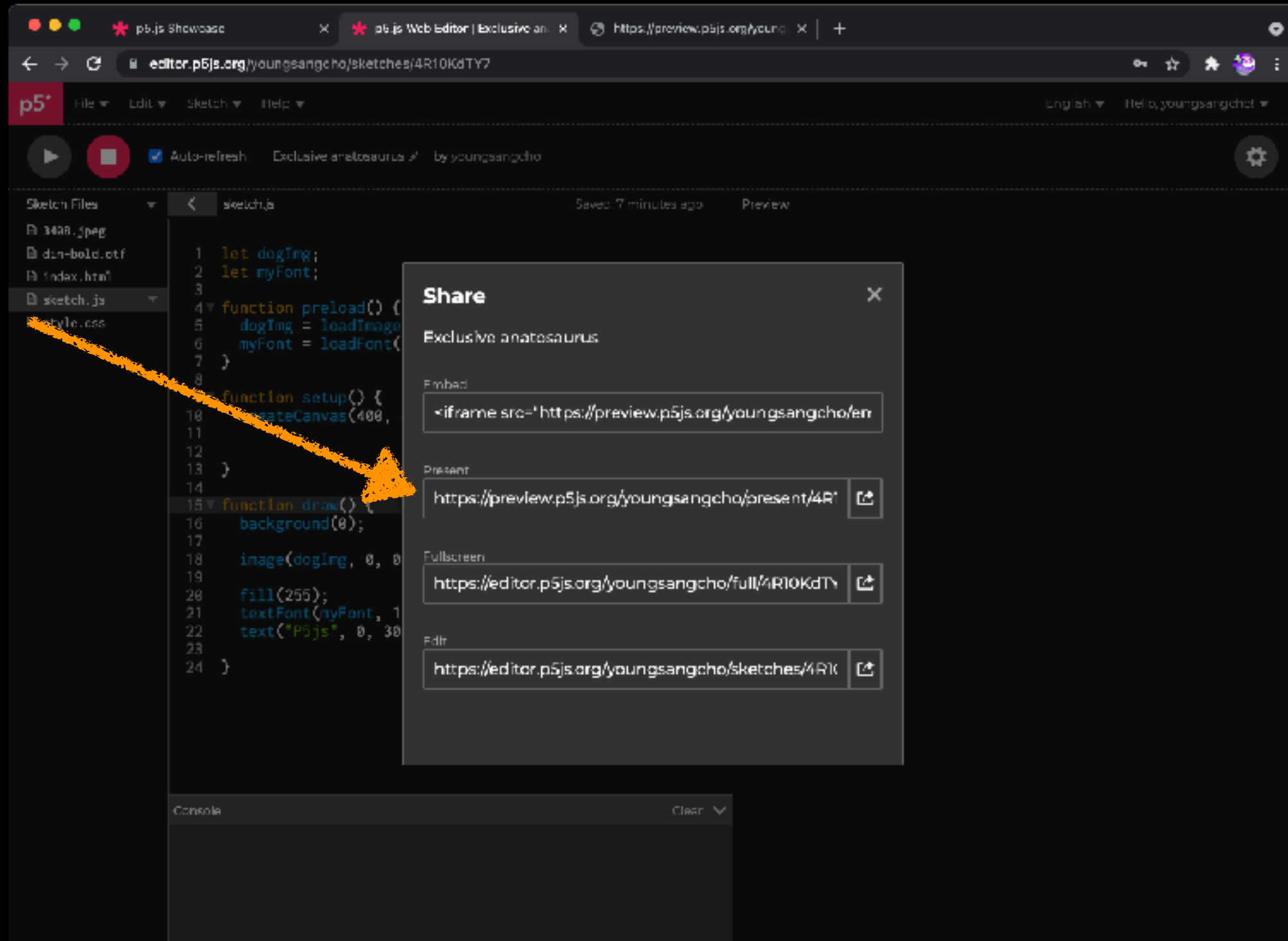
function preload() {
  img = loadImage("dog.jpg");
  img2 = loadImage("dog2.jpg");
}

function setup() {
  createCanvas(500, 500);
}

function draw() {
  image(img, 0, 0, 100, 100);
}
```

공유하기

File - Share



```
function setup(){  
  // code  
}
```

```
function draw(){  
  // code  
}
```



```
function setup(){  
  // code  
}
```



```
function draw(){  
  // code  
}
```

```
function setup(){  
  // code  
}
```



1

```
function draw(){  
  // code  
}
```

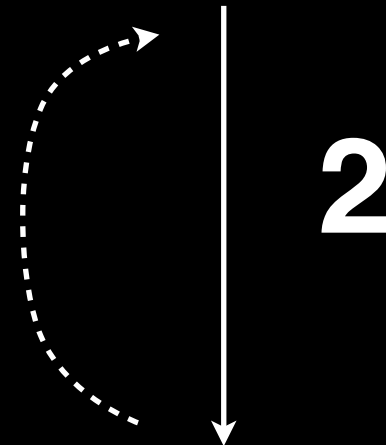


2

```
function setup(){  
  // code  
}
```

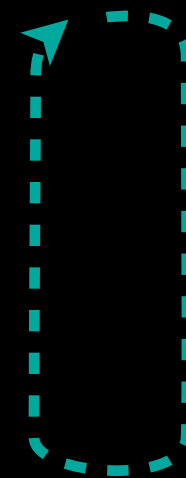


```
function draw(){  
  // code  
}
```



```
function setup(){  
  // code  
}
```

```
function draw(){  
  // code  
}
```




60fps = 1/60(sec)

```
function setup(){  
  createCanvas(500, 500);  
}
```

```
function draw(){  
  background(0);  
  rect(0, 0, 100, 100);  
}
```

도형 그리기



A screenshot of the p5.js reference website. The browser tabs show 'p5.js Showcase', 'p5.js Web Editor | exclusive on', 'https://preview.p5js.org/young', and 'reference | p5.js'. The address bar shows 'p5js.org/reference/'. The page content is organized into a grid of categories under the heading 'Shape'. The '2D Primitives' category is highlighted, and the 'ellipse()' function is underlined in red. The 'Vertex' category is also visible.

Shape

2D Primitives <ul style="list-style-type: none">arc()ellipse()circle()line()point()quad()rect()square()triangle()	Attributes <ul style="list-style-type: none">ellipseMode()noSmooth()rectMode()smooth()strokeCap()strokeJoin()strokeWeight()	Curves <ul style="list-style-type: none">bezier()bezierDetail()bezierPoint()bezierTangent()curve()curveDetail()curveTightness()curvePoint()curveTangent()
Vertex <ul style="list-style-type: none">beginContour()beginShape()bezierVertex()curveVertex()endContour()endShape()quadraticVertex()vertex()normal()	3D Primitives <ul style="list-style-type: none">plane()box()sphere()cylinder()cone()ellipsoid()torus()p5.Geometry	3D Models <ul style="list-style-type: none">loadModel()model()
Constants <ul style="list-style-type: none">HALF_PI	Structure <ul style="list-style-type: none">preload()	DOM <ul style="list-style-type: none">p5.Element

<https://p5js.org/reference/#/p5/ellipse>

도형 그리기

<http://youngsangcho.com/p5/curves/>

변수 變數

A variable is **a factor that can change in quality, quantity, or size**, which you have to take into account in a situation.

어떤 상황의 가변적 요인

A variable is a quantity **that can have any one of a set of values**.

어떤 관계나 범위 안에서 여러 가지 값으로 변할 수 있는 수.

변수 變數

A variable is a storage location paired with an associated symbolic name

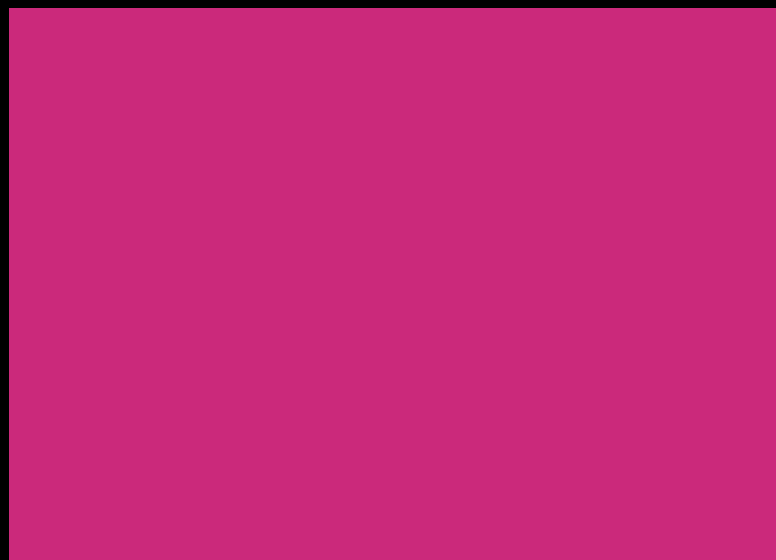
정보, 데이터, 값이 저장되어있는
(메모리 내) 장소의 이름

변수 變數

A variable is a storage location paired with an associated symbolic name

정보, 데이터, 값이 저장되어있는
(메모리 내) 장소의 이름

name



변수 變數

A variable is a storage location paired with an associated symbolic name

정보, 데이터, 값이 저장되어있는
(메모리 내) 장소의 이름

name



0

변수 變數

A variable is a storage location paired with an associated symbolic name

정보, 데이터, 값이 저장되어있는
(메모리 내) 장소의 이름

width

500

frameCount

1

mouseX

140

변수 變數

System Variables

width, height,
mouseX, mouseY,
frameCount

변수 變數

A variable is a storage location paired with an associated symbolic name

정보, 데이터, 값이 저장되어있는
(메모리 내) 장소의 이름

width

500

frameCount

1

mouseX

140

whatever

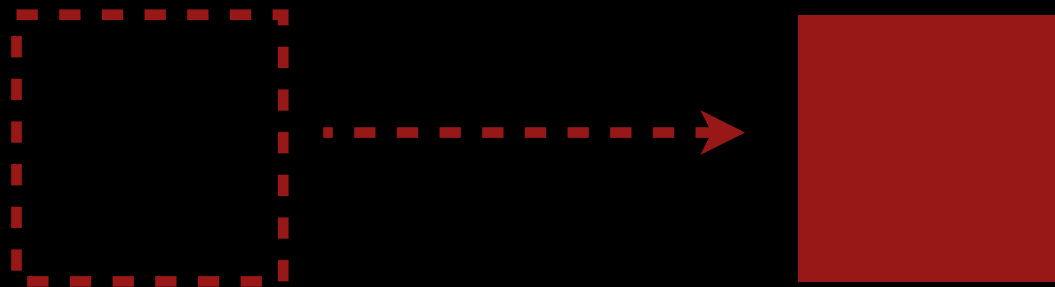
38471

변수 만들기



```
function setup(){  
}  
  
function draw()  
{  
  rect(30,30,50,50);  
}
```

변수 만들기



```
function setup(){
```

```
}
```

```
function draw()
```

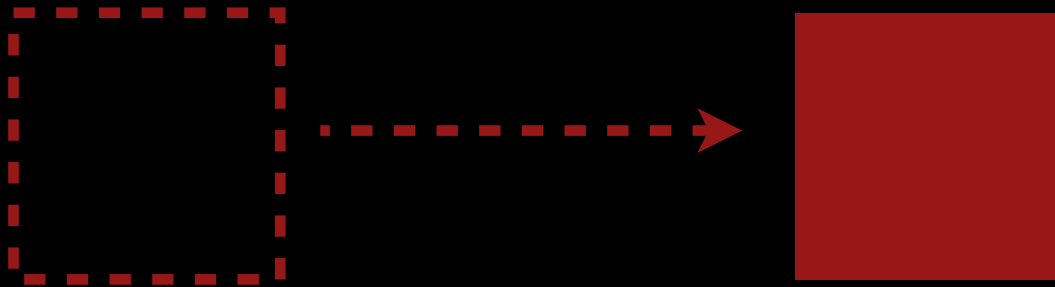
```
{
```

```
  rect(30,30,50,50);
```

```
}
```

31,32,33,34,35...49,50..

변수 만들기



```
function setup(){
```

```
function draw()  
{
```

```
  rect(30,30,50,50);
```

```
}
```

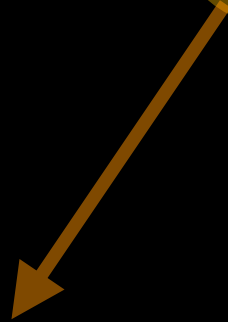
변수 變數 variable

변수 만들기

```
int val = 100;
```

변수 만들기

int val = 100;



Data Type

Name

Value

변수 만들기

```
int val = 100;
```



Data Type

Name

Value

변수 만들기

```
int val = 100;
```

Data Type

Name

Value



Data Types 데이터 형(形)

boolean	true or false
int	Integer Numbers -100 0 301 1023
float	Decimal Numbers -100.0 0.3 304.13
char	'a' 'b'
String	"University"

변수 만들기

```
let val = 100;  
var val = 100;
```

Data Type

Name

Value

Data Types 데이터 형(形)

boolean	true or false
int	Integer Numbers -100 0 301 1023
float	Decimal Numbers -100.0 0.3 304.13
char	'a' 'b'
String	"University"

변수 이름

- 코드 내에서 중복되지 않는 유일한 이름
- 이미 사용되고 있는 이름들은 피해서
width, height, mouseX, mouseY ... (System variables)
- 이름 안에 스페이스는 허용 안 됨.
xPos x pos xP os

변수 이름

- 알파벳, 숫자, _ 알파벳으로 시작해야함. 대소문자 구별.

xPos1 xPos_2 2xPos (x)

xPos , xpos

- 특수문자 허용 안 됨. ? ! * & ...

pos-11? (x) p! (x)

- (보통) 소문자로 시작

- (보통) 두 개 이상의 단어인 경우 두번째 단어 첫 문자는 대문자로.

rx RX

rectPositionX birthYear

변수 이름

- 읽기 쉽고 기억하기 쉬운 '의미 있는' 이름

xp (x) xpp12 (x) ppppp (x)
asdf (x)

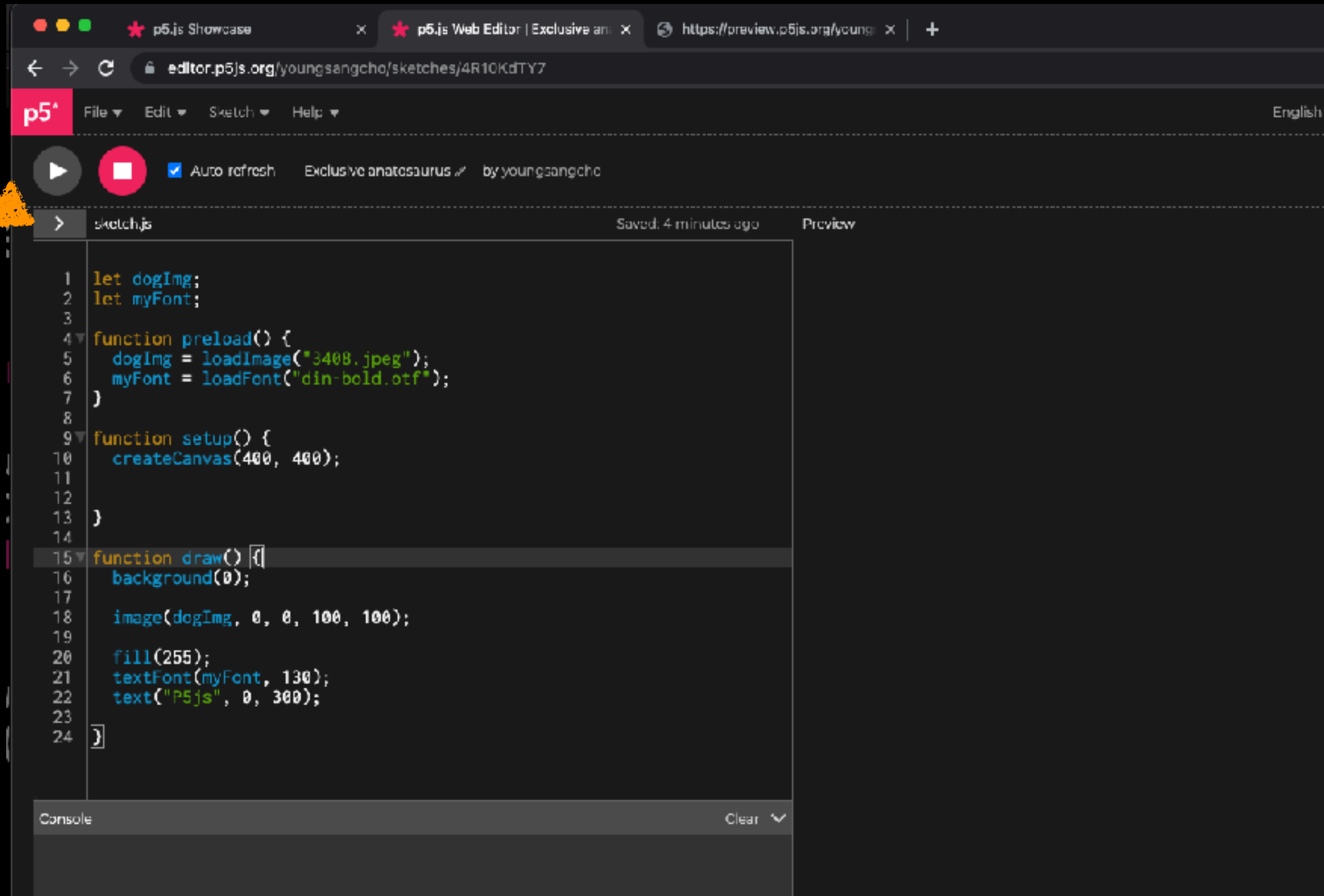
rectPosition (ok) rectPos (o)

- 이름이 목적에 부합하도록.

int x = 50;
rect(x, 0, 100, 100); (???)

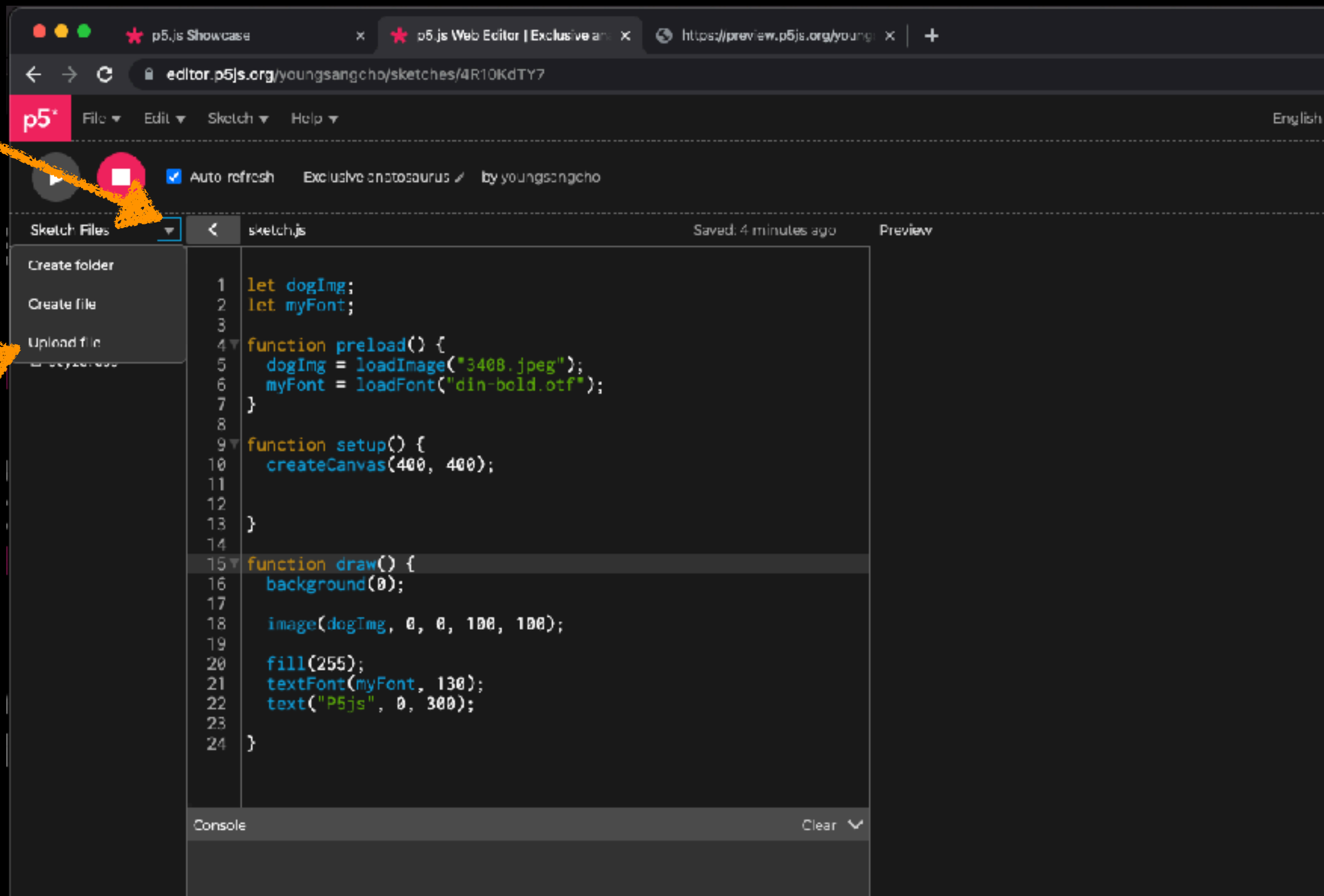
이미지 그리기

1. 파일 업로드



이미지 그리기

1. 파일 업로드



이미지 그리기

2. 코드

```
let myImage;
```

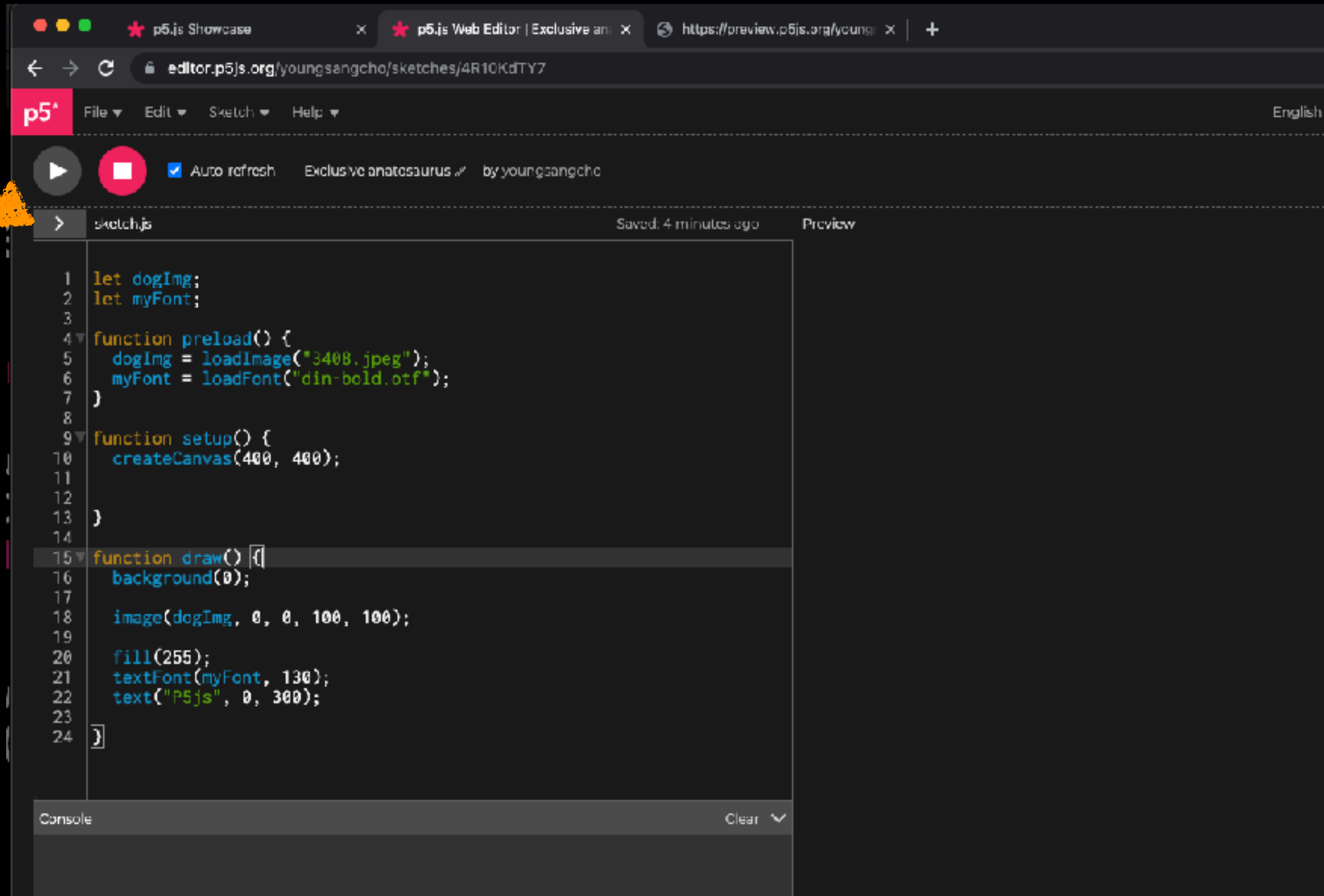
```
function preload() {  
  myImage = loadImage("dog.jpg");  
}
```

```
function setup() {  
  createCanvas(500,500);  
}
```

```
function draw() {  
  background(0);  
  image(myImage, 0, 0, 100, 100);  
}
```

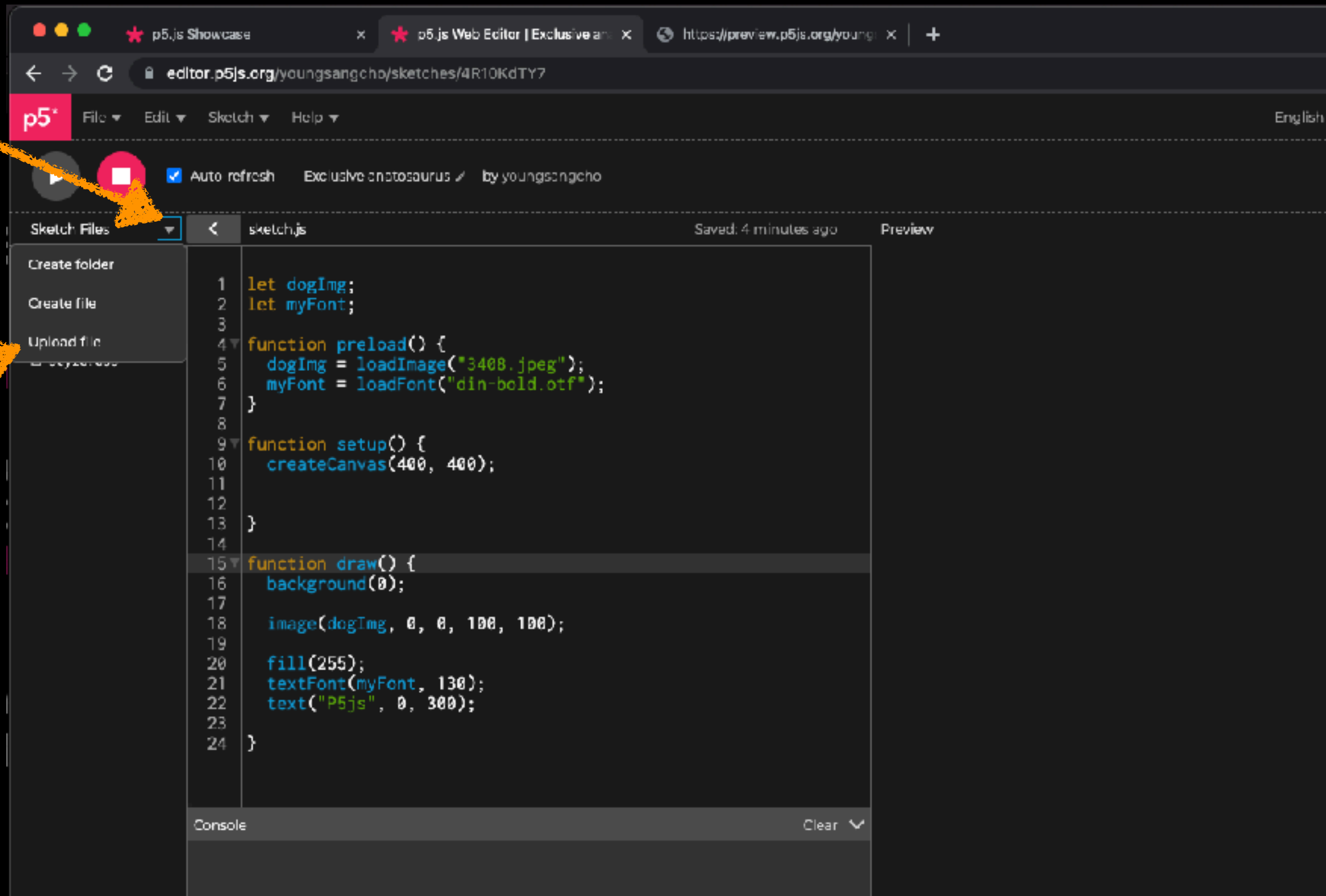
폰트 사용

1. 파일 업로드



폰트 사용

1. 파일 업로드



텍스트 그리기

2. 코드

```
let myFont;
```

```
function preload() {  
    myFont = loadFont("font.otf");  
}
```

```
...
```

```
function draw() {  
    background(0);  
    textFont(myFont, 30);  
    text("Text", 0, 30);  
}
```

과제: (Poster) Design with Code

(과거의 / 상상의) 전시, 이벤트, 영화 등의 정보를 소개하는 포스터 디자인

도형, 이미지, 텍스트

모션 / 인터랙션에 대한 아이디어를 글로
(혹은 구현)

과제: (Poster) Design with Code

- + 1개 이상의 스크린샷 이미지
- + (모션이나 인터랙션 있다면)
스크린 동영상 캡처하여 유튜브 업로드 유튜브 embed
- + 간단한 설명 (왜 / 어떻게)
- + File - Share - Present 링크
- + 코드 복사 붙여넣기

과제

매주 월요일 밤 10시

기한 맞춰 제출

평가 항목

- + 과제별 요구사항
- + 아이디어, 디자인
- + 노력, 시간

과제

남의 코드 베끼지 말기. 가능한 직접 쓰기.

다른 사람/인터넷의 코드를 참조하는 경우,
+ 블로그와 코드 안에 출처 밝히고,
+ 이해해서 내 것으로 만든 경우에만 인정. (모르면 질문)

과제 검사 시, 질문할 수도.