JavaScript HTML5 Canvas Examples

Drawing a rectangle **2**

Drawing a circle **2**

Drawing a line **3**

Drawing text **4**

Drawing an image **4**

Setting the fill style **5**

Setting the stroke style **5**

Drawing a path **6**

Setting the line width **7**

Setting the line cap style **8**

Drawing a rectangle

```
<canvas id="myCanvas"></canvas>
```

```
const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.fillRect(10, 10, 50, 50);
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The fillRect method is used to draw a rectangle at coordinates (10, 10) with a width of 50 and a height of 50.

Drawing a circle

```
<canvas id="myCanvas"></canvas>
```

```
const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.beginPath();
ctx.arc(50, 50, 30, 0, 2 * Math.PI);
ctx.fill();
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The beginPath method is used to start a new path. The arc method is used to draw a circle centered at (50, 50) with a radius

of 30. Finally, the fill method is used to fill the circle with the current fill style.

Drawing a line

```
<canvas id="myCanvas"></canvas>
```

```
const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.beginPath();
ctx.moveTo(10, 10);
ctx.lineTo(90, 90);
ctx.stroke();
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The beginPath method is used to start a new path. The moveTo method is used to move the pen to (10, 10). The lineTo method is used to draw a line to (90, 90). Finally, the stroke method is used to stroke the line with the current stroke style.

Drawing text

```
<canvas id="myCanvas"></canvas>
```

```
const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.font = '30px Arial';
ctx.fillText('Hello, World!', 10, 50);
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with `getContext('2d')`. The font property is used to set the font and font size. The fillText method is used to draw the text 'Hello, World!' at coordinates (10, 50).

Drawing an image

```
<canvas id="myCanvas"></canvas>

const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
const image = new Image();
image.onload = function() {
  ctx.drawImage(image, 10, 10);
}
image.src = 'path/to/image.png';
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). An Image object is created and assigned a source URL. When the

image is loaded, the onload function is called, which uses the drawImage method to draw the image at coordinates (10, 10).

Setting the fill style

```
<canvas id="myCanvas"></canvas>

const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.fillStyle = 'red';
ctx.fillRect(10, 10, 50, 50);
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The fillStyle property is used to set the fill color to red. The fillRect method is used to draw a rectangle at coordinates (10, 10) with a width of 50 and a height of 50.

Setting the stroke style

```
<canvas id="myCanvas"></canvas>

const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.strokeStyle = 'blue';
```

```
ctx.strokeRect(10, 10, 50, 50);
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The strokeStyle property is used to set the stroke color to blue. The strokeRect method is used to draw an outlined rectangle at coordinates (10, 10) with a width of 50 and a height of 50.

Drawing a path

```
<canvas id="myCanvas"></canvas>


const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.beginPath();
ctx.moveTo(10, 10);
ctx.lineTo(90, 90);
ctx.lineTo(10, 90);
ctx.closePath();
ctx.stroke();
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The beginPath method is used to start a new path. The moveTo method is used to move the pen to (10, 10). The lineTo method is used to draw a line to (90, 90) and then to (10, 90). The

closePath method is used to connect the last point back to the starting point. Finally, the stroke method is used to stroke the path with the current stroke style.

Setting the line width

```
<canvas id="myCanvas"></canvas>

const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
ctx.lineWidth = 5;
ctx.strokeRect(10, 10, 50, 50);
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The lineWidth property is used to set the width of lines to 5. The strokeRect method is used to draw an outlined rectangle at coordinates (10, 10) with a width of 50 and a height of 50.

Setting the line cap style

```
<canvas id="myCanvas"></canvas>

const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');
```

```
ctx.lineWidth = 10;
ctx.lineCap = 'round';
ctx.beginPath();
ctx.moveTo(10, 10);
ctx.lineTo(90, 90);
ctx.stroke();
```

This code creates a canvas element with an id of 'myCanvas'. It then gets a reference to the canvas context with getContext('2d'). The lineWidth property is used to set the width of lines to 10.