

## 1. La structure de données

Structure de la table partenaire

```
create table partenaire (  
  id    int      auto_increment primary key,  
  nom   varchar(50) not null unique,  
  url   varchar(150) not null unique,  
  fichier varchar(70) not null  
);
```

Contenu

id	nom	url	fichier
1	Courses 80	<a href="https://courses80.fr/index.php">https://courses80.fr/index.php</a>	courses80.png
2	Comité départemental d'athlétisme de la Somme	<a href="http://cda80.athle.com/">http://cda80.athle.com/</a>	cd80.png
3	Info Running en Picardie	<a href="http://info-running-pic.com/">http://info-running-pic.com/</a>	running.png
4	Klikego	<a href="https://www.klikego.com/">https://www.klikego.com/</a>	logo_klikego.png
5	Amiens se prend au jeu	<a href="https://www.amiens.fr/Grands-projets/Amiens-se-pre...">https://www.amiens.fr/Grands-projets/Amiens-se-pre...</a>	amiensseprendaujeu2024.png
6	Conseil départemental de la Somme	<a href="https://www.somme.fr/">https://www.somme.fr/</a>	somme.png
7	Région Haut de France	<a href="https://www.hautsdefrance.fr/">https://www.hautsdefrance.fr/</a>	hautdefrance.png
8	Agence française de lutte contre le dopage	<a href="https://www.afld.fr/">https://www.afld.fr/</a>	afld.png
9	Amiens métropole	<a href="https://www.amiens.fr">https://www.amiens.fr</a>	amiens.png

Déclencheur

```
create trigger avantAjoutPartenaire before insert on partenaire  
for each row  
begin  
  declare msg varchar(255);  
  # contrôle de la colonne id  
  if new.id not regexp '^[0-9]+$' or exists (select 1 from partenaire where id = new.id) then  
    signal sqlstate '45000' set message_text = '#L"identifiant n"est pas valide';  
  end if;  
  
  # contrôle de la colonne nom  
  if new.nom is null then  
    signal sqlstate '45000' set message_text = '#Le nom doit être renseigné';  
  end if;  
  # le nom doit comporter entre 3 et 50 caractères  
  if char_length(new.nom) not between 3 and 50 then  
    signal sqlstate '45000' set message_text = '#Le nom doit comporter entre 3 et 50 caractères';  
  end if;  
  # le nom doit respecter l'expression régulière suivante  
  if new.nom not regexp '^[0-9A-Za-zÀÁÂÃÄÅÇÈÉÊËÌÍÎÏÐÓÔÕÖÙÚÛÜÝàáâãäåçèéêëìíîïðóôõöùúûüýÿ](?[:/," \-]?[0-9A-Za-zÀÁÂÃÄÅÇÈÉÊËÌÍÎÏÐÓÔÕÖÙÚÛÜÝàáâãäåçèéêëìíîïðóôõöùúûüýÿ])*?$' then  
    set msg = concat('#Le nom comporte des caractères non autorisés : ', new.nom);  
    signal sqlstate '45000' set message_text = msg;  
  end if;
```

```
-- On vérifie si le partenaire existe déjà
if exists(select 1 from partenaire where nom = new.nom) then
    signal sqlstate '45000' set message_text = '#Ce partenaire est déjà référencé.';
end if;

# contrôle sur la colonne url
if new.url is null then
    signal sqlstate '45000' set message_text = '#L\'url du partenaire doit être renseignée.';
end if;
if new.url not regexp '^http:\\/\\/|https:\\/\\/(www.)?([a-zA-Z0-9-]){2,}\\.{1,4}([a-zA-Z]){2,6}\\/([a-zA-Z-\\/_\\.0-9#?:=&;,]*)??' then
    signal sqlstate '45000' set message_text = '#L\'URL du partenaire n\'est pas valide.';
end if;

if exists(select 1 from partenaire where url = new.url) then
    signal sqlstate '45000' set message_text = '#Ce partenaire est déjà référencé.';
end if;

# contrôle de la colonne fichier
if new.fichier is null then
    signal sqlstate '45000' set message_text = '#L\'URI du fichier image doit être renseigné.';
end if;
# contrôle du format du fichier image : des lettres, des chiffres, des espaces et se terminer par
une extension jpg, png, webp ou avif
if new.fichier not regexp '^([a-z0-9 -]+\\.\\.(jpg|png|webp|avif))$' then
    signal sqlstate '45000' set message_text = '#L\'URI du fichier image n\'est pas valide.';
end if;
if exists(select 1 from partenaire where fichier = new.fichier) then
    signal sqlstate '45000' set message_text = '#Cette image est déjà référencée.';
end if;
end
```

```
create trigger avantModificationPartenaire before update on partenaire
for each row
begin
    declare msg varchar(255);
    # L'identifiant n'est pas modifiable
    if new.id is null or new.id != old.id then
        signal sqlstate '45000' set message_text = '#L\'identifiant ne peut être modifié';
    end if;
    # contrôle de la colonne nom
    if new.nom is null then
        signal sqlstate '45000' set message_text = '#Le nom doit être renseigné';
    end if;
    # le nom doit comporter entre 3 et 50 caractères
    if char_length(new.nom) not between 3 and 50 then
        signal sqlstate '45000' set message_text = '#Le nom doit comporter entre 3 et 50 caractères';
    end if;
    # le nom doit respecter l'expression régulière suivante
    if new.nom not regexp '^[0-9A-Za-zÀÁÂÃÄÅÇÈÉÊËÌÍÎÏÐÓÔÕÖÙÚÛÜÝàáâãäåçèéêëìíîïðóôõöùúûüýÿ](?:[/, "\-]?[0-9A-Za-zÀÁÂÃÄÅÇÈÉÊËÌÍÎÏÐÓÔÕÖÙÚÛÜÝàáâãäåçèéêëìíîïðóôõöùúûüýÿ])*?$' then
        set msg = concat('#Le nom comporte des caractères non autorisés : ', new.nom);
        signal sqlstate '45000' set message_text = msg;
    end if;
    if new.nom != old.nom then
        if exists (select 1 from partenaire where nom = new.nom) then
            signal sqlstate '45000' set message_text = '#Un autre partenaire possède déjà ce nom ';
        end if;
    end if;
    # contrôle sur la colonne url
    if new.url is null then
        signal sqlstate '45000' set message_text = '#L\'URL du partenaire doit être renseignée';
    end if;
    if new.url not regexp '^http:\\/\\/|https:\\/\\/(www.)?([a-zA-Z0-9-]){2,}\\.[a-zA-Z0-9-]{1,4}([a-zA-Z0-9-]){2,6}(/([a-zA-Z-_\\/.0-9#?:=&;,]*)?)?' then
        signal sqlstate '45000' set message_text = '#L\'URL du partenaire n\'est pas valide';
    end if;
    if new.url != old.url then
        if exists (select 1 from partenaire where url = new.url) then
            signal sqlstate '45000' set message_text = '#Un autre partenaire possède déjà cette URL ';
        end if;
    end if;
    # contrôle de la colonne fichier
    if new.fichier is null or new.fichier != old.fichier then
        signal sqlstate '45000' set message_text = '#L\'URI du fichier image n\'est pas modifiable';
    end if;
end
```

## 2. La classe métier partenaire

```
class Partenaire extends Table {
    const DIR = RACINE . "/data/partenaire/";

    public function __construct() {
        parent::__construct('partenaire');

        // le nom du partenaire doit être renseigné entre 3 et 50 caractères
        // Lettres, lettres accentuées, chiffres séparés éventuellement par un espace, une apostrophe ou un tiret
        $input = new InputText();
        $input->Pattern = "^[0-9A-Za-zÀÁÂÃÄÅÇÈÉÊËÌÍÎÏÐÓÔÕÖÙÚÛÜÝàáâãäåçèéêëìíîïðóôõöùúûüýÿ](?![\ \-]?[0-9A-Za-zÀÁÂÃÄÅÇÈÉÊËÌÍÎÏÐÓÔÕÖÙÚÛÜÝàáâãäåçèéêëìíîïðóôõöùúûüýÿ])*";
        $input->MinLength = 3;
        $input->MaxLength = 50;
        $this->columns['nom'] = $input;

        // url du partenaire
        $input = new InputUrl();
        $input->Require = true;
        $input->VerifierExistence = true;
        $this->columns['url'] = $input;

        // fichier : nom du fichier image représentant le logo du partenaire
        // la valeur est alimentée par celle de l'objet InputFile
        $lesParametres = require RACINE . '/.config/partenaire.php';
        $input = new InputFileImg($lesParametres);
        $input->Directory = self::DIR;
        $this->columns['fichier'] = $input;

        // Définition des colonnes pouvant être modifiée unitairement
        $this->listOfColumns->Values = ['nom', 'url', 'fichier'];
    }

    public static function getAll() {
        $sql = "Select id, nom, url, fichier from partenaire;";
        $select = new Select();
        $lesLignes = $select->getRows($sql);
        // vérification de l'existence de l'image et solution au problème de non rafraichissement du
        cache
        $nb = count($lesLignes);
        for ($i = 0; $i < $nb; $i++) {
            $nomFichier = $lesLignes[$i]['fichier'];
            if (file_exists(self::DIR . $nomFichier)) {
                $lesLignes[$i]['present'] = 1;
                $lesLignes[$i]['src'] = "$nomFichier?v=" . filemtime(self::DIR . $nomFichier);
            } else {
                $lesLignes[$i]['present'] = 0;
            }
        }
        return $lesLignes;
    }
}
```

```
/**
 * Retourne les informations sur le partenaire dont l'id est passé en paramètre
 *
 * @param int $id L'id du partenaire
 * @return array | null
 */
public static function getById($id) {
    // récupération des informations sur l'étudiant
    $sql = <<<EOD
        SELECT id, nom, url, fichier
        FROM partenaire
        Where id = :id
    EOD;
    $select = new Select();
    return $select->getRow($sql, ['id' => $id]);
}

// mise à jour du champ fichier
public static function modifierLogo(int $id, $nomFichier): void {
    $sql = <<<EOD
        update partenaire
        set fichier = :nomFichier
        where id = :id
    EOD;
    $db = Database::getInstance();
    $curseur = $db->prepare($sql);
    $curseur->bindValue('id', $id);
    $curseur->bindValue('nomFichier', $nomFichier);
    $curseur->execute();
}
}
```

### 3. Les logos

Ils sont stockés dans le répertoire data/partenaire

Le fichier de configuration est stocké dans le répertoire .config : partenaire.php

```
return [
    'repertoire' => '/data/partenaire/',
    'extensions' => ["jpg", "png", "webp", "avif"],
    'types' => ["image/jpeg", "image/jpg", "image/png", "image/webp", "image/avif"],
    'maxSize' => 150 * 1024,
    'require' => true,
    'rename' => false,
    'sansAccent' => true,
    'redimensionner' => false,
    'height' => 150,
    'width' => 300,
    'accept' => '.jpg, .png, .webp, .avif'
];
```

## 4. L'affichage des partenaires sur la page d'accueil

### 4.1. Modification sur le fichier index.php

```
// les logos des partenaires
$lesPartenaires = json_encode(Partenaire::getAll());

// transmission des données à l'interface
$head = <<<EOD
    <script>
        let lesPartenaires = $lesPartenaires;
    </script>
EOD;
```

### 4.2. Modification sur le fichier index.js

```
// affichage des liens vers les partenaires
for (const element of lesPartenaires) {
    if (element.present === 1) {
        let a = document.createElement('a');
        a.href = element.url;
        a.target = "lien";
        let img = document.createElement('img');
        img.src = "/data/partenaire/" + element.src;
        img.classList.add('logo');
        img.title = element.nom;
        img.style.maxHeight = '80px';
        a.appendChild(img);
        detailPartenaire.appendChild(a);
    }
}
```

## 5. L'administration des partenaires

### 5.1. Le fichier administration/index.php

```
// Contrôle de l'accès
Administrateur::controlerAcces();

// génération d'un token pour garantir l'origine des appels vers les autres scripts du module
$token = Jeton::creer();

// chargement des données
$lesPartenaires = json_encode(Partenaire::getAll());
$lesParametres = require RACINE . '/.config/partenaire.php';
$lesParametres = json_encode($lesParametres);
```

```
$head =<<<EOD
<script >
  let lesPartenaires = $lesPartenaires;
  let lesParametres = $lesParametres;
  const token = '$token';
</script>
EOD;
```

## 5.2. Le fichier administration/index.js

```
function afficher() {
  ...
  for (const partenaire of lesPartenaires) {
    const id = partenaire.id;
    ...
    carte.id = id;
    ...
    // génération de l'entête

    // ajout de l'icône de suppression en haut à droite du cadre
    ...
    i.onclick = () => confirmer() => supprimer(partenaire.id));
    entete.appendChild(i);

    // génération du corps de la carte
    ...
    // ajout d'une balise input pour la modification du nom
    ...
    inputNom.type = 'text';
    inputNom.value = partenaire.nom;
    inputNom.dataset.old = partenaire.nom;
    inputNom.pattern = '^[0-9A-Za-
    inputNom.maxLength = 20;
    inputNom.onchange = function () {
      this.value = this.value.replace(/s{2,}/, ' ').trim();
      if (this.value !== this.dataset.old && verifier(this)) {
        modifierColonne('nom', this, partenaire.id);
      }
    };
    corps.appendChild(inputNom);

    // ajout d'une balise input pour la modification de l'url
    ...
    inputUrl.type = 'url';
    inputUrl.value = partenaire.url;
    inputUrl.dataset.old = partenaire.url;
    inputUrl.onchange = function () {
      if (this.value !== this.dataset.old && verifier(this)) {
        modifierColonne('url', this, partenaire.id);
      }
    };
    corps.appendChild(inputUrl);
```

```
// Ajout de la zone d'upload
div.id = 'photo' + id;
// Si la photo existe on la place dans la zone
if (partenaire.present) {
  const img = document.createElement('img');
  img.src = lesParametres.repertoire + '/' + partenaire.fichier;
  img.alt = "";
  div.appendChild(img);
}
// définition des événements pour gérer le téléversement et le glisser déposer
div.onclick = function () {
  idPartenaire = id;
  fichier.click();
};
div.ondragover = function (e) {
  e.preventDefault();
};
div.ondrop = function (e) {
  idPartenaire = id;
  e.preventDefault();
  controlerFichier(e.dataTransfer.files[0]);
};
...
}
}

function modifierColonne(colonne, input, id) {
$.ajax({
  url: '/ajax/modifiercolonne.php',
  method: 'post',
  data: {
    table : 'partenaire',
    colonne: colonne,
    valeur: input.value,
    id: id,
    token: token
  },
  dataType: 'json',
  success: data => {
    if (data.success) {
      input.dataset.old = input.value;
      input.style.color = 'green';
    } ...
  });
}
```



```
function supprimer(id) {
    $.ajax({
        url: '/ajax/supprimer.php',
        method: 'POST',
        data: {
            table: 'partenaire',
            id: id,
            token: token
        },
        dataType: 'json',
        success: data => {
            if (data.success) {
                lesPartenaires.splice(lesPartenaires.findIndex(x => x.id === id), 1);
                afficher();
            } ...
        });
}
```

## 5.3. Le fichier ajout.php

```
// Récupération des paramètres du téléversement
$lesParametres = require RACINE . '/.config/partenaire.php';
$lesParametres = json_encode($lesParametres);
$head =<<<EOD
<script>
    let lesParametres = $lesParametres;
    const token = '$token';
</script>
EOD;
```

## 5.4. Le fichier ajout.js

```
function ajouter() {
    let monFormulaire = new FormData();
    monFormulaire.append('table', 'partenaire');
    monFormulaire.append('fichier', leFichier);
    monFormulaire.append('nom', nom.value);
    monFormulaire.append('url', url.value);
    monFormulaire.append('token', token);
    $.ajax({
        url: '/ajax/ajouter.php',
        method: 'POST',
        async: false,
        data: monFormulaire,
        processData: false,
        contentType: false,
        dataType: 'json',
        success: (data) => {
            if (data.success) {
                retournerVers("Partenaire ajouté", '.');
            } ...
        }
    })
}
```