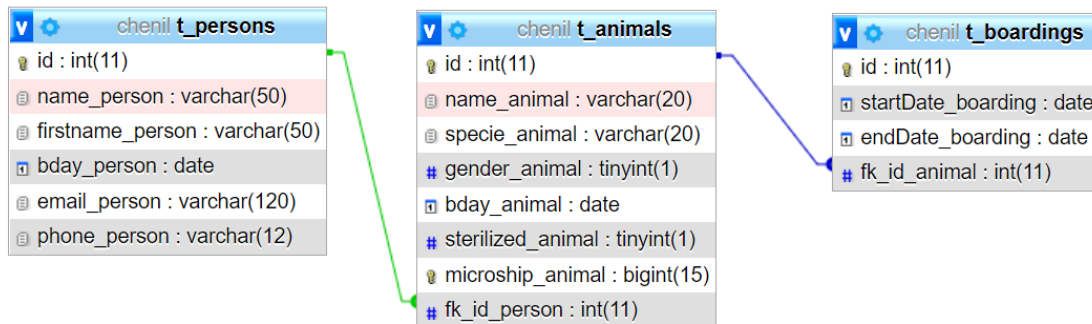


Examen 5IDV2 - Juin2022 Pauwels Donatian

Schéma relationnel base de données:



Ma base de donnée se constitue de trois table:

- **t_persons:** Reprenant toutes les informations liées à une personne.
 - **id** en INT (Servant de clef primaire)
 - **Nom** en VARCHAR
 - **Prenom** en VARCHAR
 - **Date d'anniversaire** en DATE
 - **Email** en VARCHAR
 - **Téléphone** en VARCHAR
(Le téléphone aurait pu être au format INT, mais je dû faire face à un soucis ma base de donnée effaçait constamment le 0 si il était en premier caractère).
- **t_animals:** Reprenant toutes les informations liées à un animal.
 - **id** en INT (Servant de clef primaire)
 - **Nom** en VARCHAR
 - **Espèce** en VARCHAR
 - **Genre** en TINYINT (Ceci est l'alias de Mysql pour le booléen)
 - **Date d'anniversaire** en DATE
 - **Sterilized** en TINYINT (Encore un alias)
 - **Microship** en BIGINT et UNIQUE
(Le microship aurait pû servir d'ID pour l'animal, étant logique unique par animal, seulement j'ai rendu l'édition possible du numéro de puce au cas où celle-ci devrait être changé (puce illisible, défectueuse))
 - **Forgein Key** => Qui lie une personne par son ID à un animal, le nom de la constraint est **fk_animals_persons**
- **t_boardings:** Reprenant toutes les informations liées à un séjour.
 - **id** en INT (Servant de clef primaire)
 - **Date de début** en DATE
 - **Date de fin** en DATE
 - **Forgein Key** => Qui lie un animal par son ID à un séjour, le nom de la constraint est **fk_boardings_animals**

Liste des routes et explication

Liste pour animal:

GET /animal

Réponse	Type	Status	Description
Liste d'animal	Document	200 - ok	Renvoie la page entière de la liste des animaux

GET /animal/create

Réponse	Type	Status	Description
Formulaire création d'un animal	Document	200 - ok	Renvoie la page entière pour ajouter un nouvel animal dans la DB

POST /animal/store

Paramètre	Type	Status	Description
Data	Array	Obligatoire	Data contenant les informations de l'animal

Réponse	Type	Status	Description
Data	Document / Redirect	302 - Found	Réponse reçue si Data conforme ou non.

GET /animal/show/{id}

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id de l'animal

Réponse	Type	Status	Description
	Document	200 - ok	Si l'Id correspond bien à un animal
	text/html	302 - Found	Si l'Id ne correspond à aucun animal

POST /animal/delete

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id de l'animal

Réponse	Type	Status	Description
	text/html	302 - Found	Si l'Id correspond bien à un animal

GET /animal/edit/{id}

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id de l'animal

Réponse	Type	Status	Description
	Document	200 - ok	Si l'Id correspond bien à un animal
	text/html	302 - Found	Si l'Id ne correspond à aucun animal

POST /animal/update

Paramètre	Type	Status	Description
Data	Array	Obligatoire	Data de l'animal

Réponse	Type	Status	Description
Data	Document / Redirect	302 - Found	Réponse reçue si Data conforme ou non.

Liste pour Person:

GET /person

Réponse	Type	Status	Description
Liste de personne	Document	200 - ok	Renvoie la page entière de la liste des personnes

GET /person/create

Réponse	Type	Status	Description
Formulaire création d'une personne	Document	200 - ok	Renvoie la page entière pour ajouter une nouvelle personne dans la DB

POST /person/store

Paramètre	Type	Status	Description
Data	Array	Obligatoire	Data contenant les informations de la personne

Réponse	Type	Status	Description
Data	Document / Redirect	302 - Found	Réponse reçue si Data conforme ou non.

GET /person/show/{id}

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id de la personne

Réponse	Type	Status	Description
	Document	200 - ok	Si l'Id correspond bien à une personne
	text/html	302 - Found	Si l'Id ne correspond à aucune personne

POST /person/delete

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id de la personne

Réponse	Type	Status	Description
	text/html	302 - Found	Si l'Id correspond bien à une personne

GET /person/edit/{id}

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id de la personne

Réponse	Type	Status	Description
	Document	200 - ok	Si l'Id correspond bien à une personne
	text/html	302 - Found	Si l'id ne correspond à aucune personne

POST /person/update

Paramètre	Type	Status	Description
Data	Array	Obligatoire	Data de la personne

Réponse	Type	Status	Description
Data	Document / Redirect	302 - Found	Réponse reçue si Data conforme ou non.

Liste pour Boarding:

GET /boarding

Réponse	Type	Status	Description
Liste de séjour	Document	200 - ok	Renvoie la page entière de la liste des séjour

GET /boarding/create

Réponse	Type	Status	Description
Formulaire création d'un séjour	Document	200 - ok	Renvoie la page entière pour ajouter un nouveau séjour dans la DB

POST /boarding/store

Paramètre	Type	Status	Description
Data	Array	Obligatoire	Data contenant les informations d'un séjour

Réponse	Type	Status	Description
Data	Document / Redirect	302 - Found	Réponse reçue si Data conforme ou non.

GET /boarding/show/{id}

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id d'un séjour

Réponse	Type	Status	Description
	Document	200 - ok	Si l'Id correspond bien à un séjour
	text/html	302 - Found	Si l'Id ne correspond à aucun séjour

POST /boarding/delete

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id d'un séjour

Réponse	Type	Status	Description
	text/html	302 - Found	Si l'Id correspond bien à un séjour

GET /boarding/edit/{id}

Paramètre	Type	Status	Description
Id	Int	Obligatoire	Id d'un séjour

Réponse	Type	Status	Description
	Document	200 - ok	Si l'Id correspond bien à un séjour
	text/html	302 - Found	Si l'Id ne correspond à aucun séjour

POST boarding/update

Paramètre	Type	Status	Description
Data	Array	Obligatoire	Data d'un séjour

Réponse	Type	Status	Description
Data	Document / Redirect	302 - Found	Réponse reçue si Data conforme ou non.

Problèmes rencontré:**Design Pattern**

- 1) Mon premier problème rencontré a été au niveau du design pattern strategy, même si l'idée est maintenant comprise, le fonctionnement réel me posait quelques petits problèmes.

J'ai donc dû rechercher de la documentations pour m'assurer de comprendre, voici les deux sources qui ont servis de base pour mon Design Pattern

https://sourcemaking.com/design_patterns/strategy/php

Et

https://www.tutorialspoint.com/design_pattern/strategy_pattern.htm

C'était un problème uniquement de connaissance mais cela m'a permis d'implémenter mon design pattern, la class CANIN et FÉLIN on une fonction toEat() qui renvoie un echo différent si on appelle la fonction toEat() de la class parent.

Date

- 2) Un second problème à été de récupérer la date du jour dans un bon format, encore une fois il s'agit plutôt d'une connaissance pas totalement approfondie de PHP, avec un peu de recherche, j'ai fini par trouver la fonction getDate().

Cette fonction me permet de récupérer divers données de la date qu'il m'a suffi de travailler pour obtenir le format de date voulu.

Link

- 3) Un troisième problème c'est posé, mais au niveau cette fois ci de la liaison de mon CSS avec mon code html, avec le routeur certaine page ne chargeais pas mon html. mon premier link était celui-ci:

```
<link href="css/css.css" rel="stylesheet">
<title>Liste propriétaire</title>
```

Ce link marche pour certaine page mais d'autre non, j'ai du adapter mon lien en ceci:

```
<link href="/css/css.css" rel="stylesheet">
<title><?= $person->name ?> <?= $person->firstname ?></title>
```

Pour que celui-ci fonctionne correctement sur toutes les pages.

En essayant de comprendre ce qu'il se passait avec mon premier lien le DevTools me donne ceci:

37	200	document	Other	1.7 kB	61 ms
<input checked="" type="checkbox"/> css.css	302	stylesheet / Redirect	37	368 B	104 ms
<input checked="" type="checkbox"/> home	200	stylesheet	css.css	2.0 kB	229 ms
<input type="checkbox"/> favicon.ico	200	text/html	Other	2.0 kB	256 ms

On remarque que ma page "37" appelle le css.css mais celui-ci appelle home en feuille de style(cette feuille de style n'existe pas).

Pourquoi certaines pages fonctionnent et d'autres non ?

Pour moi cela vient du router (j'en suis presque totalement convaincu) et plus précisément à sur c'est ligne si:

```
//Permet de couper la chaîne de caractere dans un tableau selon le caractere choisi (ici "/").
//Substr => retourne la chaîne a partir du caractere donnée
$serveur_uri = explode("/", substr($serveur_uri, 1));
return $serveur_uri;
```

J'aimerais en discuter avec vous pour confirmer ma pensée.

Hormis ces problèmes là, je n'ai pas rencontré trop de difficultés.

Fonctionnalités implémentées

- Tableau de bord qui affiche:
 - Le nombre total de personnes inscrites.
 - Le nombre total d'animaux enregistré.
 - Le nombre total de séjour.
 - Les séjours commencent à la date du jour.
 - Les séjours se terminant à la date du jour.
- La vue détaillée d'une personne charge tous les animaux dont elle est le maître.
- La vue détaillée d'un animal charge tous les séjours de l'animal.
- Sur la vue séjour, un séjour qui à déjà commencé ou qui est terminé, ne pourra plus être modifié ou supprimé.
 - L'effet recherché en plus d'appliquer une forme de tri visuel est une persistance des données, en effet puisqu'un animal ne peut pas être supprimé s'il a un séjour et qu'un propriétaire ne peut pas l'être s'il a un animal.
Verrouiller un séjour, empêche de supprimer les deux autres infos, l'intégralité de l'historique est donc préservée.
- La vue détaillée d'un animal tri aussi les séjours en deux catégories
 - Séjour en cours ou séjour futur
 - Séjour passé (Historique de séjour).
- Vérification qu'un séjour ne rentre pas en conflit avec un autre séjour pour un même animal.
- Souvenirs

Autocritique

Premièrement, la simplicité de mon code, je trouve mon code beaucoup trop pauvre en fonctionnalité il est bien en dessous de ce que je suis capable de fournir (si le temps m'aurait permis), j'avais pourtant quelque idée qui aurait apporté pas mal de challenge et de recherche.

- Un système de login (qui aurait tout son sens ici).
- Un fonction qui empêche de saisir une date dans le passé.
- Implémenter du JS/Jquery.

J'ai pris énormément de plaisir à faire ce code et ne pas le finir comme je le voulais me frustre pas mal.

Ma fonction pour update un séjour n'est pas non plus terminée, je n'ai pas su l'amener jusqu'au je voulais, les champs ont donc été verrouillé pour éviter de créer des erreurs.

Mon Design Pattern est simple, je ne fais qu'un ECHO, j'aurais aimé pouvoir développer un peu plus cette fonctionnalité.

Mon code se rapproche plus d'un MVP qu'un MVC.

Conclusion:

L'examen était un relevé de challenge hyper intéressant bien mené et avec plein de possibilités.

Si on ajoute à ceci l'intégralité du cours, nous avons pu voir grâce à vous la possibilité infini qu'offre le métier de développeur (même si ceci tout ce cours n'est qu'une toute petite partie de l'iceberg).

Tout ce qu'on à vu me permet de voir déjà un peu plus clair pour mon futur TFE, la montagne qu'il représente est devenue un peu moins grosse.

Je tenais à vous remercier pour cette agréable découverte en votre compagnie !