Thực hành] Funny circles

**Mục tiêu**

Luyện tập sử dụng đối tượng trong Javascript.

**Mô tả**

Phát triển một ứng dụng đơn giản với các hình tròn với nhiều hiệu ứng khác nhau.

Ứng dụng này được phát triển theo từng cấp độ (tương ứng với từng bước) đi từ dễ đến khó.

**Để hoàn thành bài thực hành, học viên cần:**

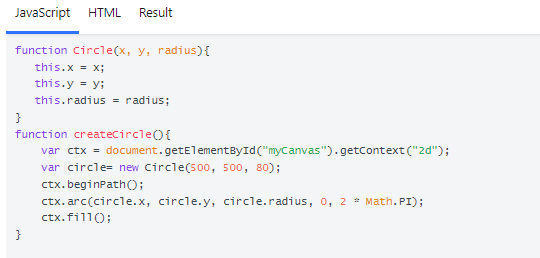
* Đưa mã nguồn lên GitHub

**Hướng dẫn**

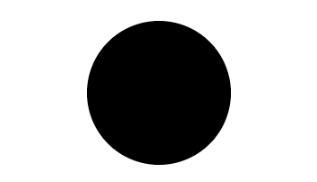
**Bước 1:** Tạo một trang HTML với thẻ canvas.

<!DOCTYPE html>  
<html lang="en">  
 <head>  
 <meta charset="UTF-8">  
 <title>Funny Circle</title>  
 </head>  
 <body>  
 <canvas id="myCanvas" width="1368px" height="768px" style="border:1px solid #d3d3d3;">  
 </body>  
</html>

**Bước 2:** Vẽ một hình tròn đơn giản với kích thước cố định







Trong đoạn mã trên, chúng ta sử dụng phương thức arc() để vẽ một hình tròn. cx và cy là các toạ độ của hình tròn. r là bán kính của hình tròn.

**Bước 3:** Vẽ hình tròn với bán kính ngẫu nhiên

JS

function Circle(x, y, radius){

this.x = x;

this.y = y;

this.radius = radius;

}

function createCircle(){

var ctx = document.getElementById("myCanvas").getContext("2d");

var radius = Math.floor(Math.random() \* 80);

var circle= new Circle(500, 500, radius);

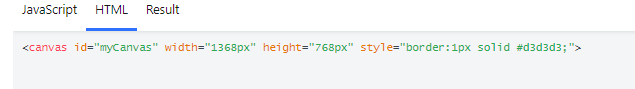
ctx.beginPath();

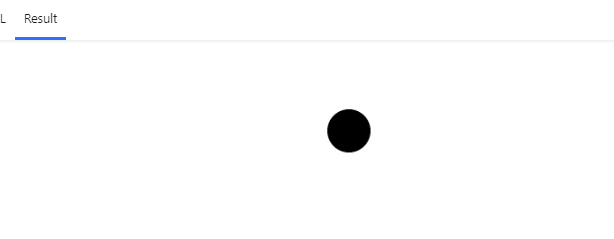
ctx.arc(circle.x, circle.y, circle.radius, 0, 2 \* Math.PI);

ctx.fill();

}

createCircle();





Trong đoạn mã trên, chúng ta sử dụng hàm Math.random() để sinh một bán kính ngẫu nhiên nằm trong khoảng từ 0-80.

**Bước 4:** Vẽ hình tròn với màu ngẫu nhiên

JS

function Circle(x, y, radius){

this.x = x;

this.y = y;

this.radius = radius;

}

function getRandomHex(){

return Math.floor(Math.random()\*255);

}

function getRandomColor(){

var red = getRandomHex();

var green = getRandomHex();

var blue = getRandomHex();

return "rgb(" + red + "," + blue + "," + green +")";

}

function createCircle(){

var ctx = document.getElementById("myCanvas").getContext("2d");

var radius = Math.floor(Math.random() \* 80);

var color = getRandomColor();

var circle= new Circle(500, 500, radius);

ctx.beginPath();

ctx.arc(circle.x, circle.y, circle.radius, 0, 2 \* Math.PI);

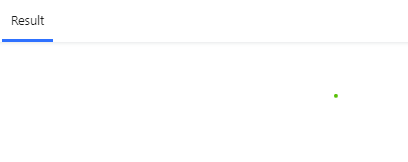
ctx.fillStyle = color;

ctx.fill();

}

createCircle();





Ở đoạn mã trên, chúng ta đã tạo 2 hàm mới để phục vụ việc sinh ra một màu ngẫu nhiên:

* Hàm getRandomHex() sẽ sinh ra một số ngẫu nhiên nằm trong khoảng 0-255.
* Hàm getRandomColor() sinh ra một màu ngẫu nhiên.

**Bước 5:** Vẽ được hình tròn với vị trí ngẫu nhiên

Ở các ví dụ trên, chúng ta vẫn đang cố định vị trí của hình tròn ở toạ độ 500, 500. Ở bước này, chúng ta sẽ sinh toạ độ của hình tròn ngẫu nhiên.

JS

function Circle(x, y, radius){

this.x = x;

this.y = y;

this.radius = radius;

}

function getRandomHex(){

return Math.floor(Math.random()\*255);

}

function getRandomColor(){

var red = getRandomHex();

var green = getRandomHex();

var blue = getRandomHex();

return "rgb(" + red + "," + blue + "," + green +")";

}

function createCircle(){

var ctx = document.getElementById("myCanvas").getContext("2d");

var radius = Math.floor(Math.random() \* 80);

var color = getRandomColor();

var x = Math.random() \* window.innerWidth;

var y = Math.random() \* window.innerHeight;

var circle= new Circle(x, y, radius);

ctx.beginPath();

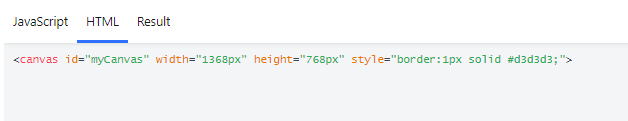
ctx.arc(circle.x, circle.y, circle.radius, 0, 2 \* Math.PI);

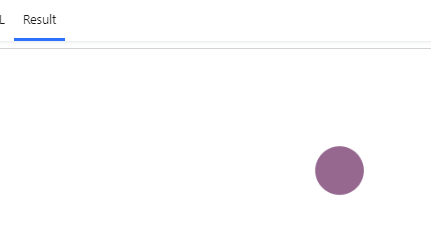
ctx.fillStyle = color;

ctx.fill();

}

createCircle();





Ở đoạn mã trên, chúng ta có sử dụng window.innerWidth và window.innerHeight là 2 thuộc tính trả về kích thước của cửa sổ trình duyệt.

**Bước 6:** Vẽ được 2 hình tròn.

Ở các bước trước, chúng ta chỉ mới tạo được 1 hình tròn. Ở bước này, chúng ta thử tạo 2 hình tròn.

function Circle(x, y, radius){

this.x = x;

this.y = y;

this.radius = radius;

}

function getRandomHex(){

return Math.floor(Math.random()\*255);

}

function getRandomColor(){

var red = getRandomHex();

var green = getRandomHex();

var blue = getRandomHex();

return "rgb(" + red + "," + blue + "," + green +")";

}

function createCircle(){

var ctx = document.getElementById("myCanvas").getContext("2d");

var radius = Math.floor(Math.random() \* 80);

var color = getRandomColor();

var x = Math.random() \* window.innerWidth;

var y = Math.random() \* window.innerHeight;

var circle= new Circle(x, y, radius);

ctx.beginPath();

ctx.arc(circle.x, circle.y, circle.radius, 0, 2 \* Math.PI);

ctx.fillStyle = color;

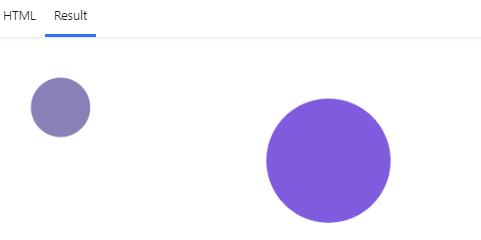
ctx.fill();

}

createCircle();

createCircle();





Cách đơn giản để tạo 2 hình tròn đó là gọi phương thức createCircle() 2 lần liên tiếp nhau.

**Bước 7:** Tạo rất nhiều hình tròn.

Để tạo rất nhiều hình tròn, chúng ta không thể sử dụng phương thức như ở bước 6. Thay vào đó, chúng ta có thể sử dụng vòng lặp để đạt được mục tiêu này.

function Circle(x, y, radius){

this.x = x;

this.y = y;

this.radius = radius;

}

function getRandomHex(){

return Math.floor(Math.random()\*255);

}

function getRandomColor(){

var red = getRandomHex();

var green = getRandomHex();

var blue = getRandomHex();

return "rgb(" + red + "," + blue + "," + green +")";

}

function createCircle(){

var ctx = document.getElementById("myCanvas").getContext("2d");

var radius = Math.floor(Math.random() \* 80);

var color = getRandomColor();

var x = Math.random() \* window.innerWidth;

var y = Math.random() \* window.innerHeight;

var circle= new Circle(x, y, radius);

ctx.beginPath();

ctx.arc(circle.x, circle.y, circle.radius, 0, 2 \* Math.PI);

ctx.fillStyle = color;

ctx.fill();

}

function createMultipleCircle(){

for(var i = 0; i < 30; i++){

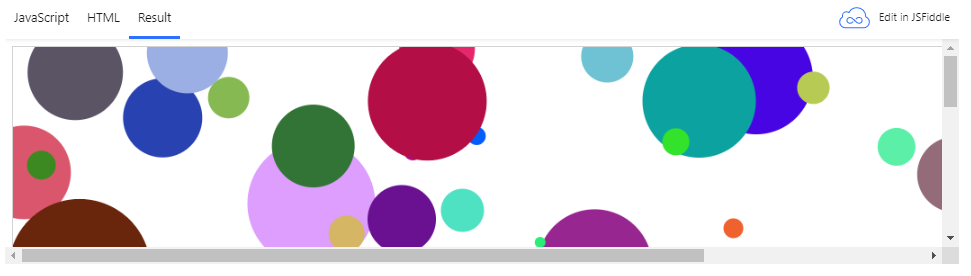
createCircle();

}

}

createMultipleCircle();





Trong đoạn mã trên, chúng ta đã sử dụng một vòng lặp for để tạo ra 30 hình tròn.

Như vậy, chúng ta đã tạo được một số hiệu ứng khá thú vị với các hình tròn.

Thông qua các bước ở trên, chúng ta đã luyện tập:

* Cách làm việc với đối tượng
* Truyền tham số vào cho đối tượng
* Sử dụng các thuộc tính của đối tượng
* Tạo các hàm để bổ sung tính năng cho ứng dụng

**Yêu cầu bổ sung:** Thêm hiệu ứng di chuyển cho các hình tròn.