



## W12 – PHP –Web Programming III

Les Paramètres dans la requête

# debug

- print\_r()
- var\_dump()
- var\_export()

```
<?php
$students;
for($i = 0; $i < 10 ; $i++){
    $students[] = $i;
}
print_r($students);
echo "<br/>";
var_dump($students);
echo "<br/>";
var_export($students);
echo "<br/>";

$adresse2 = [
    "nb" => 255,
    "rd" => "Boul Cremazie Est",
    "city" => "Montréal",
    "zipcode" => "H2E-1M2"
];
print_r($adresse2);
echo "<br/>";
var_dump($adresse2);
echo "<br/>";
var_export($adresse2);
```

# SuperGlobals

- `$GLOBALS`

An associative array containing references to all global variables currently defined in the context

- `$_SERVER`

An array containing information such as headers, paths, and script locations.

- `$_GET` (accessing get params)

An associative array of variables passed to the current script via the URL parameters

- `$_POST` (accessing post params)

An associative array of variables passed to the current script via the HTTP POST method

- `$_FILES` (accessing uploaded files)

An associative array of items uploaded to the current script via the HTTP POST method

- `$_COOKIE` (cookie management)

An associative array of items uploaded to the current script via the HTTP POST method.

- `$_SESSION` (session management)

An associative array containing session variables available to the current script.

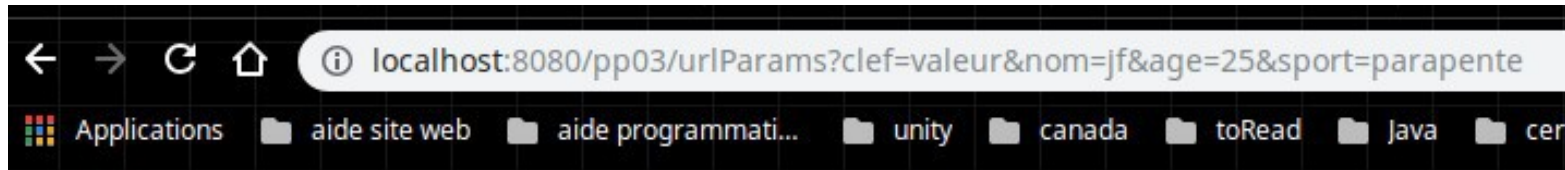
- `$_REQUEST`

An associative array that by default contains the contents of `$_GET`, `$_POST` and `$_COOKIE`.

- `$_ENV`

An associative array of variables passed to the current script via the environment method.

# Parameters in the \$\_GET URL



Array ( [clef] => valeur [nom] => jf [age] => 25 [sport] => parapente )

- Always accessed via \$\_GET superglobal.
- would return :
  - \$\_GET['clef']
  - \$\_GET['nom']
  - \$\_GET['age']
  - \$\_GET['sport']
- All returned values are Strings
- 2 solutions:
  - casting : \$age = (int)\$\_GET['age']      return 0 or the value
  - XXXval : ex int :      \$age = intval(\$\_GET['age']);

# Parameters from forms via \$\_GET & \$\_POST

```
<form action="form.php" method="post">
  <label>LastName
    <input type="text" name="lastName"/>
  </label>
  <label>FirstName
    <input type="text" name="firstName"/>
  </label>
  <label>age
    <input type="text" name="age"/>
  </label>
  <label>Telephone
    <input type="tel" name="tel"/>
  </label>
  <label>Password
    <input type="password" name="pwd"/>
  </label>
  <label>Password repeat
    <input type="password" name="pwdR"/>
  </label>
  <input type="submit" value="OK"/>
</form>
```

- action : attribute that specifies the target for form processing
- method : specifies whether GET or POST request will be used
- name : attribute on each input or select/option indicating the parameter name that will be in the \$\_GET or \$\_POST

# Parameter validation

- Never trust user input !!!
- Test if parameter exists. What is happening??
- Methods exists to check the presence of a parameter in superglobals.

For example the `isset()` method

```
if (isset($_GET['name'])) {  
    $name = $_GET['name'];  
    echo 'Hello ' . htmlspecialchars($name) . ' welcome  
to my site: '  
} else  
    echo 'parameter not valid';
```

# htmlspecialchars()

- Never trust user input !!!
- If HTML tags or JavaScript (etc.) is written in form directly, that code could be interpreted by PHP!!!
- This can be avoided with the `htmlspecialchars()` function

```
if (isset($_GET['name'])) {  
    $name = $_GET['name'];  
    echo 'Hello ' . htmlspecialchars($name) . '  
welcome to my site: '  
} else  
    echo 'parameter not valid';
```