1.

Date : 15 mai 2020

Défi rencontré : Je ne savais pas comment ouvrir le modèle logique dans SQL developper.

La solution trouvée : Je dois aller dans View -> Data modeler -> Browser ensuite clique doit sur Logical Model -> Show

L’apprentissage : J’ai appris comment ouvrir le modèle logique dans SQL developper.

2.

Date : 15 mai 2020

Défi rencontré : Je ne savais pas comment transformer mon modèle logique en script de création de base de données.

La solution trouvée : Je dois faire un clique doit sur Logical Model -> Engineer to relational model -> Generate DDL -> Generate -> Ok

L’apprentissage : J’ai appris à créer une base de données à partir d’un modèle logique.

3.

Date : 16 mai 2020

Défi rencontré : Quand je voulais REST enable j’avais une erreur.

La solution trouvée : Le p\_schema devait etre en majuscules comme ça : => 'RESTSCOTT' et le p\_object aussi : => 'UTILISATEUR'

L’apprentissage : J’ai appris comment REST enable des tables dans ma base de données.

4.

Date : 18 mai 2020

Défi rencontré : Je ne savais pas comment insérer des éléments de ma page web (input de l’utilisateur dans ma base de données).

La solution trouvée : J’ai trouvé qu’avec le méthode fetch(URL) où URL = notre lien REST avec method: "POST" en passant nos tous valeurs (dans la variable uti\_rec) body: JSON.stringify(uti\_rec).

L’apprentissage : J’ai appris comment utiliser fetch sur javascript.

5.

Date : 19 mai 2020

Défi rencontré : Quand je me créer un compte sur mon site web afin d’insérer des éléments dans ma base de données je ne voyais pas les nouveaux éléments dans ma base de données.

La solution trouvée : J’ai appris qu’on peut utiliser la console pour voir les erreurs. Dans mon HTML j’ai laissé l’ancienne méthode qui était ins\_emp1().

L’apprentissage : J’ai appris comment utiliser la console pour voir mes erreurs.

6.

Date : 19 mai 2020

Défi rencontré : Je ne savais pas comment créer des options à choix multiples sur la page pari.

La solution trouvée : On peut utiliser des balises input de type="radio" où le name sert à identifier les éléments de la liste où on veut choisir un de ses éléments.

L’apprentissage : J’ai appris comment utiliser les inputs de type="radio".

7.

Date : 21 mai 2020

Défi rencontré : Je voulais insérer ses inputs sur la page pari à partir des différents paris sur ma base de données.

La solution trouvée : Encore une fois on peut utiliser la méthode : fetch(URL) Ensuite avec ses deux méthodes :

function createNode(element) {

 return document.createElement(element);

}

function append(parent, el) {

 return parent.appendChild(el);

}

on peut créer des balises dans le HTML avec le javascript. J’ouvre mes balises comme ça :

let input = createNode("INPUT");

span = createNode("span");

br = createNode("br");

Et je les ferme comme ça :

append(pari\_div, input);

append(pari\_div, span);

append(pari\_div, br);

L’apprentissage : J’ai appris comment créer des balises HTML à partir du javascript

8.

Date : 21 mai 2020

Défi rencontré : Je ne savais pas comment insérer des attributs dans ma balise input (je voulais qu’elles soient de type="radio" avec name="pari" et value= "la cote à partir de ma base de données"

La solution trouvée :

input.setAttribute("type", "radio");

input.setAttribute("name", "paris"),

input.setAttribute("value", innerHTML = `${pari.cote}` + 'x'),

L’apprentissage : J’ai appris comment utiliser la méthode setAttribute() et le innerHTML

9.

Date : 23 mai 2020

Défi rencontré : Je voulais générer les paris de la première équipe à gauche et les paris de la deuxième équipe à droite.

La solution trouvée : J’ai dupliqué la fonction fetch et j’ai mis 2 div dans mon HTML avec 2 id différents (pari et pari2). Ensuite j’ai mis un :

if (pari.equipe\_id\_equipe === id\_equipe2 && pari.statut === 'disponible')

(=== id\_equipe1 pour la première)

J’ai fini par :

append(pari\_div2, input);

append(pari\_div2, span);

append(pari\_div2, br);

L’apprentissage : J’ai appris comment utiliser la même table dans ma base de données pour 2 divs différentes.

10.

Date : 27 mai 2020

Défi rencontré : Je voulais créer une fonction qui affiche la cote du pari qu’on a choisi

La solution trouvée : J’ai fait une boucle for qui regarde si un input radio est coché et j’affiche son "value" que j’ai initialiser avec pari.cote :

function cote() {

 var ele = document.getElementsByName('paris');

 for (i = 0; i < ele.length; i++) {

  if (ele[i].checked)

   document.getElementById("cote").innerHTML = ele[i].value;

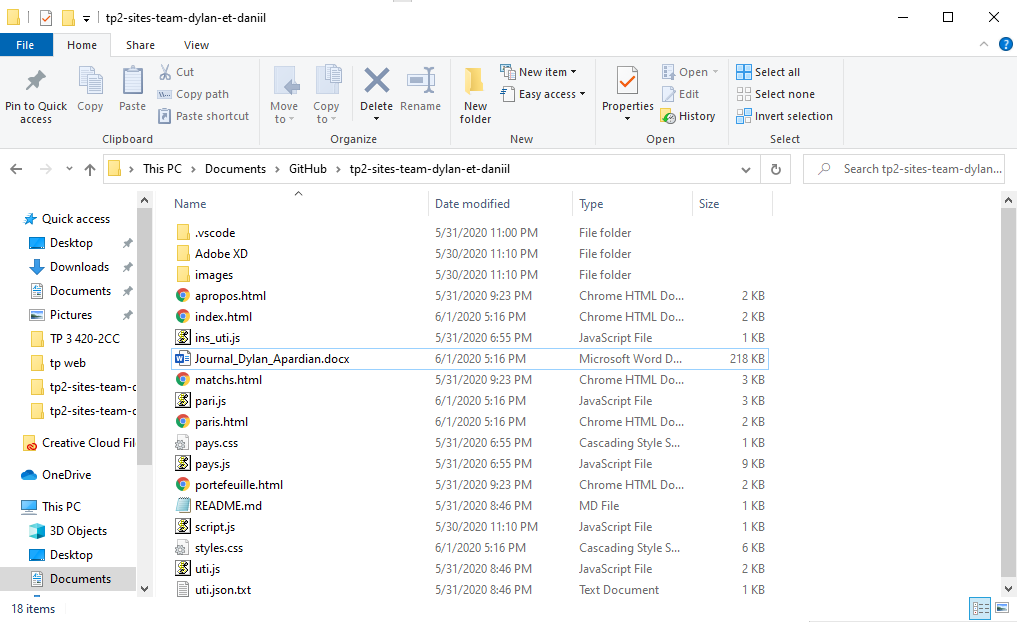
 }

L’élément avec le id="cote" est juste un paragraphe dans le HTML.

L’apprentissage : J’ai appris qu’avec javascript on peut regarder si on élément est coché avec le : element.checked

**Photos GIT**

Voici mon répertoire local pour le TP2



**GIT log**

