



DICJ

Département d'informatique

Introduction au PHP partie 2

420-JQA-JQ

Faire des constantes

- Le nom commence par une lettre ou un “_” ;
- On ne met pas de “\$” à une constante ;
- Toujours en majuscules ;
- Automatiquement globale et utilisable partout dans le script ;
- Ce déclare comme suit:
 - Define(nomDeLaVariable, valeur, insensibleALaCasse)
 - Le troisième paramètre est optionnel, sa valeur par défaut est “false” (donc sensible à la casse)

Exemples de constantes

```
define("HELLOWORLD", "Hello World!");  
define("SALUTATIONS", "Bonjour!", false);  
define("MESSAGE", "Message à l'utilisateur", true);
```

Les opérateurs

- Arithmétique:
 - +, -, *, /, %, **
- Assignment:
 - =, +=, -=, *=, /=, %=
- Comparaison:
 - ==, ===, !=, <>, !==, <, >, <=, >=
- Logique:
 - And, or, xor, &&, ||, !
- Incrémentation:
 - ++, --
- Concaténation:
 - ., .=

Les structures conditionnelles

- Comme dans tous les autres langages, soit :
 - If...Else
 - If...Elseif...Else
 - Switch

```
if(condition) {  
    //du code  
}  
elseif(condtion) {  
  
}  
else{  
    //du code  
}
```

```
if(condition) {  
    //du code  
}  
else{  
    //du code  
}
```

```
switch (valeurOuVariableOuFonction) {  
    case valeur1: //du code  
        break;  
    case valeur2: //du code  
        break;  
    default:      //du code  
}
```

GET vs POST

- Le **GET** passe les informations dans l'URL alors que le **POST** les passe dans le corps de la requête HTTP;
- Le **POST** nécessite un ajout à l'en-tête pour indiquer la taille du contenu du corps;
- Le corps d'une requête HTTP est chiffré.

```
GET /foo?par1=val1&par2=val2 HTTP/1.1
User-Agent: Mozilla/5.0
Host: Host
Accept: */*
```

```
POST /foo HTTP/1.1
User-Agent: Mozilla/5.0
Host: Host
Accept: */*
Content-Length: 19

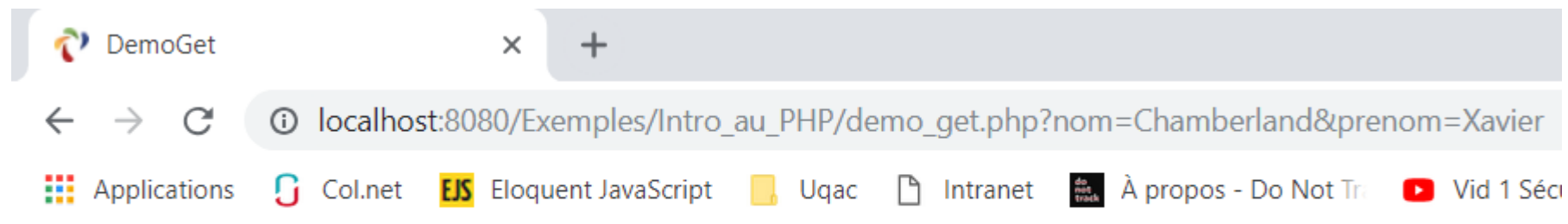
par1=val1&par2=val2
```

POST et GET j'en fais quoi?

- Pour accéder aux informations transmises via le **GET** ou le **POST**, il faut utiliser deux des constantes magiques du PHP:
 - `$_GET[]`
 - `$_POST[]`
- Les deux s'utilisent de la même façon, soit comme un tableau associatif:
 - `$_GET["nomDuParametre"]`
 - `$_POST["nomDuParametre"]`

Le nom du paramètre est le nom donné au noeud HTML dans le formulaire

POST et GET j'en fais quoi? (Exemple)



Bonjour Xavier Chamberland

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>DemoGet</title>
  </head>
  <body>
    <?php
      echo "Bonjour ". $_GET["prenom"]. " ". $_GET["nom"];
    ?>
  </body>
</html>
```


POST et GET j'en fais quoi? (Suite)

- Avant d'utiliser une valeur de **\$_GET** ou de **\$_POST**, il est de bonne pratique de vérifier si la valeur existe.
- Utiliser la fonction **"isset()"**:
 - Retourne **true** s'il y a une valeur, **false** sinon.
- Exemple:

```
if(isset($_GET["mail"]))
{
    echo $_GET["mail"];
}
else
{
    echo "Le courriel n'a pas été reçu";
}
```

← → ↻ ⓘ localhost:8080/Exemples/Suite_Intro_PHP/premierGet.php

Le courriel n'a pas été reçu

← → ↻ ⓘ localhost:8080/Exemples/Suite_Intro_PHP/premierGet.php?mail=xavierchamberland%40cegepjonquiere.ca

xavierchamberland@cegepjonquiere.ca

Informations supplémentaires

- Fonctions en PHP
- Portée des variables
- Constantes magiques