

## CODE MANAGER DB-B

### 1. getCountriesByContinent(\$continent)

```
function getCountriesByContinent($continent)
{
    // pour utiliser la variable globale dans la fonction
    global $pdo;
    $query = 'SELECT * FROM Country WHERE Continent = :cont;';
    $prep = $pdo->prepare($query);
    // on associe ici (bind) le paramètre (:cont) de la req SQL,
    // avec la valeur reçue en paramètre de la fonction ($continent)
    // on prend soin de spécifier le type de la valeur (String) afin
    // de se prémunir d'injections SQL (des filtres seront appliqués)
    $prep->bindValue(':cont', $continent, PDO::PARAM_STR);
    $prep->execute();
    // var_dump($prep); pour du debug
    // var_dump($continent);

    // on retourne un tableau d'objets (car spécifié dans connect-db.php)
    return $prep->fetchAll();
}
```

**Objectif :** Cette fonction récupère la liste de tous les pays d'un continent donné.

**Détails :**

- Elle prend un paramètre \$continent et utilise une requête SQL pour sélectionner tous les pays ayant ce continent spécifique.
- Elle utilise une requête préparée avec le paramètre :cont pour éviter les injections SQL.
- Elle retourne un tableau d'objets représentant les pays du continent spécifié.

### 2. getAllCountries()

```
function getAllCountries()
{
    global $pdo;
    $query = 'SELECT * FROM Country;';
    return $pdo->query($query)->fetchAll();
}
```

**Objectif :** Cette fonction récupère la liste de tous les pays.

**Détails :**

- Elle exécute une requête SQL pour sélectionner tous les pays dans la table Country et retourne un tableau d'objets représentant tous les pays.

### 3. getContinent()

```
function getContinent()
{
    global $pdo;
    $query = 'SELECT DISTINCT continent FROM Country;';
    return $pdo->query($query)->fetchAll();
}
```

**Objectif :** Cette fonction récupère la liste des continents distincts disponibles dans la base de données.

**Détails :**

- Elle exécute une requête SQL pour obtenir une liste de tous les continents distincts à partir de la table Country et retourne cette liste.

### 4. getLangue()

```
function getLangue()
{
    global $pdo;
    $query = 'SELECT l.Name AS Language, COUNT(DISTINCT cl.idCountry) AS NombreDePays FROM `CountryLanguage`';
    return $pdo->query($query)->fetchAll();
}
```

**Objectif :** Cette fonction récupère la liste des langues parlées dans les pays, ainsi que le nombre de pays où chaque langue est parlée.

**Détails :**

- Elle exécute une requête SQL qui joint les tables CountryLanguage et Language pour obtenir le nom de la langue et le nombre de pays où elle est parlée.
- La requête est groupée par langue et ordonnée par le nombre de pays qui parlent chaque langue (du plus grand au plus petit).

### 5. getinformationpays(\$id)

```

function getinformationpays($id)
{
    global $pdo;
    $query = 'SELECT * FROM Country WHERE id = :id;';
    $prep = $pdo->prepare($query);
    $prep->bindValue(':id', $id, PDO::PARAM_INT);
    $prep->execute();
    // var_dump($prep); pour du debug
    // var_dump($continent);

    // on retourne un tableau d'objets (car spécifié dans connect-db.php)
    return $prep->fetch();
}

```

**Objectif :** Cette fonction récupère les informations d'un pays particulier en fonction de son id.

**Détails :**

- Elle prend un identifiant de pays (\$id), exécute une requête préparée pour sélectionner les informations de ce pays dans la table Country.
- Elle retourne un seul objet représentant le pays.

## 6. getCapitale(\$idpays)

```

function getCapitale($idpays)
{
    global $pdo;
    $query = 'SELECT Name FROM City WHERE id =:id;';
    $prep = $pdo->prepare($query);
    $prep->bindValue(':id', $idpays, PDO::PARAM_STR);
    $prep->execute();
    // var_dump($prep); pour du debug
    // var_dump($continent);

    // on retourne un tableau d'objets (car spécifié dans connect-db.php)
    return $prep->fetch();
}

```

**Objectif :** Cette fonction récupère le nom de la capitale d'un pays donné en fonction de son identifiant (\$idpays).

**Détails :**

- Elle utilise une requête préparée pour rechercher le nom de la ville associée à un pays dans la table City.

## 7. getpayslangue(\$country)

```
function getpayslangue($country){
    global $pdo;
    $query = "SELECT cl.idCountry, c.Name AS Country, c.Continent, c.Population, c.Region FROM `CountryLanguage` cl JOIN `Country` c ON cl.idCountry = c.idCountry WHERE cl.Language = '$country'";
    //$prep = $pdo->prepare($query);
    //$prep->bindValue(':id', $idpays, PDO::PARAM_STR);
    //$prep->execute();
    // var_dump($prep); pour du debug
    // var_dump($continent);
    echo $query;

    // on retourne un tableau d'objets (car spécifié dans connect-db.php)
    //return $prep->fetch();
    return $pdo->query($query)->fetchAll();
}
```

**Objectif :** Cette fonction récupère tous les pays où une certaine langue est parlée.

**Détails :**

- Elle prend en paramètre le nom d'une langue et exécute une requête SQL qui joint les tables CountryLanguage, Language, et Country pour obtenir les informations des pays où cette langue est parlée.
- La requête retourne un tableau d'objets avec les détails des pays.

## 8. getPaysByName(\$name)

```
}
function getPaysByName($name)
{
    global $pdo;
    $query = "SELECT * FROM Country WHERE Name = '$name'";
    return $pdo->query($query)->fetch();
}
```

**Objectif :** Cette fonction récupère les informations d'un pays donné par son nom.

**Détails :**

- Elle exécute une requête SQL qui sélectionne un pays avec le nom donné et retourne un seul objet représentant ce pays.

**Conclusion ce code contient des fonctions pour accéder aux informations géographiques d'une base de données SQL, incluant des pays, des continents, des langues et des capitales. Ces fonctions permettent d'extraire des données spécifiques via des requêtes SQL .**