

# Grafik İşlemleri

# Grafik işlemleri

- PHP sadece HTML tabanlı çıktı almak için kullanılmaz. Çeşitli formatlardaki (gif,png,jpeg,wbmp,xpm)resim dosyaları oluşturmak, bu dosyalar üzerinde değişiklik yapmak çeşitli grafik tabanlı nesnelere oluşturmak için de kullanılmaktadır. PHP içerisinde grafik işlemlerini kullanabilmek için GD kütüphanesinin eklenmesi gerekmektedir.
- Sunucu üzerinde GD kütüphanesi yoksa <http://www.libgd.org/> adresinden temin edilebilir. Kullanılabilecek resim formatları yüklü olan GD kütüphanesine bağlıdır. Örneğin; gd-1.6'dan eski olan GD kütüphanelerinde GIF formatı kullanılabilirken PNG formatını destekleyememektedir.

- Aşağıdaki tabloda desteklenen resim biçimleri ve GD sürümleri verilmiştir.

- **Biçim**

- **Açıklama**

- GIF destekler. GD-1.6 dan eski ve GD-2.0.28'den yeni GD kütüphaneleri
- JPEG Tüm GD kütüphaneleri desteklemektedir.
- PNG Sadece GD-1.6'dan yeni GD kütüphaneleri desteklemektedir.
- Php.ini dosyası içinde bulunan **extension=php\_gd2.dll** satırının önünde bulunan noktalı virgül kaldırıldığı zaman GD kütüphanesi aktif olacaktır.
- php.ini dosyası içerisinde böyle bir satır yoksa **extension=php\_gd2.dll** ifadesini dosyanın sonuna ekleyerek kaydedip sunucuyu yeniden başlatmak gereklidir.



## image\_types( ) Fonksiyonu

- Kullanılan PHP versiyonu tarafından desteklenen resim formatlarını öğrenmek için **image\_types()** fonksiyonu kullanılmaktadır.
- PHP ile ilişkilendirilmiş GD kütüphanesinin desteklediği resim biçimleri ile ilgili bir bit alanı döndürür.
- Bu alanla sınırlanabilecek sabitlerin isimleri: **IMG\_GIF | IMG\_JPG | IMG\_PNG | IMG\_WBMP | IMG\_XPM.** dir.

# Örnek

- Yüklü olan PHP'nin desteklediği resim formatlarını belirleyelim.

```
<?php
```

```
    if (imagetypes()&IMG_PNG) echo "PNG desteği var<br>";  
    else echo "PNG desteği yok <br>";
```

```
    if(imagetypes()&IMG_JPG) echo "JPG desteği var<br>";  
    else echo "JPG desteği yok<br>";
```

```
    if(imagetypes()&IMG_GIF) echo "GIF desteği var <br>";  
    else echo "GIF desteği yok<br>";
```

```
?>
```



Grafik oluřturmaya bařlarken ilk olarak bu sayfanın grafik olarak kullanılacađı **header()** fonksiyonu yardımıyla belirtilmesi gerekir.

- `<?php`
- `header ("Content-type:image/jpeg");`  
`?>`
- Yukardaki kodlama jpeg formatında bir grafik iřlemi yapılacađını gösteriyor. Çalışma PNG formatında olursa **image/png** yazılması gerekir.

# Resim Oluřturma

- PHP'nin GD kütüphanesi yardımıyla sıfırdan bir resim oluşturulabilir ve istenirse oluşturulan resim istenilen formatta resim dosya olarak kaydedilir.
- Örneğin; sitenizde kullandığınız düğmeleri her düğmeyi ayrı ayrı oluşturmak yerine hazırlayacağınız bir fonksiyon ve GD kütüphanesinin fonksiyonları yardımıyla resim işleme programlarına ihtiyaç duymadan gerçek zamanlı oluşturabilirsiniz.

# Resim Oluřturma Fonksiyonları



# imagecreate( )

Boyutları belirtilen boş bir resim alanı oluşturur.

Fonksiyonun çalışması sırasında hata oluşursa FALSE değeri döndürür aksi takdirde resim değişkenini döndürür.

- **Kullanım Şekli ;**
- `Resim_değişkeni=imagecreate(genişlik_değeri,yükseklik_değeri)`
- Belirtilen genişlik ve yükseklik değerlerinin türü pixel'dir.
- **Örnek;** 100\*50 boyutlarında jpeg formatında boş bir resim oluşturalım.
- `<?PHP`
- `header("Content-type:image/jpeg");`
- `$resim=imagecreate(100,50);`
- `imagejpeg($resim);`
- `?>`
- 





# imagecreatetruecolor( )

- Boyutları belirtilen gerçek renkli boş bir resim alanı oluşturur. Fonksiyonun çalışması sırasında hata oluşursa FALSE değeri döndürür aksi takdirde resim değişkenini döndürür.

**imagecreate** fonksiyonu yerine bu fonksiyonun kullanılması tavsiye edilmektedir.

## Kullanım Şekli;

resim\_değişkeni =imagecreatetruecolor(genişlik\_değeri,yükseklik\_değeri)

## Örnek;

```
header("Content-type:image/jpeg");
$resim=imagecreatetruecolor (100,50);
imagejpeg($resim);
?>
```



# imagejpeg( )

- Belirtilen boyutlar ve özelliklere göre bir resmi JPEG formatında oluşturmak için kullanılır. Oluşturulan resim sadece sayfada görüntülenebileceği gibi dosya olarak saklanabilir.
- **Kullanım şekli;**
- `Imagejpeg(resim_değişkeni,dosya_ismi,kalite)`
- “resim\_değişkeni” resim alanı oluşturmak için kullanılan `imagecreatetrucolor` fonksiyonundan dönen değer.
- “kalite” isteğe bağlı olarak 0(en kötü-en küçük boyut)-100(en iyi-en fazla boyut)arasında değerler alabilir. Kullanılması zorunlu değildir.
- Oluşturulan resim dosya olarak saklanmayacaksa veya kalitesi üzerinde değişiklik yapılmayacaksa bir parametre ile çalışabilir.

# Örnek;

100\*50 boyutlarında jpeg formatında boş bir resim oluşturalım ve a.jpg isminde kaydedelim.

- `<?PHP`
- `$resim=imagecreatetruecolor(100,50);`  
`/* a.jpg isminde ve 75 kalite değerinde bir resmi */`
- `/* dosyamızla aynı yere oluşturacaktır. */`
- `imagejpeg($resim,"a.jpg", "75");`
- `imagedestroy($resim); //Resmi bellekten sil`  
`?>`



# imagegif()

Belirtilen boyutlar ve özelliklere göre bir resmi GIF formatında oluşturmak için kullanılır. Oluşturulan resim sadece sayfada görüntülenebileceği gibi dosya olarak saklanabilir.

Kullanım şekli;

```
imagegif(resim_değişkeni,dosya_ismi)
```

“resim\_değişkeni” resim alanı oluşturmak için kullanılan imagecreatetruecolor fonksiyonundan dönen değer.

“dosya\_ismi” oluşturulan resim dosya olarak saklanacaksa kullanılır. Dosya ismi yoluyla birlikte tırnak içerisinde yazılır. Kullanılması zorunlu değildir. Resim dosya olarak saklandığında sayfada görünmeyecektir. Ayrıca çağırılması gerekir.



# Örnek;

- 100\*50 boyutlarında jpeg formatında boş bir resim oluşturalım ve a.gif isminde kaydedelim.
- **<?PHP**
- `$resim =imagecreatetruecolor(100,50);`
- `/* a.gif isminde bir resmi */`
- `/* dosyamızla aynı yere oluşturacaklardır. */`
- `imagegif($resim,"a.gif", "75");`
- `imagedestroy($resim); //Resmi bellekten sil`
- `?>`

# imagewbmp( )

- Belirtilen boyutlar ve özelliklere göre bir resmi WBMP formatında oluşturmak için kullanılır. Oluşturulan resim sadece sayfada görüntülenebileceği gibi dosya olarak saklanır.
- **Kullanım şekli;**
- `Imagewbmp(resim_değişkeni,dosya_ismi,arka plan)`
- “resim\_değişkeni” resim alanı oluşturmak için kullanılan `imagecreatetruecolor`
- Fonksiyonundan dönen değer.
- “dosya\_ismi” oluşturulan resim dosya olarak saklanacaksa kullanılır. Dosya ismi yoluyla birlikte tırnak içerisinde yazılır. Kullanılması zorunlu değildir. Resim dosya olarak saklandığında sayfada görünmeyecektir.



# Örnek;

- 100\*50 boyutlarında wbmp formatında ve oluşturulan zemin renginde bir resim oluşturalım ve sayfada görüntüleyelim.
- **<?php**
- `$resim=imagecreatetruecolor(100,50);`
- `$zemin_rengi=imagecolorallocate($resim,255,120,50);`
- `imagewbmp($resim,NULL,$zemin_rengi);`
- `imagedestroy($resim);`
- **?>**
-



# imagepng( )

- Belirtilen boyutlar ve özelliklere göre bir resmi PNG formatında oluşturmak için kullanılır. Oluşturulan resim sadece sayfada görüntülenebileceği gibi dosya olarak saklanır.
- **Kullanım Şekli;**
- `Imagepng(resim_değişkeni,dosya_ismi,kalite)`
- “resim\_değişkeni” resim alanı oluşturmak için kullanılan `imagecreatetruecolor`
- Fonksiyonundan dönen değer.
- “dosya\_ismi” oluşturulan resim dosya olarak saklanacaksa kullanılır. Dosya ismi yoluyla birlikte tırnak içerisinde yazılır. Kullanılması zorunlu değildir. Resim dosya olarak saklandığında sayfada görünmeyecektir.
- “kalite” sıkıştırma düzeyidir. 0-9 arası değer alır.

# Örnek;

- 100\*50 boyutlarında PNG formatında boş bir resim oluşturalım ve a.png isminde kaydedelim.
- `<?PHP`
- `$resim =imagecreatetruecolor(100,50);`
- `/* a.png isminde bir resmi */`
- `/* dosyamızla aynı yere oluşturacaklardır */`
- `imagepng($resim,"a.png","2");`
- `imagedestroy($resim); //Resmi bellekten sil`
- `?>`



# imagedestroy ( )

- Oluřturulan resimler bellekte yer kaplayacađı için oluřturduktan sonra bellekten silmek amacıyla kullanılır. Resim sadece bellekten silinir sayfada görünmeye devam eder.
- Örnekler olarak resim oluřturma fonksiyonlarında kullanılmıřtır.



# Renk Kullanımı

# imagecolorallocate( )

- Oluşturulacak resim için renk tanımlamada kullanılır.
- **Kullanım Şekli;**
- **\$renk\_değişkeni=imagecolorallocate** (resim\_değişkeni,kırmızı,yeşil,mavi);
- “renk\_değişkeni” resim alanı oluşturmak için kullanılan imagecreatetruecolor veya imagecreate fonksiyonundan dönen değer.
- “kırmızı”, “yeşil”, “mavi” renklerin 0-255 arasındaki değerleridir.
- “renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.
- Resim oluşturmak için **imagecreate** fonksiyonu kullanılırsa yazılan kodlar içerisindeki ilk tanımlanan renk herhangi bir işleme gerek kalmadan oluşturulan resim için zemin rengi olarak kullanılır.

# Örnek;

- Resmi imagecreate fonksiyonu ile oluşturularak yeşil bir renk tanımlayalım. Kodlar çalıştırıldığında görüleceği gibi resmin zemin rengi yeşil olacaktır.

- `<?php`

- `$resim =imagecreate(100,50);`  
`$yesil=imagecolorallocate ($resim,0,255,0);`

- `imagejpeg($resim);`  
`imagedestroy($resim);`

- `?>`





# imagefill( )

- Belirtilen koordinattan başlayarak resmi belirtilen renk ile boyar .
- **Kullanım Şekli;**
- `Imagefill(resim_değişkeni,x,y,renk_değişkeni)`
- “resim\_değişkeni”resim alanı oluşturmak için kullanılan `imagecreatetruecolor` veya `imagecreate` fonksiyonundan dönen değer.
- “x” başlangıç noktasının x konumu.
- “y”başlangıç noktasının Y konumu.
- “renk\_değişkeni” `imagecolorallocate` fonksiyonu ile oluşturulan renk değişkeni.
  
- Oluşturulan resmi parametre olarak verilen x ve y koordinatlarından başlayarak diğer parametre olan renk ile boyar.Tek bir resim oluşturuluyorsa veya tek bir alan boyanacaksa x,y koordinatları 0,0 verilir.

# Örnek;

- Aşağıdaki örneğimizde imagecreatetruecolor fonksiyonu ile resim oluşturulur ve imagefill fonksiyonu yardımıyla resmin kırmızı olmasını sağlar.
- `<?php`
- `$resim=imagecreatetruecolor(100,50);`
- `$renk_kirmizi=imagecolorallocate($resim,255,0,0);`
- `Imagefill($resim,0,0,$renk_kirmizi);`
- `Imagepng($resim);`
- `imagedestroy($resim);`
- `?>`







# imagefilledrectangle( )

- İçi dolu dikdörtgen çizmek için kullanılır.
- **Kullanım şekli;**
- `Imagefilledrectangle(resim_değişkeni,x1,y1,x2,y2, renk_değişkeni)`
- “resim\_değişkeni” resim alanı oluşturmak için kullanılan `imagecreatetruecolor` veya `imagecreate` fonksiyonundan dönen değer.
- “x1” dörtgenin başlangıç noktasının x koordinatı.
- “y1” dörtgenin başlangıç noktasının y koordinatı.
- “x2” dörtgenin bitiş noktasının x koordinatı
- “y2” dörtgenin bitiş noktasının y koordinatı.
- “renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.

# Örnek;

- 100\*50 boyutundaki resim alanı içerisine 80\*30 ebatında kırmızı renkte dörtgen çizelim..
- **<?php**
- `$resim=imagecreatetruecolor(100,50);`
- `$kirmizi=imagecolorallocate($resim,255,0,0);`
- 
- *//Kırmızı renkte dikdörtgen çizelim.*
- `Imagefilledrectangle($resim,10,10,90,40,$kirmizi);`

```
//Resmi oluşturalım  
imagepng($resim);
```

```
//Resmi bellekten silelim  
imagedestroy($resim);  
?>
```



# Örnek:

- 425\*50 boyutundaki resim alanı içerisinde dört farklı renkte dörtgen çizelim.
- **<?php**
- `$resim=imagecreatetruecolor(425,50);`
- `/* Beyaz,Kırmızı,Yeşil,Mavi renklerini oluşturalım */`
- `$beyaz=imagecolorallocate($resim,255,255,255);`
- `$kirmizi=imagecolorallocate($resim,255,0,0);`
- `$yesil=imagecolorallocate($resim,0,255,0);`
- `$mavi=imagecolorallocate($resim,0,0,255);`
  
- `/* Resmi oluşturalım */`
- **`Imagefilledrectangle($resim,5,5,105,45,$beyaz);`**
- **`Imagefilledrectangle($resim,5,5,105,45,$kirmizi);`**
- **`Imagefilledrectangle($resim,5,5,105,45,$yesil);`**
- **`Imagefilledrectangle($resim,5,5,105,45,$mavi);`**
- `/* Resmi oluşturalım */`
- `Imagepng($resim);`
- `/* Resmi bellekten silelim */`

?>





# imagefilledellipse( )

- İçi dolu elips çizmek için kullanılır. Daire çizmek için genişlik ve yükseklik parametreleri verilmelidir.
- **Kullanım Şekli;**
- `Imagefilledellipse(resim_değişkeni,x,y,genişlik,yükseklik,renk_değişkeni)`
- “resim\_değişkeni” resim alanı oluşturmak için kullanılan `imagecreatetruecolor` veya `imagecreate` fonksiyonundan dönen değer.
- “x” elipsin merkezinin x koordinatı.
- “y” elipsin merkezinin y koordinatı.
- “genişlik” elipsin genişliği.
- “yükseklik” elipsin genişliği.
- “renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.

# Örnek;

- 300\*100 boyutundaki resim alanı içerisinde kesim şeklinde iki adet elips çizelim.

```
<?php
```

```
  $resim=imagecreatetruecolor(300,100);
```

```
  $zemin_rengi=imagecolorallocate($resim,222,150,10)
```

```
  imagefill($resim,0,0,$zemin_rengi);
```

```
  //Elipsin rengini oluşturalım
```

```
  $yesil=imagecolorallocate($resim,0,255,0);
```

```
  $red=imagecolorallocate($resim,255,0,0);
```

```
  //Elipsi çizelim
```

```
  imagefilledellipse($resim,100,50,200,90,$yesil);
```

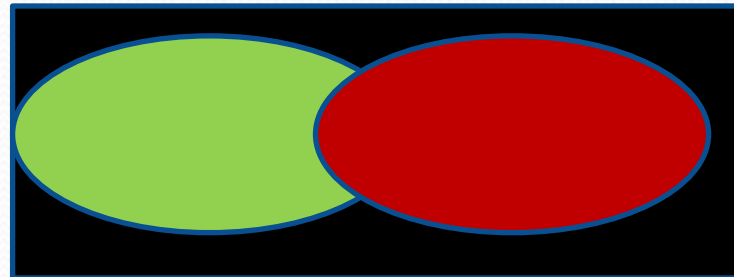
```
  imagefilledellipse($resim,200,50,200,90,$red);
```

```
  //Resmi oluşturalım
```

```
  imagepng($resim);
```

```
  //Resmi bellekten silelim
```

```
?>
```





# imagefilledpolygon( )

İçi dolu çokgen çizmek için kullanılır.

## **Kullanım Şekli;**

Imagefilledpolygon(resim\_değişkeni,noktalar,nokta\_sayisi,renk\_değişkeni)

“*resim\_değişkeni*”resim alanı oluşturmak için kullanılan imagecreatetruecolor

Veya imagecreate fonksiyonundan dönen değer.

“*noktalar*”çokgeni oluşturan noktaların koordinatlarını içeren dizi değişken.

“*nokta\_sayısı*”çokgeni oluşturan noktaların sayısıdır.3'den fazla olmalıdır.

“*renk\_değişkeni*”tanımlanan rengin tutulması için kullanılan değişkendir.



# Örnek;

- Çokgen ve elips kullanarak Türk bayrağını çizelim .

```
<?php
```

```
/* Yıldızın birinci bölümünü oluşturacak noktalar */
```

```
$noktalar 1 =array(
```

```
    200,70          //Nokta 1 (x,y)
```

```
    209,130        //Nokta 2 (x,y)
```

```
    220,105        //Nokta 3 (x,y),
```

```
    248,110        //Nokta 4 (x,y)
```

```
);
```

```
/* Yıldızın ikinci bölümünü oluşturacak noktalar */  
$noktalar2=array(  
    182,105,          //Nokta 1(x,x)  
    235,70 ,         //Nokta 2(x,x)  
    220,108,         //Nokta 3(x,x)  
);
```

```
/* Bayrağın ebatları */  
$resim=imagecreatetruecolor(300,200);
```

```
/* Ay, Yıldız ve Zemin Rengi */  
$beyaz=imagecolorallocate($resim,255,255,255);  
$kirmizi=imagecolorallocate($resim,255,0,0);
```

```
/* İki elips üst üste getirilerek ay oluşturması */  
imagefilledellipse($resim,110,100,155,$beyaz);  
imagefilledellipse($resim,110,100,155,$beyaz);
```



```
/* Noktaları belirtilen çokgenler ile */  
/* yıldızın oluşturulması */  
imagefilledpolygon($resim,$noktalar 1, 4, $beyaz);  
imagefilledpolygon($resim,$noktalar 2, 3, $beyaz);
```

```
/* Bayrağın zemin renginin doldurulması */  
imagefill($resim,0,0,$kirmizi);
```

```
/* Bayrağın oluşturulması */  
imagejpeg($resim,null,100);  
imagedestroy($resim);
```





# imageline( )

Çizgi çizmek için kullanılır.Çizginin kalınlığı değiştirilemez.Kalın çizgi çizmek İsteniyorsa yakın koordinatlar verilerek yapılabilir.

## **Kullanım Şekli;**

İmageline(resim\_değişkeni,x1,y1,x2,y2,renk\_değişkeni)

“*resim\_değişkeni*” resim alanı oluşturmak için kullanılan imagecreatetruecolor

Veya imagecreate fonksiyonundan dönen değer.

“x1” çizginin başlangıç noktasının x koordinatı.

“y1” çizginin başlangıç noktasının y koordinatı.

“x2” çizginin bitiş noktasının x koordinatı .

“y2” çizginin bitiş noktasının y koordinatı.

“renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.

# Örnek;

Aşağıdaki örneğimiz çizgi kullanarak dörtgen oluşturacak ve içerisine birisi kalın iki çizgiyi çapraz olarak çizecektir.

```
<?php
```

```
$resim=imagecreate(201,101 );
```

```
$beyaz=imagecolorallocate($resim,255,255,255 );
```

```
$kirmizi=imagecolorallocate($resim,255,0,0);
```

```
$mor=imagecolorallocate($resim,100,50,100);
```

```
imagefill($resim,0,0,$beyaz );
```

```
imageline($resim,0,0,200,0,$kirmizi); //üst kenar çizildi
```

```
imageline($resim,0,100,200,100,$kirmizi); //alt kenar çizildi
```

```
imageline($resim,0,0,0,100,$kirmizi); //sol kenar çizildi
```

```
imageline($resim,200,0,200,100,$kirmizi); //sağ kenar çizildi
```

```
imageline($resim,0,0,200,100,$mor); //ilk çapraz çizildi
```

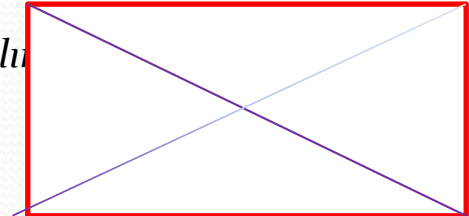
```
imageline($resim,0,100,200,0,$mor); //ikinci çapraz çizildi
```

```
imageline($resim,1,100,200,1,$mor); //ikinci çapraz biraz kalın
```

```
Imagejpeg($resim); //resim oluşturuldu
```

```
ImageDestroy($resim); //resim bellekten silindi
```

```
?>
```





# imagestring( )

Yatay olarak grafik tabanlı yazı yazdırmak için kullanılır.

## **Kullanım Şekli**

Imagestring(resim\_değişkeni,yazı\_tipi,x,y,renk\_değişkeni)

“resim\_değişkeni” resim alanı oluşturmak için kullanılan imagecreatetruecolor veya imagecreate fonksiyonundan dönen değer.

“yazı\_tipi”1,2,3,4,5 değerlerinden birisini alır.Değer büyüdükçe yazı tipi de büyür.

“x” yazının üst sol köşesinin x koordinatı

“y” yazının üst sol köşesinin y koordinatı

“yazı” oluşturulacak yazı.

“renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.



# Örnek;

Sayfamıza “Merhaba Dünya” yazan bir resim oluşturalım.

```
<?PHP
```

```
    $resim=imagecreate(150,30);
```

```
    $zemin_rengi=imagecolorallocate($resim,255,100,55);
```

```
    $yazi_rengi=imagecolorallocate($resim,255,255,255);
```

```
    //Yazıyı oluşturalım
```

```
    imagestring($resim,5,15,5,"Merhaba Dünya",$yazi_rengi);
```

```
    //Resmi oluşturalım
```

```
    imagepng($resim);
```

```
    imagedestroy($resim);
```

```
?>
```



MERHABA DÜNYA

# imagestringup( )

90 derece döndürülmüş olarak grafik tabanlı yazı yazdırmak için kullanılır.

## **Kullanım Şekli;**

imagestringup(resim\_değişkeni,yazı\_tipi,x,y,yazi,renk\_değişkeni)

“resim\_değişkeni” resim alanı oluşturmak için kullanılan imagecreatetruecolor  
Veya imagecreate fonksiyonundan dönen değer.

“yazı\_tipi” 1,2,3,4,5 değerlerinden birisini alır.Değer büyüdükçe yazı tipi de büyür.

“x “yazının üst sol köşesinin x koordinatı

“y” yazının üst sol köşesinin y koordinatı

“yazi” oluşturulacak yazı

“renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.



# Örnek;

Bu örneğimiz Ankara Üniv. yazısını yatay ve dikey yazdırılmış resmini oluşturur.

```
<?php
```

```
    $resim=imagecreate(201,101);
```

```
    $beyaz=imagecolorallocate($resim,255,255,255);
```

```
    $kirmizi=imagecolorallocate($resim,255,0,0);
```

```
    $mor=imagecolorallocate($resim,100,50,100);
```

```
    imagefill($resim,0,0,$beyaz);
```

```
    imagestring($resim,5,30,10," Ankara Üniv. ",$kirmizi); //üst yatay yazı
```

```
    imagestring($resim,5,10,100," Ankara Üniv. ",$mor); // sol dikey yazı
```

```
    imagestring($resim,5,30,85," Ankara Üniv. ",$kirmizi); //alt yatay yazı
```

```
    imagestring($resim,5,125,100," Ankara Üniv. ",$mor); //sağ dikey yazı
```

```
    imagejpeg($resim);
```

```
    imagedestroy($resim);
```

```
?>
```



# imageTtfText( )

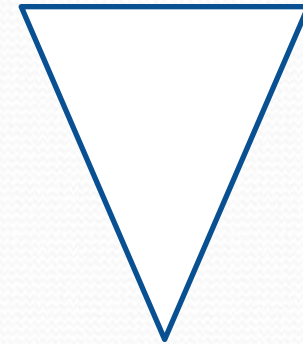
- Mevcut yazı tiplerini kullanarak grafik tabanlı yazı oluşturur.
- **Kullanım Şekli;**
- `imageTtfText(resim_değişkeni,boyut,açı,x,y,renk_değişkeni,yazı_tipi,yazı)`
- “resim\_değişkeni” resim alanı oluşturmak için kullanılan `imagecreateTrueColor` veya `imagecreate` fonksiyonundan dönen değer.
- “boyut” yazının yüksek değeri.GD1’de pixel,GD2’de punto türündedir.
- “açı” derece cinsinden açı değeri.Soldan sağa yazı için açı 0 kullanılır.Dikey için 90 derece kullanılır.İstenilen açı değeri girilebilir.
- “x” yazının üst sol köşesinin x koordinatı
- “y” yazının üst sol köşesinin y koordinatı.
  - “renk\_değişkeni” tanımlanan rengin tutulması için kullanılan değişkendir.
  - “yazı\_tipi” kullanılacak yazı tipi ttf uzantısı ile birlikte yazılır.Yazı tipi dosyası dosyanızla aynı yerde bulunmalıdır veya yazı tipi dosyasının yolu tam olarak belirtilmelidir.
  - “yazı” oluşturulacak yazı.

# Örnek;

- " Ankara Üniv. " yazısını üç farklı şekilde ve iki farklı yazı tipinde grafik tabanlı olarak sayfaya yazdıralım.
- Not:Kullanılan iki yazı tipi dosyasını kodlarınızın bulunduğu yere kopyalamayı
- Unutmayınız.
- <?php
- \$resim=imagecreatetruecolor(250,200);//Resmi boş olarak oluşturalım
- //Renkleri tanımlayalım
- \$beyaz=imagecolorallocate(\$resim,255,255,255);
- \$gri=imagecolorallocate(\$resim,128,128,128);
- \$siyah=imagecolorallocate(\$resim,0,0,0);
- \$yesil=imagecolorallocate(\$resim,0,255,0);
- \$mavi=imagecolorallocate(\$resim,0,0,255);
- Imagefill(\$resim,0,0,\$beyaz);



- \$metin='Ankara Ün. '; //Görüntülenecek yazı
- \$font='broadw.ttf'; //yazı tipi dosyası ismi
- \$font2='comic.ttf'; //yazı tipi dosyası ismi
  
- İmagettfext(\$resim,20,0,27,21,\$gri,\$font,\$metin); //Yazıya gölge verelim
- İmagettfext(\$resim,20,0,26,20,\$siyah,\$font,\$metin); //Yazıyı ekleyelim
  
- //Çapraz yazıları ekleyelim
- imagettftext(\$resim,20,-60,10,35,\$mavi,\$font2,\$metin);
- imagettftext(\$resim,20,60,130,170,\$yesil,\$font2,\$metin);
  
- imagepng(\$resim);
- İmagedestroy(\$resim);
- ?>





## Hazır Resimler Üzerinde Deęişiklik Yapmak

- Buraya kadar anlatılan grafik işlemlerinin tamamında sıfırdan başlayarak resimler oluşturuldu.
- Hazır olarak bulunan resimler üzerinde de yukarıda bahsedilen işlemlerin tamamı uygulanabilir.
- Resim oluşturulurken imagecreate veya imagecreatetruecolor fonksiyonlarını kullanmıştık.Hazır resim üzerinde deęişiklik yapılacaksa bu fonksiyonlar yerine hazır resmi çalışma alanı olarak kullanmak için aşağıdaki fonksiyonlar kullanılır.

## Hazır Resimler Üzerinde Deęişiklik Yapmak

- **imagecreatefromgif():**Mevcut olan bir gif dosyasını kullanarak resim oluşturur.
- **imagecreatefromjpeg():**Mevcut olan bir jpeg dosyasını kullanarak resim oluşturur.
- **imagecreatefrompng():**Mevcut olan bir png dosyasını kullanarak resim oluşturur.
- **imagecreatefromwbmp():**Mevcut olan bir wbmp dosyasını kullanarak resim oluşturur.
- Bu fonksiyonların tamamı parametre olarak sadece mevcut dosya adını alır.



# Örnek;

- Aşağıdaki örneğimiz png formatında var olan resim dosyamızın üzerindeki yazıyı silerek yenisini yazmaktadır.
- **<? Php**
- *//buton.png dosyamızı açalım*
- `$resim=imagecreatefrompng("buton.png");`
- *//Renk tanımlamalarımızı yapalım*
- `$beyaz=imagecolorallocate($resim,255,255,255);`
- `$yesil=imagecolorallocate($resim,0,255,0);`
- `$mavi=imagecolorallocate($resim,0,0,255);`
- *//Resim üzerinde bulunan kırmızı bölgeyi yeşile boyayalım*
- `Imagefilledrectangle($resim,4,5,145,45,$yesil);`
- *//Yeni yazımız olan "Kadromuz" yazısını yazalım*
- `Imagefttext($resim,15,0,20,30,$mavi,"verdana.ttf", "Kadromuz")`
- *//Yapılan değişiklikleri PNG formatında resim dosyası haline getirelim*
- `Imagepng($resim,"butonyeni.png");`
- `Imagedestroy($resim);`
- **?>**



Resmin Önceki Hali



Resmin Sonraki Hali

