Introduction à PHP

Compléments

Updated: 2017/05/29

IUT de Fontainebleau

1. Création d'images dynamiques

2. Manipulation de fichiers

3. Upload de fichiers

4. SQLite (3)

5. Services Web

Création d'images dynamiques

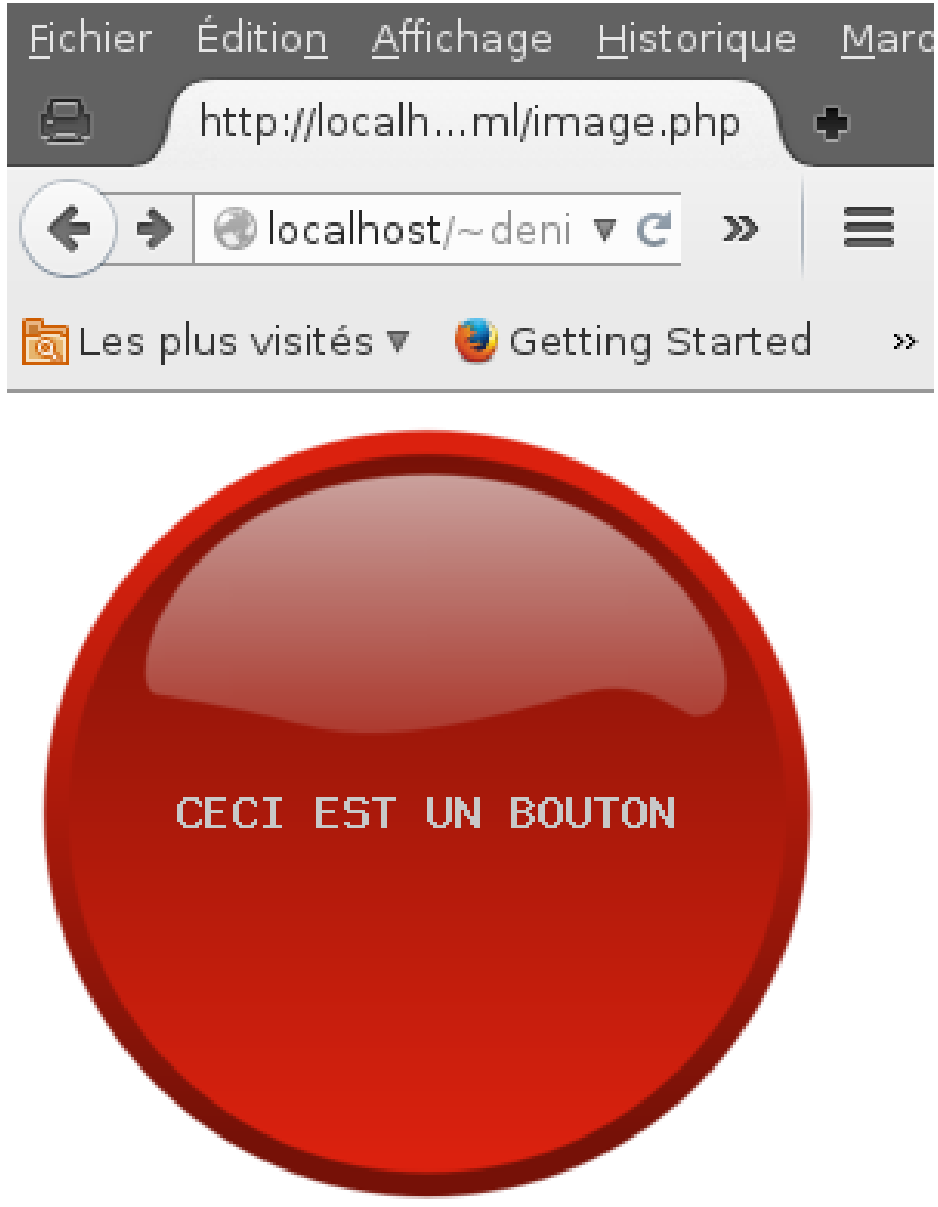
La librairie GD (*gif draw*), disponible à avec php permet de créer, manipuler et générer des images de formats standards : gif, png, jpeg, etc.

Une image est incluse dans une page HTML par

```
<img src="bouton.php?texte=ceci est un bouton">
```

Le script `bouton.php` récupére le contenu de la varibale texte, l'inscrit sur le fond d'une image et la renvoie.

```php
header("Content-type:image/png");
$string=$_GET['texte'];
$im=imagecreatefrompng("images/button.png"); // on crée l'image
imageAlphaBlending($im, false);
imageSaveAlpha($im, true);
$coul=imagecolorallocate($im,220,10,20);
$px=(imagesx($im)-imagefontwidth(3)*strlen($string))/2;
$py=(imagesy($im)-imagefontheight(3))/2;
imagestring($im,3,$px,$py,$string,$coul);// on affiche la chaine
imagepng($im);      // au centre et on envoie
imagedestroy($im); // sur la sortie standard
```


Fichier Édition Affichage Historique Marc
http://localh...ml/image.php
localhost/~deni
Les plus visités
Getting Started
CECI EST UN BOUTON

Créer une image

- vide :

```
resource imagecreate ( int $width , int $height )
resource imagecreatetruecolor ( int $width , int $height )
```

- à partir d'une image existante :

```
resource imagecreatefrompng ( string $filename )
resource imagecreatefromjpeg( string $filename )
resource imagecreatefromgif( string $filename )

/* il existe d'autres formats */
```

Penser à **libérer les ressources sur le serveur** avec `imagedestroy`

Allocation de couleurs

```php
<?php
// création de l'image
//
$image = imagecreatetruecolor(200,200);

// allocation couleurs

$rouge = imagecolorallocate($image,255,0,0);
$noir = imagecolorallocate($image,0,0,0);
$vert = imagecolorallocate($image,0,255,0);

//libération mémoire

imagedestroy($image);
?>
```

Affichage dans le navigateur

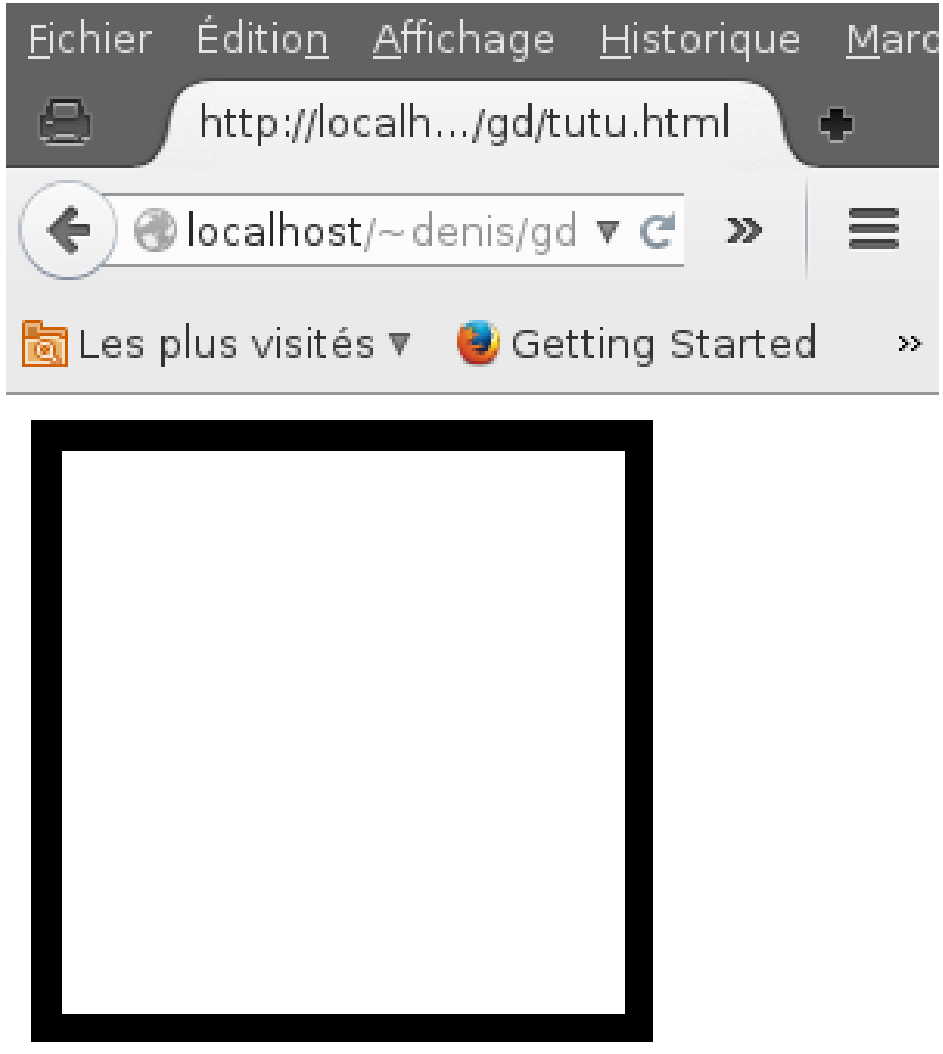
On peut envoyer directement comme résultat d'un script une image

```php
header("Content-Type : image/png");

$image = imagecreatetruecolor(200,200);
$tc  = imagecolorallocate($image, 255, 255, 255);

imagefilledrectangle($image, 10, 10, 190, 190, $tc);
imagepng($image);
imagedestroy($image);
```

```php
<img src="image.php">
```

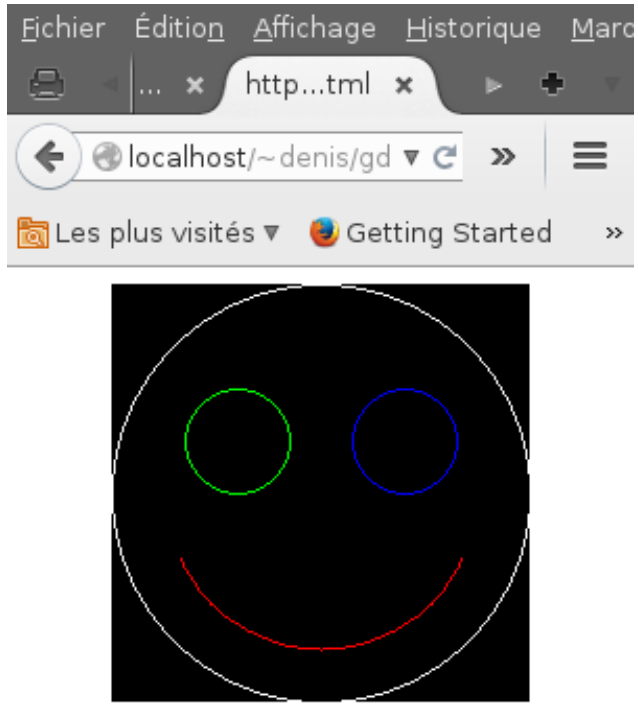
ou la sauvegarder dans un fichier sur le serveur

```php
imagepng($img, "file.png");
```


Fichier Édition Affichage Historique Marc
http://localh.../gd/tutu.html
localhost/~denis/gd
Les plus visités
Getting Started

Tracé de formes

`imageline`	Trace une ligne
`imagedashedline`	idem en pointillé
`imagearc`	crée un arc à partir de son centre, largeur et hauteur
`imagefilledarc`	
`imagerectangle`	crée un rectangle
`imagefilledrectangle`	
`imageellipse`	trace une ellipse (entière)
`imagefilledellipse`	
`imagesetpixel`	allume un pixel avec une couleur




```
imagearc($img, 100, 100, 200, 200,  0, 360, $white);
imagearc($img, 100, 100, 150, 150, 25, 155, $red);
imagearc($img,  60,  75,  50,  50,  0, 360, $green);
imagearc($img, 140,  75,  50,  50,  0, 360, $blue);
```

Tracé de textes

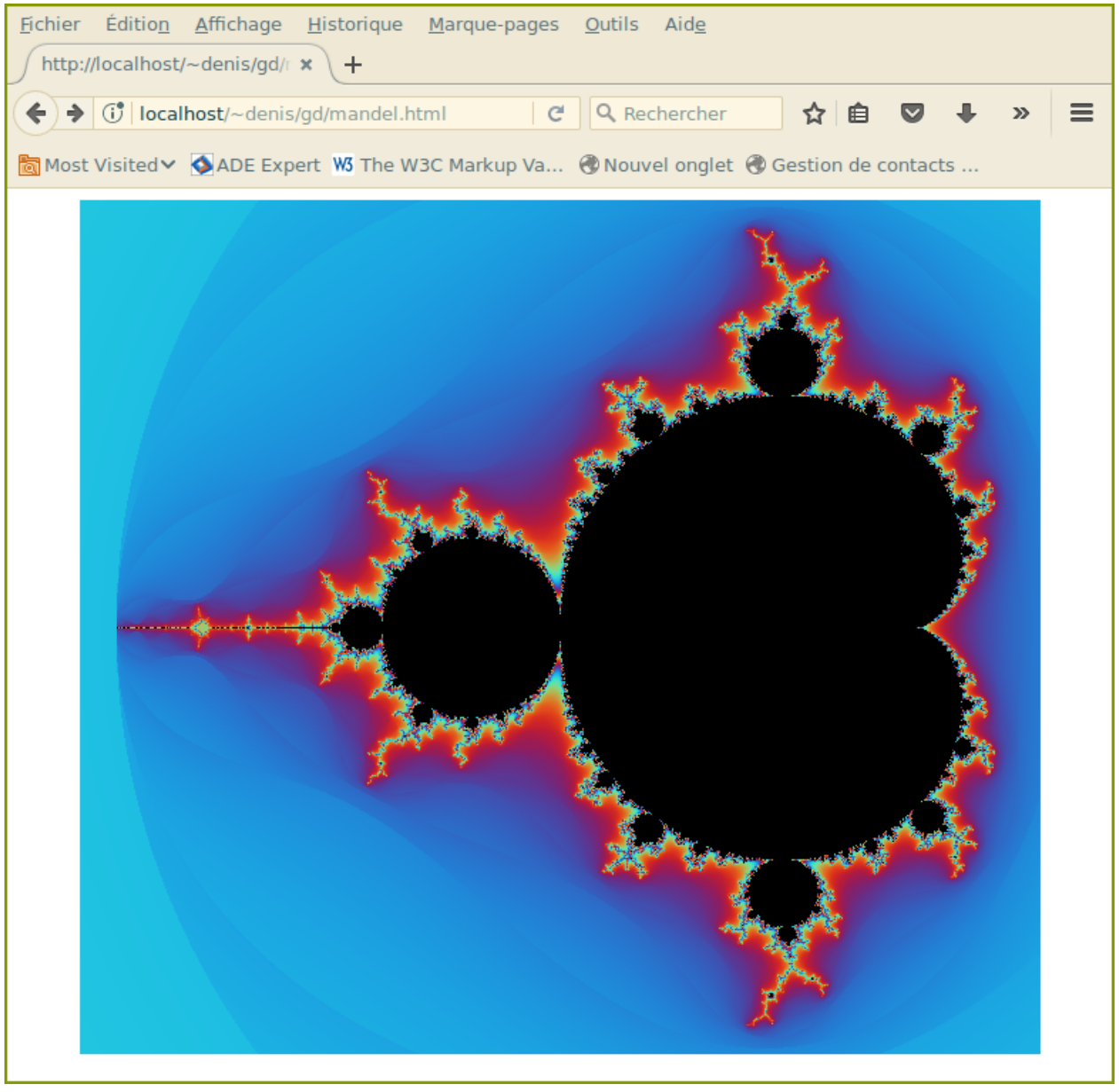
`imagestring`	crée une chaîne de caractères à partir d'un point (x,y). La police peut être une police par défaut (1 à 5), ou une police personnalisé ouverte avec `imageloadfont`
`imagestringup`	crée une chaîne de caractères orientée verticalement
`imageloadfont`	charge une police
`imagefontheight`	retourne la hauteur de la police
`imagefontwidth`	retourne la largeur de la police

Manipulations images

`imagecopy`	permet de copier une portion d'une image sur une autre
`imagecopymerge`	idem, mais en fusionnant les deux selon un coefficient d'opacité
`imagecopyresized`	copie avec redimensionnement
`imagerotate`	rotation de l'image
`getimasize`	determine la taille d'une image stockée dans un fichier.

Gestion des couleurs

`imagecolorallocate`	alloue une couleur
`imagecolorallocatealpha`	alloue une couleur partiellement transparente (si le format le supporte)
`imagecolortransparent`	associe une couleur à une couleur transparente

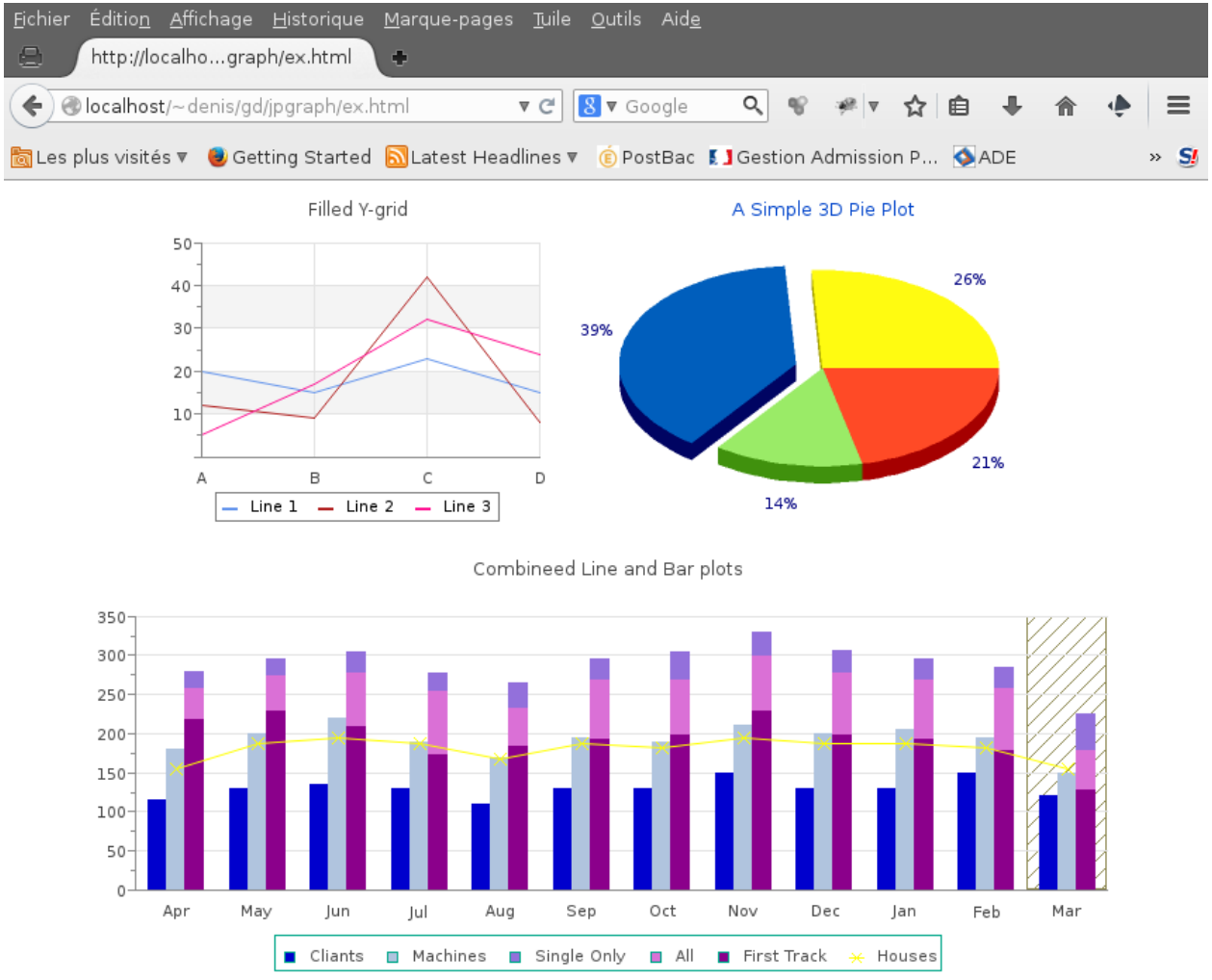

Fichier Édition Affichage Historique Marque-pages Outils Aide
http://localhost/~denis/gd/
localhost/~denis/gd/mandel.html
Rechercher
Most Visited
ADE Expert
The W3C Markup Va...
Nouvel onglet
Gestion de contacts ...

```php
$x1 = -2.1;$x2 = 0.6;$y1 = -1.2;$y2 = 1.2;
$zoom = 250;
$iterations_max = 100;
$image_x = ($x2 - $x1)*$zoom;
$image_y = ($y2 - $y1)*$zoom;
$image = imagecreatetruecolor($image_x, $image_y);
for($x = 0; $x < $image_x; $x++){
  for($y = 0; $y < $image_y; $y++){
    $c_r = $x/$zoom+$x1;$c_i = $y/$zoom+$y1;
    $z_r = 0;$z_i = 0;
    $i   = 0;
    do{
      $tmp = $z_r;
      $z_r = $z_r*$z_r - $z_i*$z_i + $c_r;
      $z_i = 2*$tmp*$z_i + $c_i;
      $i++;
    } while($z_r*$z_r + $z_i*$z_i < 4 AND $i < $iterations_max);
    $continuous_index = $i  - log(log(sqrt($z_r*$z_r+$z_i*$z_i))) / log (2);
    $r = sin(0.3* $continuous_index + 4)*100 +127;
    $v = sin(0.3 * $continuous_index + 2)*100+127;
    $b = sin(0.3 * $continuous_index + 1)*100+127;
    imagesetpixel($image, $x, $y, imagecolorallocate($image,floor($r),floor($v),floor($b)
  }
}
header('Content-type: image/png');
imagepng($image);
```

Pour aller plus loin

Librairie JpGraph

Api objet qui utilise la librairie GD pour produire des graphiques.



```php
require_once ('../jpgraph-3.5.0b1/src/jpgraph.php');
require_once ('../jpgraph-3.5.0b1/src/jpgraph_line.php');
$datay1 = array(20,15,23,15);
// Setup the graph
$graph = new Graph(300,250);
$graph->SetScale("textlin");
$theme_class=new UniversalTheme;
$graph->SetTheme($theme_class);
$graph->img->SetAntiAliasing();
$graph->xgrid->Show();
$graph->xgrid->SetLineStyle("solid");
$graph->xaxis->SetTickLabels(array('A','B','C','D'));
// Create the first line
$p1 = new LinePlot($datay1);
$graph->Add($p1);
$p1->SetColor("#6495ED");
$p1->SetLegend('Line 1');
// Output line
$graph->Stroke();
```

Manipulation de fichiers

Gestion des fichiers

Très proche du C

`fopen`	ouverture d'un fichier (on peut fournir une url http ou ftp)
`fclose`	fermeture
`fread,fwrite`	lecture/écriture binaire
`file_get_contents`	récupère le contenu d'un fichier dans une chaîne
`file`	récupère les lignes d'un fichier dans un tableau
`fgets,fputs`	lecture/écriture de lignes
`fgetss`	lecture de lignes en supprimant les balises
`rewind,fseek,ftell`	positionnement dans un fichier
`fpasstrhu`	lit le fichier jusqu'à sa fin et l'affiche sur la sortie standard
`file_exists`	teste l'existence d'un fichier
`copy,rename,unlink`	copie/renomme/efface un fichier

Exemples

Affichage des lignes (numérotés) d'un fichier

```php
$lines = file('http://www.example.com/');

foreach ($lines as $line_num => $line) {
  echo "Line #<b>{$line_num}</b> : "
    . htmlspecialchars($line) .
    "<br />\n";
}
```

Copie d'un fichier

```php
$fichier="logo_iut.gif";
if (file_exists($fichier))
  copy($fichier,"/tmp/".$fichier);
else
  echo "Fichier $fichier inexistant";
```

Manipulation de répertoires.

`chdir()`	changement de répertoire courant
`opendir()`,`closedir()`	ouverture, fermeture, création ; supression
`mkdir()`,`rmdir()`	création ; supression
`readdir()`	lit l'entrée suivante dans le répertoire
`rewinddir()`	repositionnement au début du répertoire
`dir()`	constructeur objet répertoire

Exemple : commande ls.

```php
chdir("/etc");
$rep=dir(".");
$rep->rewind();
while($f=$rep->read()) echo "$f<br/>";
```

Fichiers CSV

Deux fonctions : `fgetcsv` et `fputcsv`

```php
$row = 1;
if (($handle = fopen("test.csv", "r")) !== FALSE) {
  while (($data = fgetcsv($handle, 1000, ",")) !== FALSE) {
    $num = count($data);
    echo "<p> $num fields in line $row: <br /></p>\n";
    $row++;
    for ($c=0; $c < $num; $c++) {
      echo $data[$c] . "<br />\n";
    }
  }
  fclose($handle);
}
```

Upload de fichiers

```php
$row = 1;
if (($handle = fopen("test.csv", "r")) !== FALSE) {
  while (($data = fgetcsv($handle, 1000, ",")) !== FALSE) {
    $num = count($data);
    echo "<p> $num fields in line $row: <br /></p>\n";
    $row++;
    for ($c=0; $c < $num; $c++) {
      echo $data[$c] . "<br />\n";
    }
  }
  fclose($handle);
}
```

Le tableau _FILE contient un tableau pour chaque fichier transféré (clé correspondant à l'attribut name) contenant :

- `tmp_name` : le nom (et chemin) du fichier temporaire sous lequel le fichier a été stocké.
- `name` : le nom qu'avait le fichier dans l'espace de l'utilisateur
- `size` : la taille du fichier en octets
- `type` : le "MIME type" du fichier (ex : text/html, image/gif, etc.)

- PHP offre la possibilité de recevoir des fichiers texte ou binaire en provenance du client et d'y associer un traitement.
  - réception par la méthode POST ; une boîte de dialogue permet à l'utilisateur de selectionner un fichier local.
- le fichier téléchargé sera stocké temporairement dans le répertoire `$TMPDIR` sur le serveur.
- `move_upload_file()` permet de déplacer un fichier téléchargé par PHP.
- `is_uploaded_file()` permet de vérifier qu'un fichier a bien été téléchargé par la méthode POST.

SQLite (3)

SQLite

- SQLite est une bibliotheque écrite en C proposant un moteur de Base de données et implémentant en grande partie le standard SQL92.
- PHP inclut SQLite depuis sa version 5. Chaque base de données est stockée dans un fichier directement sur le disque où tourne PHP.
- Langage SQL un peu différent.
- `SQLite` est faiblement typé. Comme PHP, ce moteur accepte indifféremment du texte ou des nombres.
- On peut spécifier de manière indicative un type de données pour chaque champ lors de la création de la base.
- La suite concerne SQLite version 3. Il s'agit d'une API objet.

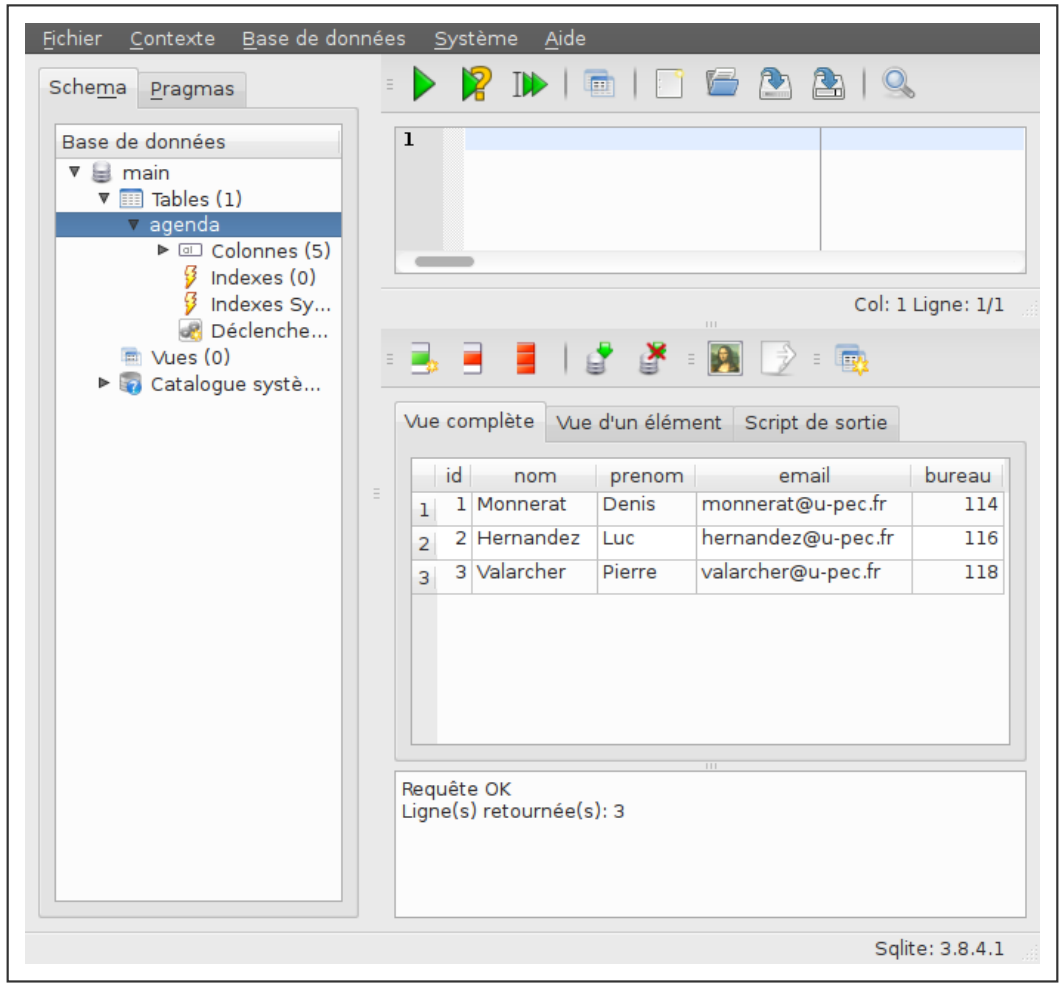
Il n'y a pas de types mais des "classes" de stockage.

```
CREATE TABLE "agenda" (
  "id" INTEGER PRIMARY KEY AUTOINCREMENT,
  "nom" TEXT,
  "prenom" TEXT,
  "email" TEXT,
  "bureau" INTEGER
);
```

NULL	La valeur est NULL
INTEGER	Entier signé signed sur 1, 2, 3, 4, 6, or 8 octets suivant la grandeur du nombre
REAL	Flottant sur 8 octets (IEEE floating point number)
TEXT	texte, stocké en utilisant l'encoage de la base (UTF-8, UTF-16BE ou UTF-16LE)
BLOB	données "binaires

documentation : `http://sqlite.org`

`http://sqliteman.com`, une application (écrite en C) de gestion de bases de données Sqlite3.



Api sqlite3

1. Connexion : ouverture d'un fichier.

```php
<?php
$db = new SQLite3('./agenda.db');
?>
```

2. Requêtes : Deux méthodes : `exec` et `query`.
   - `exec` pour les requêtes "sans résultats" (INSERT, UPDATE, ou DELETE).
   - `query` pour les requêtes de selections (SELECT).

```php
<?php
$query = "INSERT INTO users VALUES ( '$user', '$pass' )";
$db->exec($query) or die("Unable to add user $user");
?>
```

Requête query

```php
<?php
$db = new SQLite3('essai.db');
$results = $db->query('SELECT nom,prenom,email FROM agenda');
while ($row = $results->fetchArray()) {
  echo "<li>".$row['nom']." ".$row['prenom'].
    " : ".$row['email']."</li>";
}
?>
```

Requête préparée

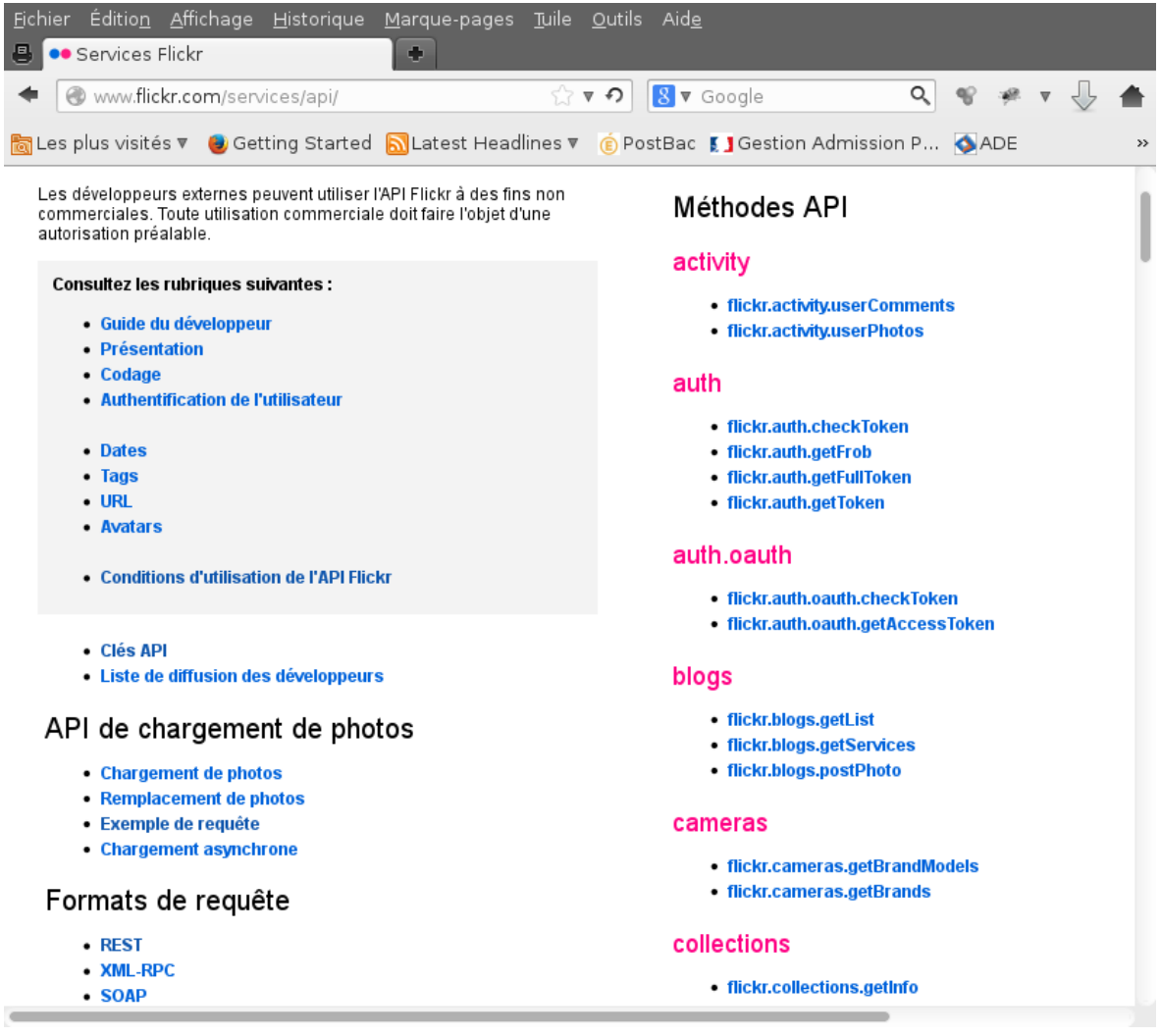
```php
<?php
$db = new SQLite3('mysqlitedb.db');

$db->exec('CREATE TABLE foo (id INTEGER, bar STRING)');
$db->exec("INSERT INTO foo (id, bar)
  VALUES (1, 'Ceci est un test')");

$stmt = $db->prepare('SELECT bar FROM foo WHERE id=:id');
$stmt->bindValue(':id', 1, SQLITE3_INTEGER);

$result = $stmt->execute();
var_dump($result->fetchArray());
?>
```

On peut attaquer une base SQLITE avec PDO (il faut le driver nécessaire)

```php
function initDatabase() {
  $dir = dirname(__FILE__);
  try {
    $db = new PDO('sqlite:' . $dir .'/database.sq3');
  } catch (PDOException $e) {
    die('DB error: ' . $e->getMessage());
  }
  return $db;
}
```

Services Web

De plus en plus de sites mettent à disposition des données, et leurs gestions, des fonctionnalités via une api web (Flickr, Amazon, lastfm, google,...).



Open Data

Ensembles de données numériques dont l'accès, structuré, est libre.

La ville de Rennes a mis les données de transport en accès libre.

`http://data.keolis-rennes.com/`

D'autres acteurs :

- SNCF
- RATP
- OpenStreetMAP
- etc.

Au niveau politique, volonté de structurer et de libérer des données publiques

`data.gouv.fr`

Questions ?

- Qu'est que je peux demander ?
- Comment ?
- Comment exploiter le résultat ?

Définition/Formalisation

- API.
- Accessible via HTTP. (donc avec AJAX depuis JS)
- Exécution distante.
- Exemples
  - Servives XML-RPC ou SOAP (format de données en XML)
  - Services RESTFUL

XML-RPC

- Exécution d'une procédure distante par une requête HTTP.
- Format d'échange des données : XML
- "Ancêtre" de SOAP.

SOAP

- Simple Object Access Protocol.
- Echange de messages XML, le plus souvent avec HTTP.
- Possibilité de description du service au moyen du langage WSDL.

PHP possède des extensions capables de consommer de tels services.

REST

Roy Fielding en 2000

- Representational State Transfer.
- interface uniforme pour accéder à des ressources (n'importe quoi !).
  - Syntaxe universelle pour identifier des ressources : `URI`
  - Un ensemble bien défini d'opérations `GET`, `POST`, `PUT`, `DELETE` (fonctions basiques CRUD)

HTTP	CRUD	SQL
POST	Create	INSERT
GET	read	SELECT
PUT	Update	UPDATE
DELETE	Delete	DELETE

  - Un ensemble de formats différents pour les données : `xml`, `json`, `csv`, `pdf`, ...

REST-ful vs REST-like

- Une application est REST-ful (Totalement REST) si les ressources sont accessibles par des représentions :

```
http://airfrance.com/vols/123
```

  Cette ressource peut faire l'objet d'une requête `POST`, `GET`, `PUT`, ou `DELETE`.

- Certains services utilisent différentes url pour des actions sur la même ressources :

```
http://airfrance.com/vols/getvols?id=123
http://airfrance.com/vols/deletevols?id=123
```

- REST s'est imposait par rapport à SOAP.
- Beaucoup d'api de données, avec le format JSON.

- Je crée un utilisateur :

```
POST /user
prenom=John&nom=Doe&age=25
```

- Je récupére l'utilisateur

```
GET /user/123
```

- Pour changer

```
PUT /user/123
prenom=Johnny
```

Réponse du serveur

```
201 Created
Location: /user/123
```

Réponse du serveur

```
200 OK
<prenom>John</prenom>
<nom>Doe</nom>
<age>25</age>
```

Utilisation d'une api depuis PHP avec cURL

La bibliothèque libcurl permet (entre autres) de faire des requêtes http(s) depuis PHP. Version en ligne de commande => curl.

```
curl https://randomuser.me/api
```

Cette url renvoie, au format JSON, un "utilisateur" aléatoire.


```
{
  "results": [
    {
      "gender": "female",
      "name": {
        "title": "mademoiselle",
        "first": "alice",
        "last": "mathieu"
      },
      "location": {
        "street": "2511 rue de l'abbé-de-l'épée",
        "city": "etagnières",
        "state": "basel-stadt",
```

En PHP

```php
<?php

$curl = curl_init();

$opts = [
    CURLOPT_URL => 'https://randomuser.me/api/',
    CURLOPT_RETURNTRANSFER => true,
];

curl_setopt_array($curl, $opts);

$response = json_decode(curl_exec($curl), true);

print_r($response);
```

En PHP

API OpenWeatherMap

```php
$curl = curl_init();
$opts = [
CURLOPT_URL => 'http://api.openweathermap.org/data/2.5/weather/?'
.'q=Fontainebleau, FR'
.'&APPID=824124b860cfccde7a00b390636217a2'
.'&units=metrics'
.'&lang=fr',
CURLOPT_RETURNTRANSFER => true,
];
curl_setopt_array($curl, $opts);
$response = json_decode(curl_exec($curl), true);
print_r($response);
```

```
Array
(
    [coord] => Array
        (
            [lon] => 2.7
            [lat] => 48.4
        )

    [weather] => Array
        (
            [0] => Array
                (
                    [id] => 211
                    [main] => Thunderstorm
                    [description] => orages
                    [icon] => 11d
                )

        )

    [base] => stations
```

Ecriture d'api en PHP

But : fournir un service par http(s).

- API de données : permettre l'accès à des données depuis http.
  - souplesse
  - abstraction par rapport au sgbd
  - réutilisation
  - http est ton ami (?)
- API de calcul (géolocalisation, etc.)

Besoins techniques :

- Routage
- Filtrage des données
- entrées sorties HTTP
- Mise en cache
- etc.

Quelques micro-frameworks PHP

- `https://www.slimframework.com/`
- `http://flightphp.com/`

Certains framework sont capables de générer une api de données automatiquement.