



Notion de session

- **Données stockées sur le serveur**
 - Plus fiables que les cookies
 - Affectation d'un identifiant au client
 - Attention, problème si deux fenêtres d'un même navigateur accèdent à un site avec session.
- **Récupération de l'information avec les pages consultées**

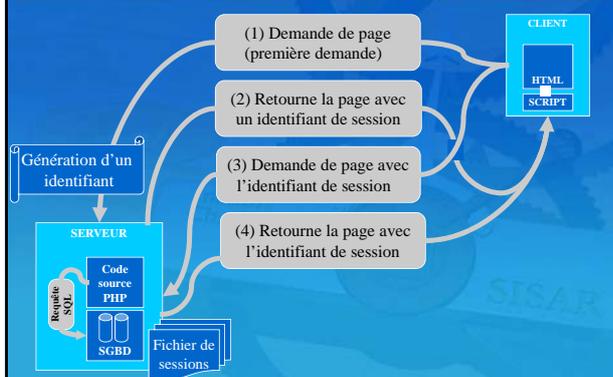
Notion de session

- **Manipulation transparente pour le développeur**
- **Mémoriser un accès au site par un utilisateur**
 - Session : mémorisation d'information dans une variable superglobale particulière
 - `$_SESSION`
- **Permet d'adapter le contenu à l'historique des accès**
 - Premier accès à la page : login
 - Accès suivant : traitement

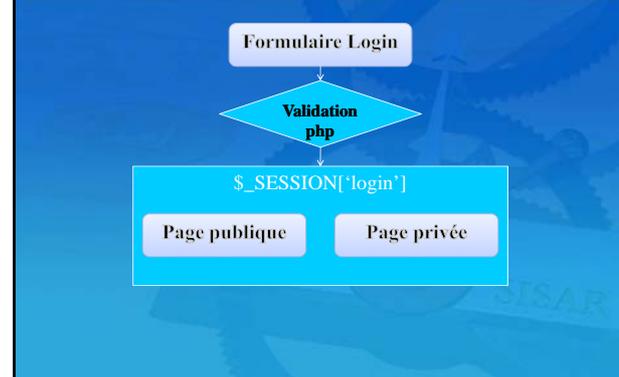
Notion de session

- **Utilisation de la fonction `session_start()`**
 - Doit être la première ligne du code php.
 - Initialise la session
 - Lecture de l'identifiant du client
 - Si il n'existe pas :
 - ✓ Création d'un identifiant unique
 - ✓ Ouverture d'un fichier contenant les informations sur le client
 - Si il existe
 - ✓ Renseigne la variable `$_SESSION` avec les informations du fichier

Système client / serveur : sessions



Système avec login



Ouverture d'une session sécurisée

```
<?php
session_start();

if (isset($_SESSION['login']))
{ unset($_SESSION['login']);
} else
{ if ($_POST['passwd']==$_POST['login'])
{ $_SESSION['login']=$_POST['login'];
}
}
header('Location: page_securisee.php');
```

Validation

```
<?php
session_start();

if (!isset($_SESSION['login'])
{ // page par défaut : formulaire de login
} else
{ // contenu de la page avec droit d'accès
}
?>
```

Page sécurisée

Traitements sécurisés

```
<?php
session_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title>Page sécurisée</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
</head>
<body bgcolor="#FFFFFF">
<?php
if (isset($_SESSION['login'])
{ echo "Bonjour '". $_SESSION['login']. "' .Voici l'image à accès sécurisé :<br>
<img src='img/illustration.jpg'><br />";
} else
{ echo "Cette page est sécurisée, saisissez un login/mot de passe valide pour
y accéder<br />";
}
echo '<a href="formulaireLogin_out.php">Login</a>';
?>
</body>
</html>
```

Exemple



Libération des données de session

- **Traitement non obligatoire**
 - Fait automatiquement par le serveur après un temps de non utilisation
 - Permet de s'assurer de la destruction des traces de la session

```
<?php
session_destroy();

unset($_SESSION);

echo "valeur = ", $_SESSION['valeur']; // n'affiche rien
?>
```

Exemple avec une base de données

```
<?php
$log = $_POST[login];
$mdp = $_POST[passwd];
session_start();

connectToDB($link);

$query = "select numen,binome,groupe from etudiantSI where
nom='$log' and Numen='$mdp'";
$result = mysql_query($query);
if (mysql_num_rows($result)=1) {
$row = mysql_fetch_array($result);
$_SESSION['binome'] = $row['binome'];
mysql_close($link);
header("Location: tpsi.php?groupe=".$row['groupe']);
} else {
unset($_SESSION['binome']);
mysql_close($link);
header("Location: index.php");
}
?>
```

Développer en PHP

Envoyer des mails



Envoi de mail depuis PHP

- Intérêts
 - Vérification d'une adresse
 - Validation d'une affectation de mot de passe
 - Permet d'éviter l'ajout automatique de compte par des robots
 - Gestion de mailing-list en PHP
 - Suivi de traitements (commande) par mail
 - News, forum...
 - Gestion d'un webmail en PHP
 - IMP : Internet Messaging Program
 - Nocc : No Carbon Copy

Envoyer des mails

- Fonction générale
 - mail(adresse, sujet, contenu);
 - Adresse peut contenir plusieurs adresses séparées par des ','
- Entêtes spécifiques
 - mail(adresse, sujet, contenu, \$entete);
 - \$entete pour modifier l'expéditeur
 - ✓ « From: x@y.z\n »
 - \$entete pour modifier l'adresse de réponse
 - ✓ « Reply-to: x@y.z\n »
 - \$entete pour modifier préciser des destinataires en copie
 - ✓ « Cc: x@y.z\n »
 - ✓ « Bcc: x@y.z\n » (copie cachée)
 - Priorités
 - ✓ « X-Priority: 1 » (5 basse, 3 normale, 1 urgente)

Envoyer des mails

- Mails au format HTML
 - Préciser le format mime dans l'entête
 - « Content-Type: text/html \n »
- Fichiers attachés
 - Plusieurs éléments dans l'entête
 - Format mime : « Content-Type: multipart/mixed »
 - Séparateur unique des éléments du mail :
boundary="Lkjhg765gfdsdcfvbg"
 - Le corps
 - Le message
 - La séparation avec le fichier : "--Lkjhg765gfdsdcfvbg"
 - Le format mime du fichier
 - le fichier lui même puis une nouvelle séparation.

Bibliothèque de gestion des mails

- Bibliothèque libre
 - HTML Mime mail
 - Encapsule les différents traitements de gestion des mails complexes dans une classe
 - ✓ HTML
 - ✓ Éléments attachés
 - ✓ ...

```
include('htmlMimeMail.php');
$mail = new htmlMimeMail();

$mail->setSubject('new mail...');

$mail->setHtml($html,$texte);
$result = $mail->send(array('piranda@univ-mlv.fr'),'smtp');
```

Recevoir des mails à l'aide de PHP

- Ouvrir un flux vers un serveur de mail
 - \$boite = imap_open(adresse, login, password)
- Lire les entêtes des mails
 - \$tab = imap_headers(\$boite);
- Fonctions de traitements des mails
 - Lecture du corps du mail : imap_body()
 - Vérifie le courrier : imap_check();
 - Supprime le mail : imap_delete();
 - Fermeture du canal : imap_close();
 - ...



Manipulation des images sous PHP

Coordonnées graphiques classiques

Coordonnées écran



Modèles de couleurs sous PHP

- Image avec palette de couleurs
 - 1 octet / pixel (index de couleur)
 - 256 couleurs possibles
 - Format GIF
- Image « true color » 24 bits
 - 3 octets / pixel (RGB)
 - 256³ couleurs possibles
 - Format JPEG
- Image « true color » 32 bits
 - 4 octets / pixel (RGBA)
 - 256³ couleurs possibles x 256 niveaux de transparence
 - Format PNG

Étapes de création d'une image

- Quatre étapes de génération d'une image
 - 1) Création du modèle de l'image
 - Allocation des ressources mémoire
 - Chargement de l'image originelle
 - 2) Travail sur le modèle
 - Ajout
 - Modification de la forme
 - 3) Fabrication de l'image finale
 - Envoi au navigateur
 - Ou création d'un fichier
 - 4) Effacement des données de la mémoire

1) Création du modèle de l'image

- Création d'une feuille blanche
 - `imagecreate(largeur, hauteur);`
 - Images 256 couleurs
 - Formats : GIF ou PNG
 - `imagecreatetruecolor(largeur, hauteur);`
 - Images en 32 bits
 - Formats : JPEG ou PNG
- Fonctions retournant un identifiant d'image

```
<?php
$width=200;
$height=200;
$image = imagecreatetruecolor($width,$height);
:
?>
```

1) Création du modèle de l'image

- Création d'un image à partir d'un fichier image
 - `imagecreatefromjpeg(fichier);`
 - `imagecreatefromgif(fichier);`
 - `imagecreatefrompng(fichier);`
 - `imagecreatefrombmp(fichier);`
 - `imagecreatefromxbm(fichier);`
 - `imagecreatefrombpm(fichier);`
- Fonctions retournant un identifiant d'image

```
<?php
$image = imagecreatefromjpeg("Images/fond.jpg");
:
?>
```

1) Création du modèle de l'image

- Modes images disponibles
 - Fonction : `gd_info()`
 - Retourne un tableau des informations sur la configuration de gd

```
<?php
$info = gd_info();
foreach ($info as $i=>$v)
{ echo "$i = $v<br>";
}
?>
```

```
GD Version = 2.0 or higher
FreeType Support = 1
FreeType Linkage = with freetype
TLib Support = 1
GIF Read Support = 1
GIF Create Support = 1
JPG Support = 1
PNG Support = 1
WBMP Support = 1
XBM Support =
JIS-mapped Japanese Font Support =
```

2) Travail sur le modèle

- **Manipulation dynamique des couleurs**
 - **Couleurs disponibles en fonction du type d'image**
 - Palettes indexées (256 couleurs)
 - Mode true color 24/32 bits : pas de problème de palette
 - **Demande d'une nouvelle couleur**
 - Récupération d'un identifiant de couleur
 - `imagecolorallocate(image,rouge,vert,bleu);`
 - `imagecolorallocatealpha(image,r,v,b,a);`
- **Remarques**
 - Première couleur correspond à l'arrière plan
 - Palette associée à l'image
 - Nombre de couleurs disponibles réduit

2) Travail sur le modèle

- **Gestion d'une palette de couleurs**
 - **Affectation d'une couleur à la palette**
 - `imagecolorset(image,index,r,v,b);`
 - **Recherche de la couleur de la palette la plus proche de la couleur désirée**
 - `imagecolorclosest(image,r,v,b);`
 - `imagecolorclosestalpha(image,r,v,b,a);`
 - **Copie de la palette d'une image**
 - `imagepalettecopy(destination,source);`
 - **Affectation de la couleur de transparence**
 - Image de format GIF
 - `imagecolortransparent(image,couleur);`

2) Travail sur le modèle

- **Tracé au niveau du pixel**
 - **Couleur d'un pixel de l'image**
 - `imagecolorat(image,x,y);`
 - Renvoie un entier codant #RRVBB
 - **Modification de la couleur d'un pixel**
 - `imagepixel(image,x,y,couleur);`
 - **Récupération des dimensions de l'image**
 - `imagesx(image);`
 - ✓ Retourne la largeur de l'image dans un entier
 - `imagesy(image);`
 - ✓ Retourne la hauteur de l'image dans un entier
 - `getimagesize(fichier);`
 - ✓ Taille d'une image dans un fichier
 - ✓ Retourne un tableau de 2 entier lx,ly

2) Travail sur le modèle

- **Exemple de traitement d'image sous GD**

```
<?php
$image = imagecreatefrompng('img00.png');
$lx = imagesx($image);
$ly = imagesy($image);

for ($ix=0; $ix<$lx; $ix++)
{ for ($iy=0; $iy<$ly/2; $iy++)
  { $couleur = imagecolorat($image,$ix,$iy);
    $rouge = ($couleur >> 16) & 0xFF;
    $vert = ($couleur >> 8) & 0xFF;
    $bleu = $couleur & 0xFF;

    $gris = (int)(0.35*$rouge + 0.5*$vert + 0.15*$bleu);
    $couleur = imagecolorallocate($image,$gris,$gris,$gris);
    imagepixel($image,$ix,$iy+$iy/2,$couleur);
  }
}
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>
```

2) Travail sur le modèle

- **Exemple de traitement d'image sous GD**

Exemple



2) Travail sur le modèle

- **Tracé de formes**
 - **Arc**
 - `imagearc(img,x,y,w,h,angle1,angle2,couleur);`
 - `imagefilledarc(...);`
 - **Ligne / ligne pointillée**
 - `imageline(image,x1,y1,x2,y2,couleur);`
 - `imagedashedline(image,x1,y1,x2,y2,couleur);`
 - **Rectangle**
 - `imagerectangle(image,x1,y1,x2,y2,couleur);`
 - `imagefilledrectangle(...);`
 - **Ellipse**
 - `imageellipse(img,x,y,w,h,couleur);`
 - `imagefilledellipse(...);`

2) Travail sur le modèle

● Exemple de dessin sous GD

```

<?php
$lx = 200;
$ly = 200;
$image = imagecreate (200,200);

$gris = imagecolorallocate($image,200,200,200);
$rouge = imagecolorallocate($image,255,0,0);
$vert = imagecolorallocate($image,0,0xFF,0);

for ($i=0; $i<10; $i++)
{
  $a_x = mt_rand(30,170);  $a_y = mt_rand(30,170);
  $a_tx = mt_rand(0,60);  $a_ty = mt_rand(0,60);

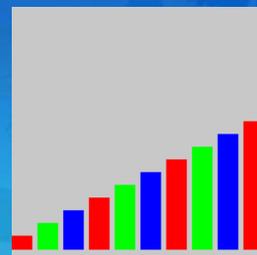
  imagefilledrectangle($image,$a_x,$a_y,$a_tx,$a_ty,0,360,$rouge);
}
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>

```

2) Travail sur le modèle

● Exemple de dessin sous GD

Exemple



2) Travail sur le modèle

● Remplissage de zone

- Remplit une zone fermée de pixels avec la couleur précisée
- `imagefill (image,x,y,couleur);`

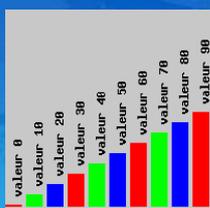
2) Travail sur le modèle

● Fonctions de police de caractères

- Polices prédéfinies 1-5
- Demande d'un identifiant de police
 - `imageloadfont (nom_police);`
- Retourne la hauteur des caractères en pixels
 - `imagefontheight();`
- Affichage de caractères
 - `imagechar (img,x,y,police,caractère,coul);`
 - `imagecharup (...);`
- Affichage de chaînes de caractères
 - `imagestring (img,x,y,police,chaîne,couleur);`
 - `imagestringup (...);`

2) Travail sur le modèle

Exemple



2) Travail sur le modèle

● Manipulation d'images

- Copie simple
 - Une zone de dimensions (w, h),
 - de l'image source (src) en position (src_x,src_y),
 - vers l'image de destination (dst) en position (dst_x,dst_y)
 - `imagecopy (dst,src,dst_x,dst_y,src_x,src_y,w,h);`
- Copie avec mélange
 - Coefficient d'opacité (opac)
 - `imagecopymerge (dst,src,dst_x,dst_y,src_x,src_y,w,h,opac);`

2) Travail sur le modèle

```
<?php
$image = imagecreatefrompng('img00.png');
// dimensions de la mosaïque : dx,dy
$lx = imagesx($image);           $ly = imagesy($image);
$dx=(int)($lx/4);                $dy=(int)($ly/4);
$lx=4*$dx;                       $ly=4*$dy;
$melange = imagecreatetruecolor ($lx,$ly);

$ordre=array(5,7,3,2,4,12,9,10,1,6,11,15,8,14,13,0);
// modification aléatoire
shuffle($ordre);
for ($i=0; $i<16; $i++)
{ $ix0 = ((int)($i%4))*$dx;
  $iy0 = ((int)($i/4))*$dy;
  $ix1 = ((int)($ordre[$i%4]))*$dx;
  $iy1 = ((int)($ordre[$i/4]))*$dy;
  imagecopy($melange, $image, $ix0, $iy0, $ix1, $iy1, $dx-1, $dy-1);
}
header('Content-Type: image/png');
imagepng($melange);
imagedestroy($image);
imagedestroy($melange);
?>
```

2) Travail sur le modèle



2) Travail sur le modèle

● Manipulation d'images

– Copie avec redimensionnement

- `imagecopyresized(dst,src,dst_x,dst_y,src_x,src_y,src_w,src_h,dst_w,dst_h);`

– Meilleure qualité du rééchantillonnage

- `imagecopyresampled(dst,src,dst_x,dst_y,src_x,src_y,src_w,src_h,dst_w,dst_h);`

```
<?php
// Fichier et nouvelle taille
$filename = 'img00.png';
$percent = 0.5;

// Content type
header('Content-type: image/jpeg');

// Calcul des nouvelles dimensions
list($width, $height) = getimagesize($filename);
$newwidth = $width * $percent;
$newheight = $height * $percent;

// chargement
$thumb = imagecreatetruecolor(2*$newwidth, $newheight);
$source = imagecreatefrompng($filename);

// Redimensionnement
imagecopyresized($thumb, $source, 0, 0, 0, 0, $newwidth, $newheight,
$width, $height);
imagecopyresampled($thumb, $source, $newwidth, 0, 0, 0, $newwidth,
$newheight, $width, $height);

// Affichage
imagejpeg($thumb);
?>
```

2) Travail sur le modèle

<code>imagecopyresized(\$thumb, \$source, 0, 0, 0, 0, \$newwidth, \$newheight, \$width, \$height);</code>	<code>imagecopyresampled(\$thumb, \$source, \$newwidth, 0, 0, 0, \$newwidth, \$newheight, \$width, \$height);</code>
---	--



2) Travail sur le modèle

● Rotation d'image

- fait tourner l'image d'un angle exprimé en degrés.
- `bgd_color` spécifie la couleur des zones qui seront découvertes après la rotation.
- `imagerotate(image,angle,bgd_color);`

3) Fabrication de l'image finale

- **Deux solutions**
 - **Affichage de l'image dans le navigateur**
 - Calculée et affichée dans le navigateur à chaque appel de la page.
 - **Sauvegarde d'une image dans un fichier**
 - Exploitation en deux étapes
 - ✓ Génération d'un fichier
 - ✓ Affichage du fichier
- **Remarque : le format GIF**
 - Utilise la compression LZW, breveté
 - Lecture possible
 - Pas de fonction d'écriture disponible

3) Fabrication de l'image finale

- **Entête**
 - Définition du format de l'image
 - **header(« code mime »);**
 - 'Content-Type: image/png'
 - 'Content-Type: image/jpeg'

3) Fabrication de l'image finale

- **Affichage de l'image sur le navigateur**
 - Php renvoie le code binaire de l'image
 - Affichée en pleine page
- **Fonction d'envoi du contenu de l'image**
 - `imagepng(image[, fichier]);`
 - `imagejpg(image[, fichier, qualite]);`
 - Qualité : compression
 - ✓ 1 = forte compression / forte dégradation
 - ✓ 100 = faible compression / pas de dégradation

```
<?php
$image = imagecreatefromgif('test.gif');
header('Content-Type: image/png');
imagepng($image);
?>
```

3) Fabrication de l'image finale

- **Utilisation du cache**
 - Évite de recalculer plusieurs fois la même image
 - Cache = pages pré-calculées

4) Libération de la mémoire

- **Libération de la mémoire**
 - Nécessaire pour supprimer la mémoire occupée par l'image temporaire sur le serveur
 - `imagedestroy(image);`

Images dans les bases de données PHP

- Image non mémorisées directement dans la base
- Image identifiées par leur nom
- Mémorisation sur le site hors base

Création d'un album personnalisé

Outils open source

- Bibliothèque JpGraph (<http://www.php.net/jpgraph>)
- Gallery (<http://gallery.mambo.com>)
 - Gestionnaire d'album photo

Création d'un compteur personnalisé

- Générer une image à partir d'une source
 - Source : ensemble des imagettes
 - Liste des numéros dans l'ordre croissant
 - Avec un fond transparent
- But
 - Extraire les chiffres dans l'image
 - Pouvoir les superposer à un fond

0123456789

Création d'un compteur personnalisé

```

<?php
$nbre = (int)($_GET['nbre']);
$longueur=( $nbre==0)?1:$longueur = (int)Log10($nbre)+1;
$imageSRC = ImageCreateFromPNG('chiffres.png');
$imageDEST = ImageCreateTrueColor($longueur*32,48);

ImageAlphaBlending($imageSRC, false);
ImageAlphaBlending($imageDEST, false);
imagesavealpha($imageDEST, true);

while ($longueur-->0)
{
    $val= $nbre%10;
    $nbre/=10;
    ImageCopy($imageDEST,$imageSRC,$longueur*32,0,$val*32,0,32,48);
}

header('Content-Type: image/png');
imagepng($imageDEST);
ImageDestroy($imageSRC);
ImageDestroy($imageDEST);
?>
    
```

Création d'un compteur personnalisé

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title>Sans Titre</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1"/>
<meta name="generator" content="HAPedit 3.1">
</head>
<body background="img/illustration.jpg">
<hr />
mon compteur : <br> <img src = "compteur.php?nbre=310570">
</body>
</html>
    
```

PHP

Notions d'objet

Benoît Piranda
Équipe SISAR
Université de Marne La Vallée

PHP et programmation objet

- Encapsulation
 - Regroupement des données et des traitements
- Référencement
 - Plusieurs identifiants pour un seul objet
- Héritage
 - Organisation hiérarchique des objets
- Classes abstraites et interfaces

Encapsulation en PHP

- PHP 4
 - Mot clé « class »
 - Méthodes définies comme les fonctions
 - Méthodes particulières
 - ✓ Fonction nom_de_la_classe : constructeur
 - Attribut défini par l'opérateur var

```
class etudiant
{
    var $num;
    var $nom;
    var $date_du_jour;

    function etudiant()
    {
        $this->date_du_jour = date("d/m/Y");
    }
}

$paul = new etudiant;
$paul->nom = "Paul";
echo $paul->date_du_jour;
```

Encapsulation en PHP

- PHP 5
 - Mot clé « class »
 - Méthodes définies comme les fonctions
 - Méthodes particulières
 - ✓ __construct
 - ✓ __destruct
 - Contrôle d'accès aux attributs
 - Public
 - Protected
 - Private
 - Final

Exemple de classe simple

```
<?php
class etudiant
{
    private $num;
    public :
    $nom;

    function __construct($num, $nom)
    {
        $this->num = $num;
        $this->nom = $nom;
    }
    function getNum()
    {
        return $num;
    }
}

$paul = new etudiant(1001,"Paul");
echo $paul->nom, " n° : ", $paul->getNum();
?>
```

Méthodes particulières

- Constructeur / Destructeur
 - Méthode __construct();
 - Méthode __destruct();
 - Appelé par la fonction unset();
- Affichage de l'objet
 - Méthode __toString();
 - Lancée lors du cast automatique par les fonctions et opérateurs d'affichage
 - Méthode __clone();
 - Copie d'un objet : \$clone= clone original;
 - Méthodes __get() et __set()
 - Appelées lorsque l'on accède à un attribut inexistant

Exemple de classe

```
<?
class maClasse
{
    private $valeur= array();

    public function __set($nom,$valeur)
    {
        $this->valeur[$nom]=$valeur;
    }
    public function __get($nom)
    {
        return $this->valeur[$nom];
    }
    public function __toString()
    {
        $res = "<table><tr><td>".implode('</td><td>', $valeur)
        . "</td></tr></table>";
        return $res;
    }
}

$monObjet = new maClasse();
$monObjet->propriete = 'texte'; // attribut inexistant : __set
echo $monObjet->propriete."<br />"; // attribut inexistant : __get
echo $monObjet;
?>
```

Héritage

- Héritage d'une classe
 - Partage les attributs et les méthodes
 - Opérateur : extends
 - Accès à un méthode la classe mère
 - parent::
- Héritage strict
 - Nombre de paramètres Obligatoires de la méthode filles <= Nombre de paramètre de la méthode mère
 - Nombre de paramètres possibles de la méthode filles >= Nombre de paramètre de la méthode mère

Exemple d'héritage

```
<?php
class vehicule
{ public function freine() {...}
  public function avance() {...}
}

Class voiture extends vehicule
{ public function klaxonne() {...}
  public function freine($temps=NULL) {...}
  public function avance($temps=NULL)
  { this->passer_la_vitesse($vitesse);
    parent::avancer();
  }
  public function demarre();
}
?>
```

Source : PHP 5 avancé, Eric Dasnet & Cyril Pierre de Gever (Evrilles)

PHP Objet

- Classes abstraites
 - Mot clé abstract
 - Force les classes dérivées à implémenter certaines méthodes

```
abstract class vehicule
{ abstract function freine();
  abstract function avance();
}
```

- Classes finales
 - Mot clé final
 - Interdit la surcharge de méthode par les classes dérivées

```
final class voiture extends vehicule
{ ...
}
```

PHP Objet

- Les interfaces
 - Permet d'imposer une API
 - Mot clé
 - interface : pour déclarer l'interface
 - implements : pour associer une classe

```
interface PeutAvancer
{ public function arreter();
  public function avancer();
}

interface FaitDeLaLumiere
{ public function allumer();
  public function eteindre();
}

class voiture implements PeutAvancer, FaitDeLaLumiere {
}
```

Utilisation d'objets sous PHP

Création de fichier pdf



Benoît Piranda
Équipe SISAR
Université de Marne La Vallée

La bibliothèque FPDF

- Principales fonctionnalités
 - Choix des unités, du format des pages et des marges;
 - Gestion des en-têtes et pieds de page;
 - Saut de page, saut de ligne et justification
 - Images (JPEG et PNG);
 - Couleurs;
 - Liens;
 - Support des polices TrueType et Type1;
 - Compression des pages.
- fonctionne avec PHP 4 et PHP 5
 - Simple à installer (fichiers au niveau du site)

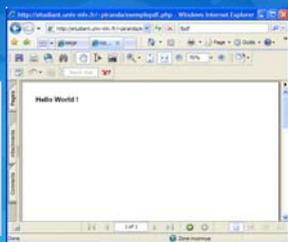
La bibliothèque FPDF

● Exemple simple

- Utilisation d'un objet de classe fpdf
- Créer une nouvelle page
- Choix de la police
- Affichage d'un texte
- Sortie du résultat sur le navigateur

```
<?php
require('fpdf.php');

$pdf=new FPDF();
$pdf->AddPage();
$pdf->SetFont('Arial','B',16);
$pdf->Cell(40,10,'Hello World !');
$pdf->Output();
?>
```



Exemple avancé

● Créer une entête et un pied de page

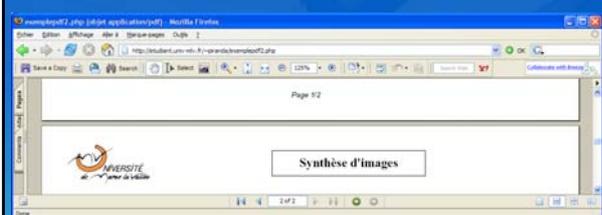
- Principe : créer une classe dérivée de FPDF

```
require('fpdf.php');

class PDF extends FPDF
{
function Header() // surcharge de la méthode En-tête
{
$this->Image('UMLV.png',10,8,33); //Logo
$this->SetFont('Times','B',14); //Police Arial gras 14
$this->Cell(80); //Décalage à droite
$this->Cell(60,10,'Synthèse d\'images',1,0,'C'); //Titre
$this->Ln(20); //Saut de ligne
}

function Footer() //surcharge de la méthode Pied de page
{
$this->SetY(-15); //Positionnement à 1,5 cm du bas
$this->SetFont('Arial','I',8); //Police Arial italique 8
//Numéro de page
$this->Cell(0,10,'Page '.$this->PageNo().'/{nb}',0,0,'C');
}
}
$pdf=new PDF();
$pdf->AliasNbPages(); // associe {nb} au nombre de pages
```

Exemple avancé



Exemple avancé

```
//Tableau simple
function BasicTable($header,$data)
{
//En-tête
foreach($header as $col)
$this->Cell(40,7,$col,1);
$this->Ln();
//Données
foreach($data as $row)
{
foreach($row as $col)
$this->Cell(40,6,$col,1);
}
$this->Ln();// saut de ligne
}
}
```

 A screenshot of a web browser displaying a PDF document. The document contains a table with the following data:

Nom	Prénom	Moyenne
ALGELDEH	Mohamed	13
BAUER	Charlotte	12.75
BUTH	Elise	12
COURTOIS	Nicolas	12.5
CRAMER	Cédric	12.5
DUCHAUSSOY	Teddy	14
EAR	Maxime	12.25
FILLION	Frédéric	11.75
GABRYELEWICZ	Julien	12
LIADRYA	Elzoumana	14
LIMANON	Laurent	16
MELANGE	Yves	13.25
MARLIN	Mikhael	11.25
SHIBUKI	Kenji	13

Exemple avancé

```
//Tableau coloré
function FancyTable($header,$data,$forme)
{
//Couleurs, épaisseur du trait et police grasse
$this->SetFillColor(0,0,255); $this->SetTextColor(255);
$this->SetDrawColor(0,0,128); $this->SetLineWidth(.3);
$this->SetFont('', 'B');
//En-tête
for($i=0;$i<count($forme['largeurs']);$i++)
$this->Cell($forme['largeurs'][$i],7,$header[$i],1);
$this->Ln(); // saut de ligne
//Restauration des couleurs et de la police
$this->SetFillColor(224,235,255);
$this->SetTextColor(0); $this->SetFont('', 'N');
//Données
$fill=0;
foreach($data as $row)
{
for($i=0;$i<count($forme['largeurs']);$i++)
{
$this->Cell($forme['largeurs'][$i],6,$row[$i],1);
}
}
$this->Ln();
$fill=!$fill;
}
$this->Cell(array_sum($forme['largeurs'],0),10,'');
}
```

 A screenshot of a web browser displaying a PDF document. The document contains a color-coded table with the following data:

Nom	Prénom	Moyenne
ALGELDEH	Mohamed	13
BAUER	Charlotte	12.75
BUTH	Elise	12
COURTOIS	Nicolas	12.5
CRAMER	Cédric	12.5
DUCHAUSSOY	Teddy	14
EAR	Maxime	12.25
FILLION	Frédéric	11.75
GABRYELEWICZ	Julien	12
LIADRYA	Elzoumana	14
LIMANON	Laurent	16
MELANGE	Yves	13.25
MARLIN	Mikhael	11.25
SHIBUKI	Kenji	13