



UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERÍA Y ARQUITECTURA  
PROGRAM MICROSOFT IT ACADEMY  
MTA-98-375

---

# HTML 5 Application Development Fundamentals



Microsoft  
IT Academy

Lección 7: Manejando interfaz gráfica con  
CSS.

---

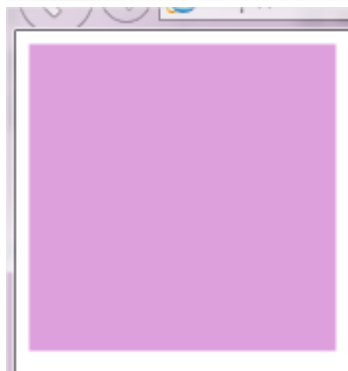
## Utilizando SVG y Canvas

### Ejercicio 1.

En este ejercicio se utilizarán los filtros para los elementos SVG, los cuales presentan efectos sobre elementos SVG.

1. Crear un proyecto en WebMatrix y agregar un nuevo documento HTML, asignarle el nombre de index.html.
2. Se creará el documento HTML, sin estilo como se muestra a continuación:

```
<svg>
  <rect width="150" height="150" stroke="plum" stroke-width="3" fill="plum" />
</svg>
```



3. Ahora se procederá a ocupar unas determinadas etiquetas de SVG, para crear un filtro determinado, en este caso se creará el filtro de desvanecimiento gaussiano. Al filtro se le coloca el nombre de a1.

```
<defs>
  <filter id="a1" x="0" y="0">
    <feGaussianBlur in="SourceGraphic" stdDeviation="20" />
  </filter>
</defs>
```

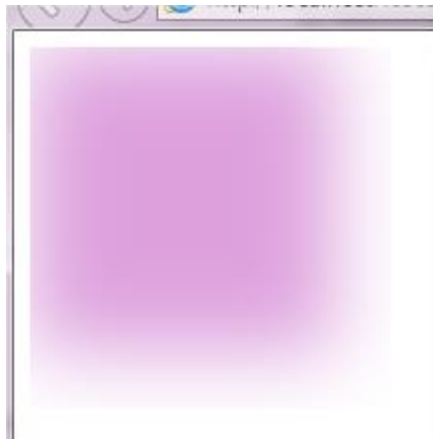
4. Ya creado el filtro se procede a agregar el atributo de filtro al elemento rect, en el cual se especifica que filtro se utilizará.

```
filter="url(#a1)" />
```

El código de svg completo queda como el siguiente.

```
<svg>
  <defs>
    <filter id="a1" x="0" y="0">
      <feGaussianBlur in="SourceGraphic" stdDeviation="20" />
    </filter>
  </defs>
  <rect width="150" height="150" stroke="plum"
    stroke-width="3" fill="plum" filter="url(#a1)" />
</svg>
```

El resultado de la aplicación del filtro se muestra en la imagen.



5. Investigar la aplicación de alguno de los filtros vistos en clase y utilizarlos en este mismo elemento SVG.

## Ejercicio 2.

Se creará un proyecto para el manejo de transformaciones en elementos utilizando JavaScript, en los elementos que se encuentran en el elemento Canvas.

1. Crear un proyecto en WebMatrix y agregar un nuevo documento HTML, asignarle el nombre de index.html.
2. Se creará el documento HTML, sin estilo como se muestra a continuación:

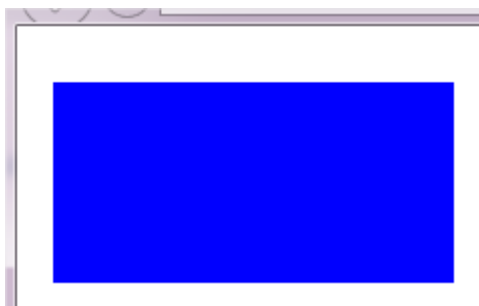
```
<body onload = "fun1();" >  
<canvas id="rectangulo" height='300' width='500'>  
</canvas>  
</body>
```

Se observa que se agrega el atributo "onload" al elemento body, esto implica que la función que se encuentra como valor de ese atributo, se ejecutará en el momento en que la página se cargue.

3. Si el programa se presenta ahorita, generaría un error, ya que la función no está establecida, se crea un elemento de tipo <script> y dentro de él se crea la función en JavaScript, para que este cree un cuadrado en el área de Canvas.

```
<script>
function fun1() {
var canvas =
document.getElementById("rectangulo");
context = canvas.getContext("2d");
context.fillStyle = "blue";
context.fillRect(10, 20, 200, 100);
}
</script>
```

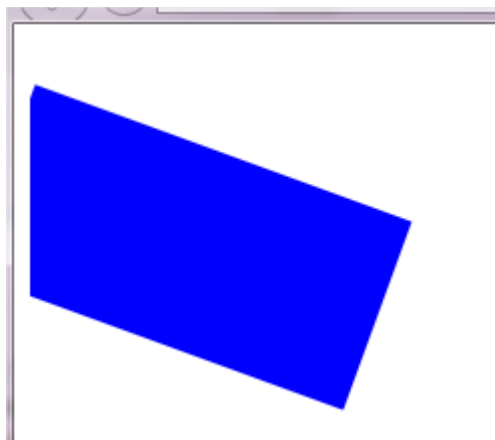
Ya creada la función, al presentar la página web, esta se observara así:



4. Para aplicar una transformación, solo se llama a una función determinada de JavaScript y se le asignan los parámetros deseados. La siguiente línea se agrega previo al relleno del rectángulo.

```
context.rotate(20*Math.PI/180);
```

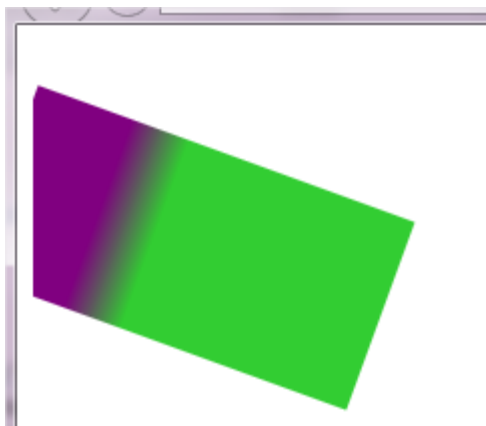
Al agregar la línea, el rectángulo se gira antes de ser rellenado, se presenta como en la imagen siguiente.



5. Ahora se creara un gradiente, para asignarle un color con degradado como relleno, así en vez de un color plano, el rectángulo se rellenara con un efecto más agradable, esto se realiza antes del relleno del rectángulo.

```
var grd=context.createLinearGradient(0,0,150,0);  
grd.addColorStop(0.6,"limegreen");  
grd.addColorStop(0.4,"purple");  
context.fillStyle=grd;
```

Esta última línea de código, es sustituida por el estilo de llenado que ya existía, así se aplica el gradiente como nueva forma de llenado.



6. Investigue y aplique otras transformaciones en el elemento Canvas, utilizando JavaScript.