

# UD-07b: Generación de PDFs desde PHP

Desarrollo Web en Entorno Servidor

Curso 2020/2021

# Objetos gráficos

## Líneas

Para dibujar segmentos rectilíneos (líneas) existe el método:

```
$documento->Line(x0, y0, x1, y1, $estilo_segmento)
```

Donde `x0`, `y0`, `x1` e `y1` son la coordenadas de los extremos del segmento y `$estilo_segmento` es un **array** que incluye los valores requeridos por el método `SetLineStyle` visto anteriormente. Si no se especifica este valor la línea utilizará el estilo por defecto o el último que haya sido definido.

# Objetos gráficos


## Rectángulos

Para dibujar rectángulos tenemos el método:

```
$documento->Rect(x, y, ancho, alto, tipo_de_trazado, estilo, color_fondo);
```

- Las coordenadas **x** e **y** marcan el vértice superior izquierdo.
- Las dimensiones se denotan en **ancho** y **alto**.
- El parámetro **tipo\_de\_trazado** puede tener estos valores:
  - ✓ **D** → Traza la línea sin cerrar la poligonal.
  - ✓ **d** → Cierra la poligonal (agregando un segmento que une el primer punto y el último) y dibuja la línea.
  - ✓ **F** → Rellena el trazado sin dibujar la línea. Utiliza la regla del número distinto para determinar la región a rellenar.
  - ✓ **F\*** → Rellena el trazado sin dibujar la línea. Utiliza la regla del par-impar para determinar la región a rellenar.
  - ✓ **DF** → Traza la línea (sin cerrarla) y rellena el trazado usando la regla del núm. distinto para determinar la región a rellenar.
  - ✓ **DF\*** → Traza la línea (sin cerrarla) y rellena el trazado, usando la regla del par-impar para determinar la región a rellenar.
  - ✓ **Df** → Cierra el trazado, dibuja la línea y rellena usando la regla del número distinto.
  - ✓ **df\*** → Cierra el trazado, dibuja la línea y rellena usando la regla par-impar.
  - ✓ **CNZ** → Permite usar elementos gráficos cerrados (rectángulos, círculos, etc.) para recortar imágenes.



- 
- El parámetro `estilo` define el estilo de cada uno de los lados **mediante array asociativo** que puede tener la forma:

- ✓ Si se trata de aplicar un estilo único a los cuatro lados:

```
array('all' => estilo_segmento);
```

- ✓ Si pretendemos que cada lado tenga un estilo diferente debemos:

```
array('L' => estilo_izquierda, 'R' => estilo_derecha,  
      'T' => estilo_superior, 'B' => estilo_inferior);
```

- ✓ Si se incluye un cero como valor de alguna de las posiciones, estas no serán dibujadas.

- Para el caso de trazados cerrados con `color_de_fondo` podremos especificar un array que contenga los valores RGB del color de fondo aplicable.

# Objetos gráficos

## Círculos

Este método dibuja un círculo con la línea y el relleno indicados en los parámetros, con su centro situado en el punto **x** e **y**, y con el **radio** especificado.

Los valores por defecto de **inicio** y **final** son 0 y 360. Representaría ángulos inicio y final con el eje horizontal, del sector circular delimitado por estos radios.

```
$documento->Circle(x, y,  
                  radio,  
                  inicio, final,  
                  tipo_de_trazado,  
                  estilo_segmento,  
                  color_fondo);
```

# Objetos gráficos

## Elipses

Este método dibuja una elipse (o una porción de ella) de forma similar al círculo. La única diferencia en cuanto a parámetros es que se incluyen `semieje_horizontal` y `semieje_vertical` (medida de los semiejes de la elipse) e `inclinación` que indica el ángulo que forma con la horizontal el eje de la elipse.

```
$documento->Ellipse( x, y,  
                    semieje_horizontal, semieje_vertical,  
                    inclinacion,  
                    inicio, final,  
                    tipo_de_trazado,  
                    estilo_segmento,  
                    color_fondo)
```

# Objetos gráficos

## Rectángulo redondeado

Este método dibuja un rectángulo del `ancho` y `alto` especificados con sus esquinas redondeadas y con un `radio_esquina`, cuyo vértice superior izquierdo viene especificados por los valores de `x` e `y`.

El parámetro `forma_esquinas` es una cadena del tipo "1011" que indica cuales de las esquinas (1) han de ser redondeas y cuales no (0). El primer valor corresponde a la esquina superior derecha continuando en sentido horario. Los parámetros `tipo_de_trazado`, `estilo_segmento`, `color_fondo` son igual que en casos anteriores.

```
$documento->RoundedRect( x, y,  
                           ancho, alto,  
                           radio_esquina,  
                           forma_esquinas,  
                           tipo_de_trazado,  
                           estilo_segmento,  
                           color_fondo);
```

# Imágenes

Esta clase también permite integrar imágenes en nuestro pdf, con este método:

```
$documento->Image(imagen,  
                  x, y,  
                  ancho, alto,  
                  tipo,  
                  enlace,  
                  punto_insercion,  
                  remuestreo,  
                  resolucion,  
                  alineacion,  
                  imk,  
                  imgk,  
                  borde,  
                  proporcionalidad,  
                  ocultar,  
                  encajar);
```



# Imágenes

- **imagen** es la ruta y nombre de la imagen
- **x** e **y** las coordenadas donde se posicionará la esquina superior izquierda de la misma
- **ancho** y **alto** sus dimensiones después de ser insertada en el documento. Si se omitiera una de las dos dimensiones (o se asignara el valor cero) se redimensionaría la otra de forma proporcional a sus dimensiones originales.
- Si **encajar** es *false* la imagen se incluye según sus dimensiones originales. Si es **true** la imagen se redimensiona manteniendo la proporcionalidad y limitando su tamaño de forma que no sobrepase los márgenes del documento.
- **tipo** puede ser GIF, JPG, PNG... Si no se especifica lo toma de la extensión.
- **enlace** permite incluir una cadena conteniendo el destino de un eventual hiperenlace.
- **punto\_insercion** permite especificar la posición en la que quedará el punto de inserción después de haber incluido la imagen. Admite uno de estos valores:
  - ✓ **T** → arriba a la derecha de la imagen
  - ✓ **M** → en el centro a la derecha de la imagen
  - ✓ **B** → en la parte inferior de la imagen a la derecha de la misma
  - ✓ **N** → en una nueva línea debajo de la imagen





- El parámetro **remuestreo** es un valor booleano que está íntimamente relacionado con **resolucion** que es el parámetro que le sigue. La visualización en pantalla no se ve afectada por estos parámetros. Sólo se aconseja para documentos que sean impresos. Si **remuestreo** es **false**, la imagen no será remuestreada y mantendrá su resolución original, pero si se le asigna el valor **true** o **1**, remuestreará la imagen para ajustarla a la resolución especificada en **resolucion**, siempre que el valor de esta (en puntos por pulgada, ppi) sea superior al de la imagen original. En caso de que **remuestreo** tuviera valor **2** el proceso se efectuaría tanto si aumenta como si disminuye la resolución original.
- Con el parámetro **alineacion** es posible alinear una imagen a la izquierda, centro o derecha (**L**, **C** o **R**) en la línea en que se encuentre el punto de inserción.
- El parámetro **imk** se refiere a la creación de áreas de transparencia en imágenes, para lo que resulta mejor utilizar imágenes en formato **PNG** ó **GIF**, que ya permiten transparencias.
- La opción **imgk** puede tener utilidad cuando se trata de recortar imágenes siguiendo el borde de una figura cerrada.
- El parámetro **borde** permite asignarle un borde a la imagen. Valores posibles: **0** → sin bordes, **1** → borde por defecto, **array** → como el descrito para el caso de los rectángulos.
- Si **proporcionalidad** es **false** la imagen se acomoda a lo especificado en *ancho* y *alto*. Si **no es false**, la imagen mantiene la proporcionalidad.
- Si **ocultar** es **true**, no se ve la imagen. Si es **false** (valor por defecto), se ve.

# Imágenes

## Transparencias y modos de superposición

Para añadir transparencias a los elementos de nuestros PDF podemos usar este método:

```
$documento->SetAlpha(opacidad, modo)
```

El primer parámetro, **opacidad** es un valor comprendido entre 0 y 1 (default 1, opacidad total).

El segundo parámetro, **modo**, permite uno de los siguientes valores: **Normal, Multiply, Screen, Overlay, Darken, Lighten, Color Dodge, Color Burn, Soft Light, Hard Light, Difference, Exclusion, Hue, Saturation, Color, Luminosity**.

## Calidad de las imágenes

En caso de incluir imágenes **JPG** es posible establecer su grado de compresión. Por defecto, la clase establece un 75%. Si queremos modificarlo podemos usar el método siguiente.

```
$documento->setJPEGQuality(porcentaje)
```

# Imágenes

## Encabezados y pies de página

Antes de incluir encabezado y pie de página en un documento, tendremos en cuenta:

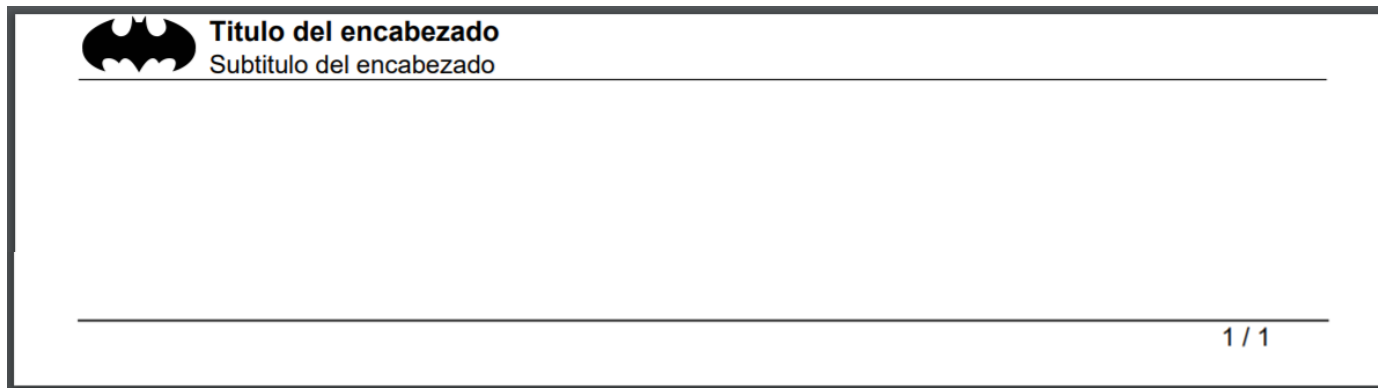
- Los métodos `setPrintHeader` y `setPrintFooter` han de estar configuradas como `true` (opción por defecto) para permitir que se visualicen los encabezados y pies de página.
- El método `setHeaderMargin` debe tener asignado un valor numérico positivo, que será la distancia del margen superior de la página.
- El método `setFooterMargin` debería estar establecido ya que con su valor por defecto (cero) empezaría a escribir el pie de página debajo del borde inferior del papel.

```
$documento->setHeaderFont( array(nombre, estilo, tamaño) ); // array('times', 'I', 12);  
$documento->setFooterFont( array(nombre, estilo, tamaño) );
```

# Imágenes

```
$documento->setHeaderData(logo, ancho, titulo, subtítulo);
```

Con este método crearemos un **logo** con el nombre de imagen, **ancho** que queremos que ocupe esa imagen, **titulo** del encabezado, **subtitulo** que iría debajo de la anterior.



El pie de página se incluye de forma automática y contiene el número de página y el número total de páginas. Se dibuja una raya horizontal delimitadora del encabezado y pie de página que ocupa todo el espacio entre márgenes. Los pies de página se alinean al margen derecho y los encabezados al margen izquierdo.

## EJERCICIO



Crea una carpeta de nombre **ud07ejer02** que incluya los 6 archivos siguientes: *cinelogin.php*, *cinepagina.php*, *cinecomprada.php*, *cinepdfentrada.php*, *cineregistro.php* y *cinelogout.php*


Esta actividad servirá para repasar conceptos vistos en unidades anteriores y, también, para generar un PDF en una situación “real”.

- Realizaremos una aplicación web para un humilde cine (que dispone de una única sala con 50 butacas).
- Podremos registrarnos y hacer login en la web para acceder a una página donde se nos mostrarán las butacas disponibles (en amarillo) y las ocupadas (en rojo) para cada película. No existen diferentes horarios, solo películas.
- Podremos cambiar de película con un menú desplegable (select).
- Si hacemos click en una butaca amarilla (en las otras no podremos) iremos a otra página donde se nos dirá que hemos comprado la entrada y desde la cual podremos generar un PDF de la entrada adquirida.




Crear base de datos

bdcine latin1\_spanish\_ci Crear

 Importar



- ☐ Tabla
- ☐ películas
- ☐ usuarios
- 2 tablas**

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo
<input type="checkbox"/> 1	usuario 	varchar(50)	utf8_general_ci		No
<input type="checkbox"/> 2	contrasena	varchar(100)	utf8_general_ci		No

- Uno para el **título** de la peli (clave primaria).
- Otro llamado **butacas**. Se trata de una cadena de 50 caracteres que sirve para saber, en cada película, las butacas que están libres (1) y cuáles están ocupadas (0). Por defecto, es una cadena con 50 ‘unos’ para que estén todas libres.

## cinelogin.php

- Esta primera página servirá para hacer el login.
- Se avisa si hay campos vacíos, y los otros se conservan.
- Si el login es correcto, vamos a **cinepagina.php**.



Introduce tus credenciales para entrar

Usuario

Contraseña

¿Aún no tienes cuenta? ¡[Regístrate!](#)

Introduce tus credenciales para entrar

Usuario

Contraseña  ¡Debes introducir una contraseña!

Introduce tus credenciales para entrar

Usuario  ¡Debes introducir un nombre de usuario!

Contraseña

Datos incorrectos. Prueba de nuevo


Usuario

Contraseña



## cineregistro.php

- En la página de registro se avisa si hay campos vacíos y los otros se conservan.
- Si ambos están rellenos, el usuario se añade a la BD (codificando contraseña con sha1) y se redirige a la página **cinelogin.php**



### Registrarme

Usuario:

Contraseña:

### Registrarme

Usuario:

Contraseña:  ¡Debe introducir un password!

### Registrarme

Usuario:  ¡Debe introducir un nombre de usuario!

Contraseña:

## cinelogout.php

Cierra sesión y redirige directamente a **cinelogin.php**

En la página principal, como se puede apreciar, aparece arriba un mensaje de bienvenida y un enlace al logout.

A continuación, el título *Comprar entradas* y un subtítulo con el nombre de la película actual (la primera vez, como no se ha indicado ninguna, por defecto podéis mostrar la primera película de la base de datos, o mostrar solamente el formulario).

Luego aparece la imagen de la pantalla y después 50 imágenes de butacas colocadas en 5 filas de 10. Más tarde se explicará mejor la gestión de butacas.

Es importante tener claro que **EL SELECT DE PELICULAS ES DINÁMICO** (sus opciones serán las películas que haya en ese momento en la BD).



Por tanto, si elegimos *Titanic* y damos al botón de submit, la página cambia.

Titanic ▼ Seleccionar película



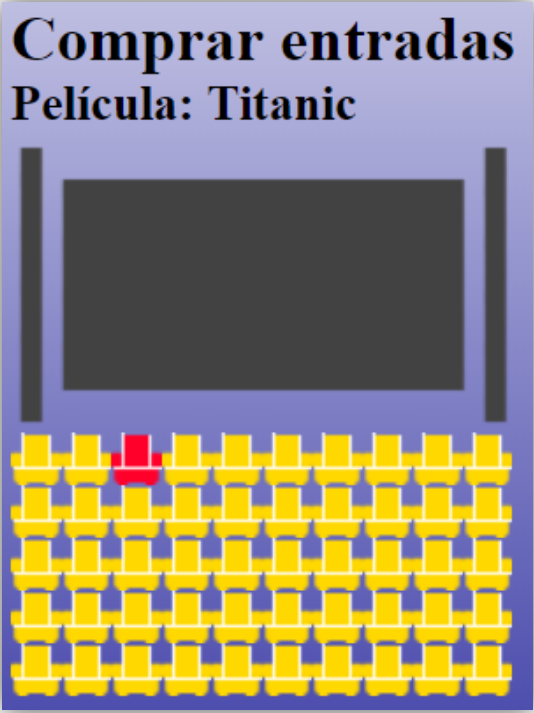
Usaremos GET porque nos vendrá bien para otra parte del ejercicio.

Aquí podremos observar cómo cambian las butacas (si es que existen reservas).

En el caso de esta imagen de la izquierda, si miramos la BD, Titanic tiene un 0 en la tercera posición de su array de butacas. Por eso, la tercera butaca aparece de color rojo.

[illegible]

Recuerda que en el desplegable, la película elegida se ha conservado gracias al atributo **selected** en la `<option>`.



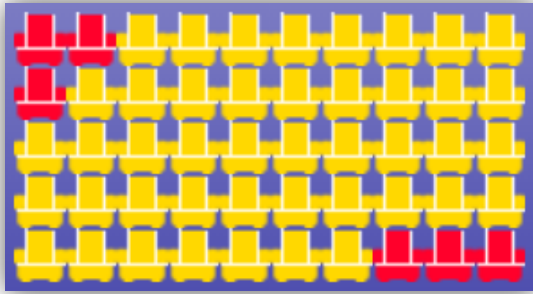
Entonces, por ejemplo, si en un futuro en la base de datos Titanic estuviera así...

[illegible]

El proceso para mostrar butacas sería el siguiente:

De la película actual, se recorre el array **butacas**:  

- Si nos encontramos un **0**, ponemos la imagen de una butaca roja. Será una imagen normal.
- Si nos encontramos un **1**, ponemos la imagen de la silla amarilla, que será a su vez un enlace a la página **cinecomprada.php**, pasándole además por GET el número de butaca (o el nº de fila y nº de butaca).



...sus sillas se verían así.

**cinecomprada.php**

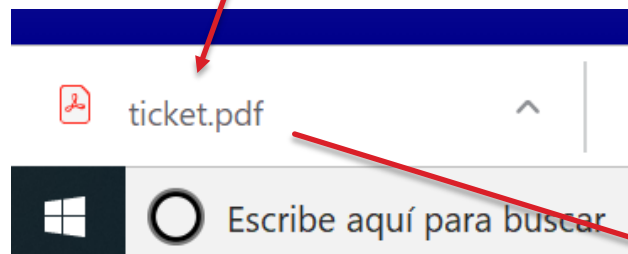
Si hacemos un click en una butaca (recuerda que sólo podemos hacer click en las amarillas) nos lleva a esta otra página, pasando por GET el título, el número de fila y número de silla.

En esta página se hace el UPDATE, cambiando la disponibilidad de las butacas de la película en la que estamos. Cambiará sólo un dígito de 1 a 0, de la butaca que se ha comprado.

La palabra [AQUÍ](#) descargará el PDF (se explica en la siguiente página), y el botón de abajo nos devuelve a **cinepagina.php** para seguir comprando (volveremos a dicha página en la misma película en la que habíamos comprado la entrada).



cinepdfentrada.php



¡Enhorabuena!

Has adquirido una entrada. Para descargarla, haz click [AQUÍ](#)

Haciendo clic en [AQUÍ](#) se descarga directamente un PDF llamado *ticket.pdf* con este formato.

Ocurre gracias a que dicha palabra es un enlace a `cinepdfentrada.php`, que recibe también por GET los tres valores y “escupe” ese documento.

El PDF se ha creado con elementos HTML: una imagen y una tabla (con la información que viene del GET).





## Warning:

- Respecto a la posición de las butacas, hay que diferenciar entre:
  - La posición que le diremos al usuario cuando compre la entrada, que está dada por:
    - ✓ El número de fila (de 1 a 5)
    - ✓ El número de butaca (de 1 a 10)
  - La posición en el campo **butacas** de la BD, que está dada por:
    - ✓ Un número del 0 al 49
- Por GET pasamos la fila y la butaca, pero de 0 a 4 y de 0 a 9.
  - Por ejemplo, para la 8ª butaca de la 3ª fila, tendríamos un 2 en fila y un 7 en butaca. ¿En qué posición del campo **butacas** cambio un 1 por un 0?  
 $(2 \times 10) + 7 = 27 \rightarrow$  Posición [27] del string.
- Esta es una forma de hacerlo, pero no es la única. Podría hacerse pasando por GET el número absoluto (de 0 a 49), o pasando por GET la fila y columna ya de 1 a 5 y de 1 a 10.
- Haremos que nuestra web ‘escupa’ un PDF con la entrada de cine que acaba de comprar el usuario. Para ello, el orden de acciones en nuestro archivo **cinepdfentrada.php** será:
  - Incluir la librería TCPDF; Crear un nuevo objeto TCPDF; Desactivar el encabezado y el pie de página; Establecer título del documento; Añadir una nueva página al documento; Usar la función WriteHTML para añadir el contenido; Llamar a Output con "D" para que se descargue el PDF.