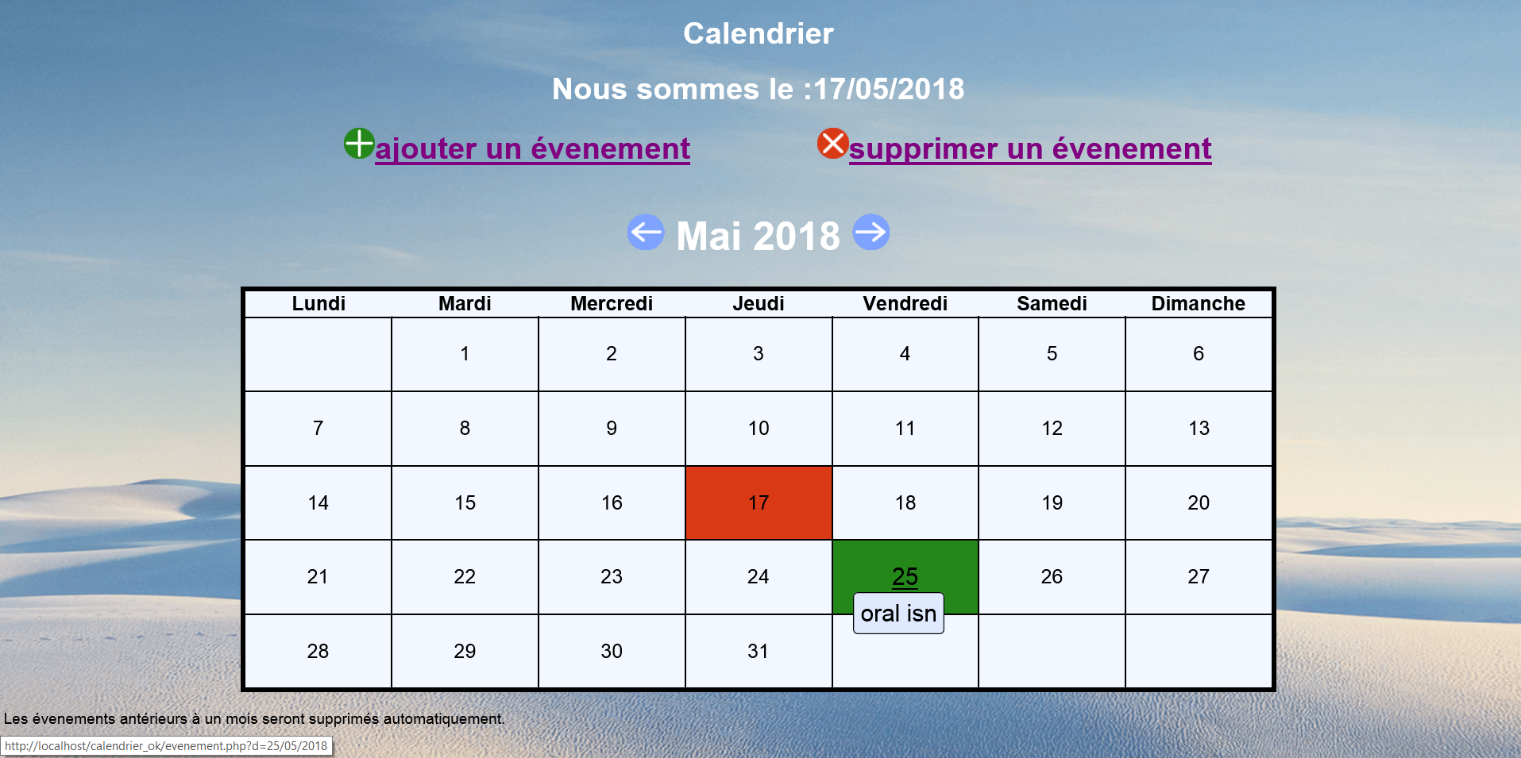
Notice du calendrier



Projet d’ISN

2017/2018

Lycée Antoine Bourdelle

LORENZO Julien

MENGELLE Alexandre

ZOCCA—LAMARTINIERE Vincent

Présentation du projet

Réalisation d’un calendrier permettant d’enregistrer et supprimer des mémos avec photos afin de personnaliser celui-ci.

Son but est aussi de supprimer les événements ayant déjà eu lieu dans l'intention de protéger la vie privée de l'utilisateur et de ne pas saturer la base de données dans temps.

Les outils :

Afin de mener à bien ce projet nous programmerons en html, PHP, CSS et nous utiliserons une base de données en SQL. Dans le but de pouvoir exécuter ces différents langage nous avons utilisé Xampp qui nous permet de mettre en place des serveurs WEB locaux.

Pourquoi ce projet?

Nous voulions que notre calendrier regroupe un large panel de fonctionnalités, permettant à l’utilisateur de faciliter son utilisation et de protéger sa vie privée qui semble être un concept flou de nos jours. Nous avons réalisé notre propre vision d’un calendrier d’aujourd’hui : intuitif, personnalisable et simple d’utilisation.

Lancement du projet :

Tout d’abords nous avons créé l’ossature du calendrier (calendrier.php) en modifiant des programmes déjà existant sous PHP, HTML. Nous avons ensuite crée une base de données locale grâce à MySQL qui nous permet de stocker et de modifier des données. Pour relier ces deux entités on a mis en place un programme de connection en PHP (sql\_connect). L’ajout d’un évènement (ajoutevent.php) et sa suppression (supprevent.php) fut dans la suite des choses pour le bon fonctionnement de notre calendrier. Dans l’intention de consulter les évènements nous créons un programme (evenement.php) en PHP nous permettant de consulter le ou les évènements enregistrés. Pour un visuel agréable on utilise un programme CSS (calendrier.css).

Répartition des taches :

* CSS et interconnections entre les programmes Julien
* Calendrier et base de données Alexandre
* Insertion d’évènement et base de données Vincent
* Améliorations et résolutions des problèmes en groupe

Schéma des liens entre les différents programmes :

Calendrier

*calendrier.php*

Visuel en CSS

*design/calendrier.css*

Suppressions évènements

*supprevent.php*

Connection à la Base de données SQL

*sql\_connect.php*

Consultation des événements

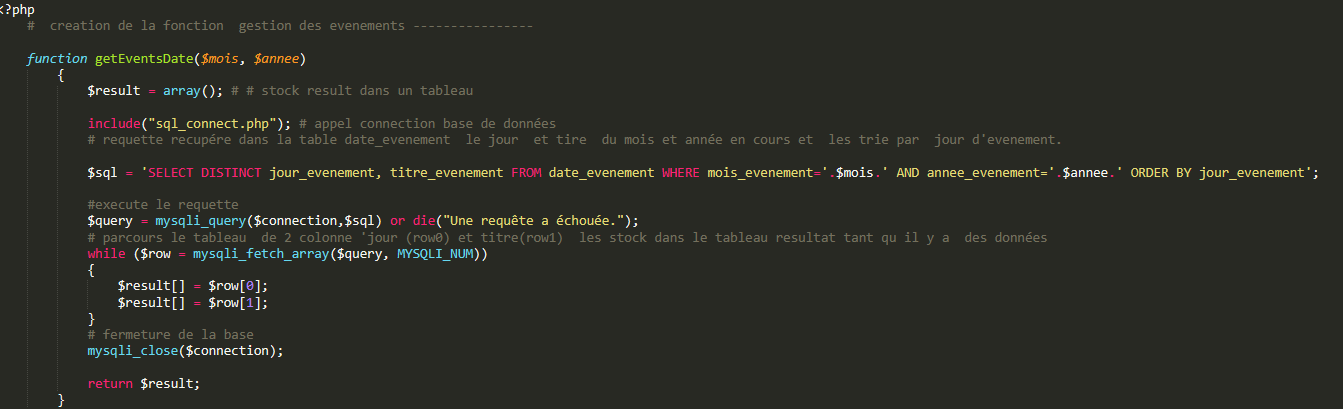
*evenement.php*

Enregistrement des évènements et images

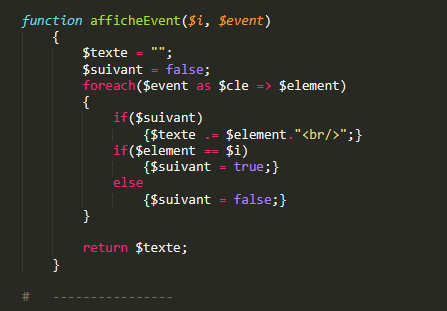
*ajoutevent.php*

Descriptif du programme

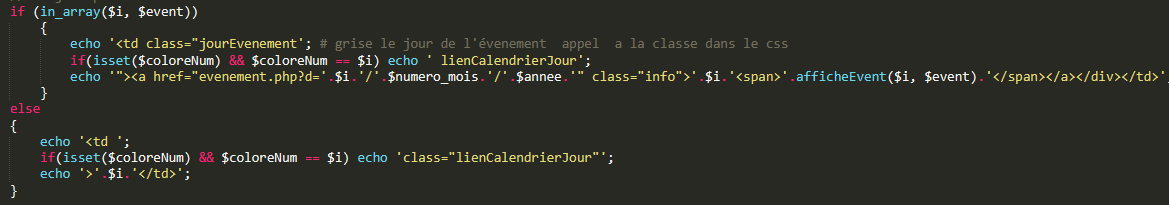
Calendrier.php



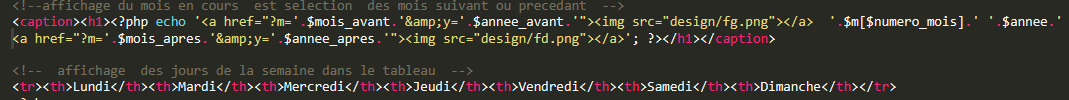
Fonction récupérant les dates des évènements grâce à requête en SQL.



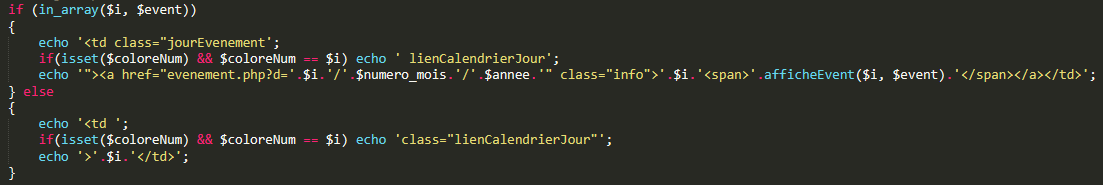
Fonction recherchant tous les évènements.



Appel de la fonction précédente pour afficher les événements en pop-up.



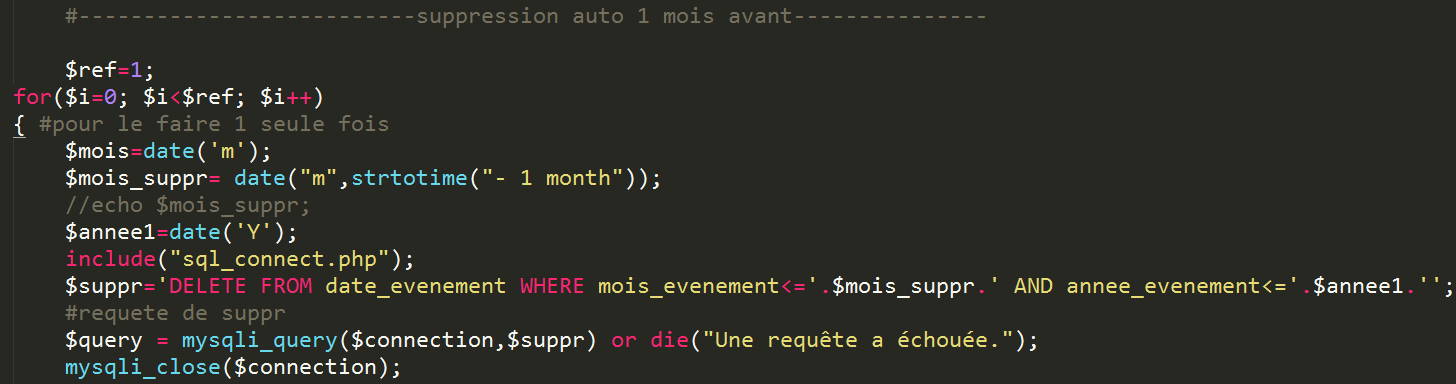
Affichage des jours ; des mois en cours, et un bouton pour faire précédant et suivant avec l’année. Caption désigne la zone juste au-dessus du tableau. Dans cette zone on y met le mois et l'année en cours encadrée par deux flèches permettant de défiler de mois en mois. La dernière ligne du code nous montre la 1ère ligne du tableau rempli par le nom des jours.



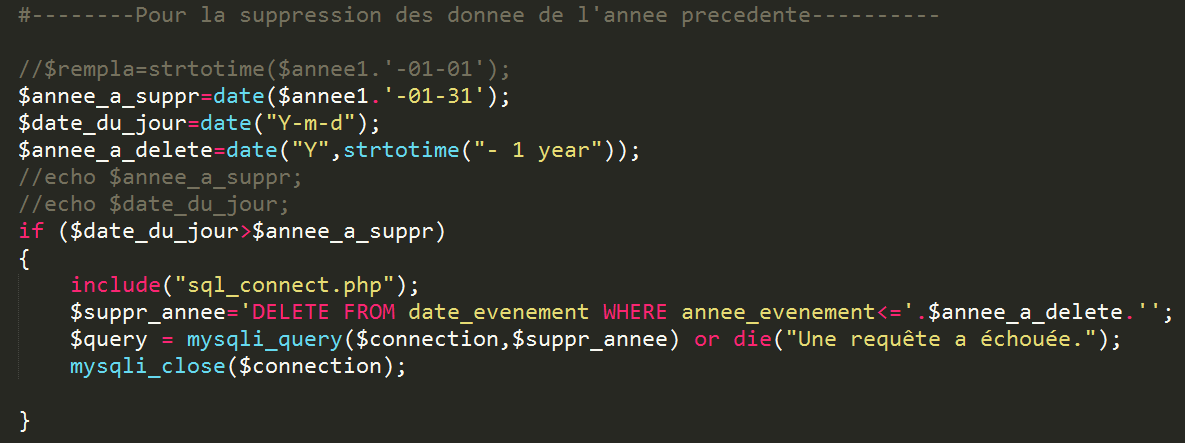


Colorie la case s’il y a un événement avec un appel de fonction CSS.

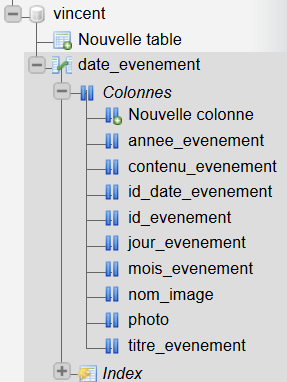
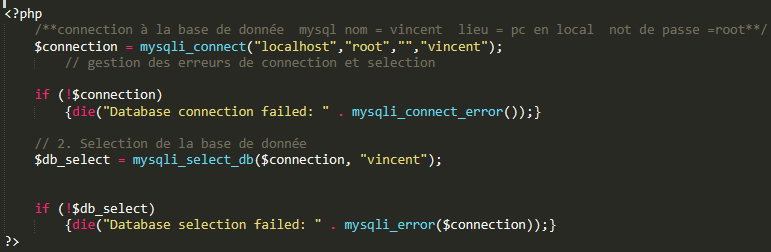
Présentation de la suppression automatique:



On utilise une boucle for afin de ne pas faire tourner ce programme en boucle. On créé une variable de temps à J-1mois et pour tout événements inférieurs à cette date, on les supprime. Cependant cette commande ne marche que pour les variables de l'année en cours. Pour pallier à cela, on fait une autre fonction :

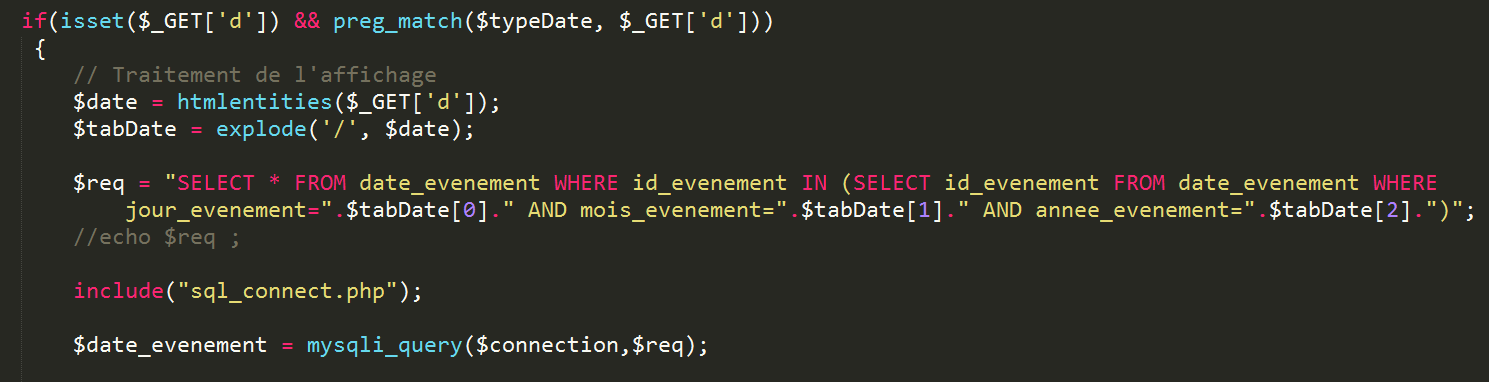


Ici on efface tous les événements des années précédant l'année en cours. Pour cela on créé une variable récupérant l'année précédente grâce à la fonction date et à strtotime qui nous permet de la modifier. On utilise une requête pour supprimer tous les événements de l'année précédente et inférieure.

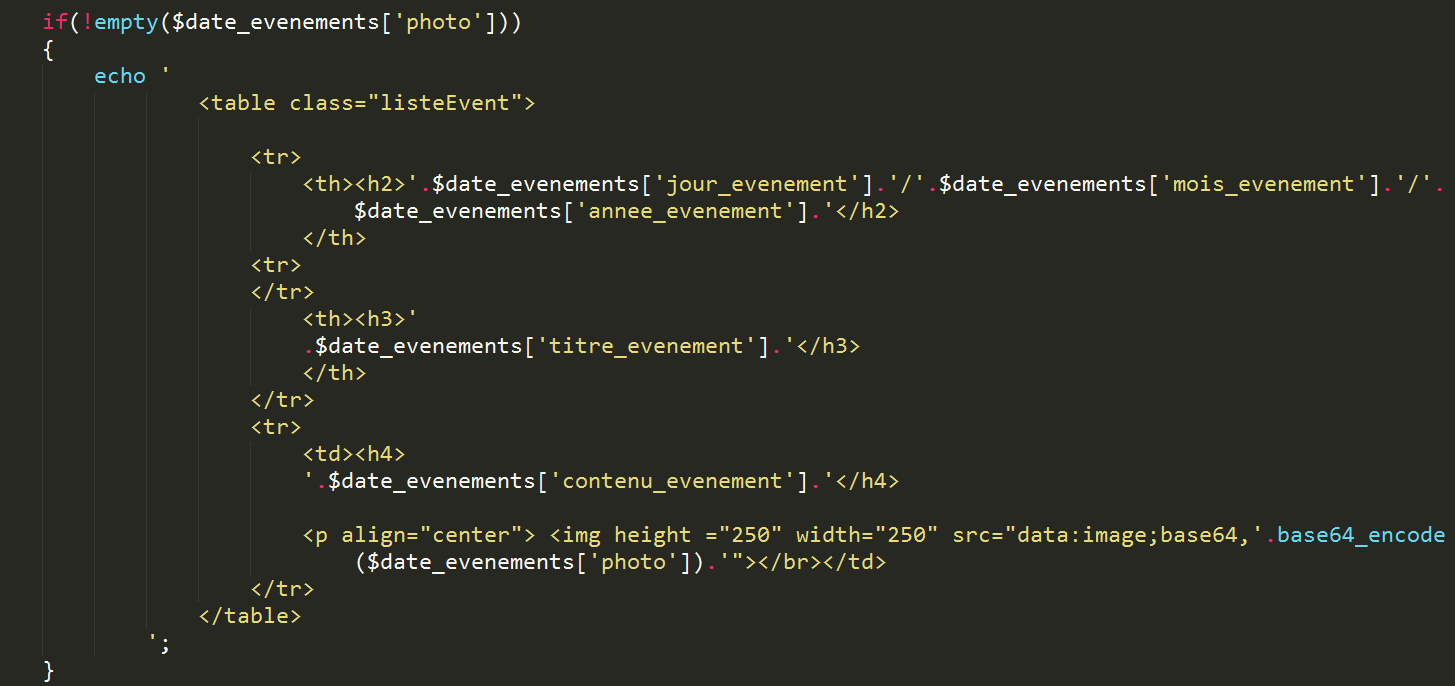


Connection entre la base de donnée SQL et le programme en PHP. La fonction mysqli\_connect nous permet de relier le programme en se connectant à la base de données phpmyadmin nommé vincent où root est le nom d'utilisateur de base et localhost le nom de l'hébergeur.

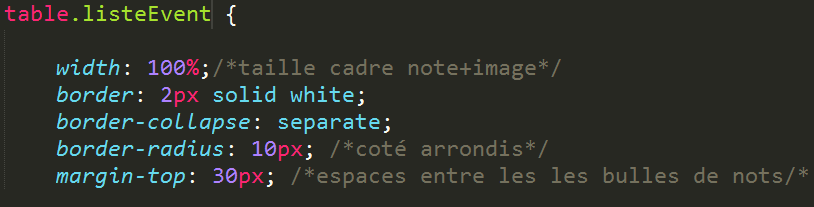
Evenement.php



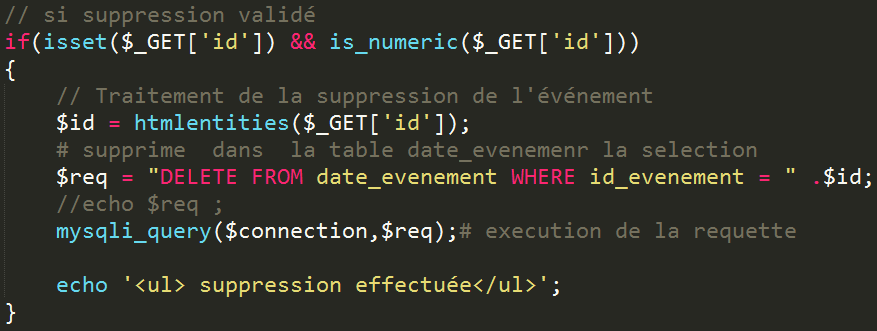
Récupère la date en une seule variable, puis la découpe en 3 groupes: jour, mois, année. On sélectionne ensuite la totalité des données des événements grâce à l'identifiant de l'événement pour lesquelles la date correspond à l'aide d'une requête SQL. Ces données vont être stockées afin de les afficher :



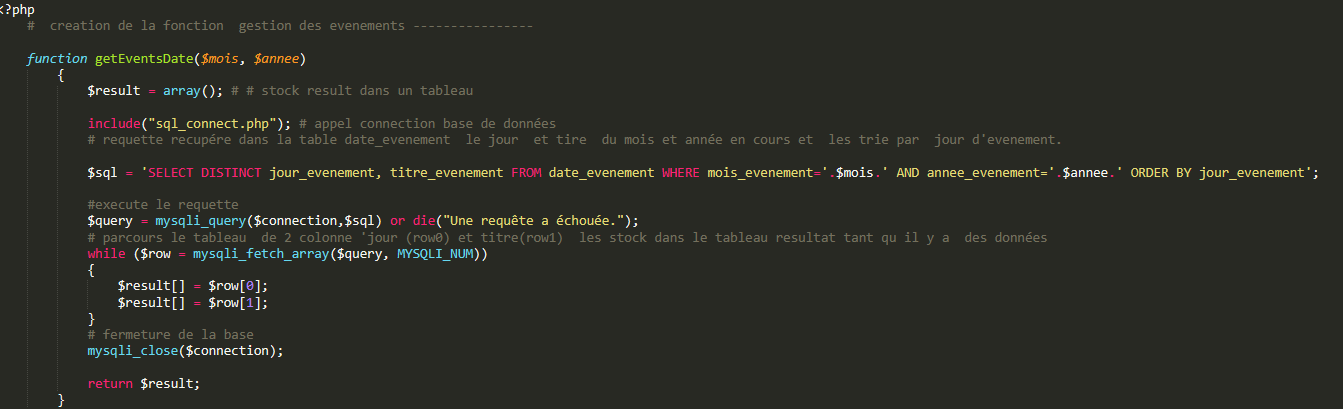
Elles vont être affichées grâce au code ci-dessus et centré avec le CSS. Le code ci-dessus affiche lorsqu’une image est sélectionnée, cependant le code où aucune image n’est sélectionnée diffère peu.

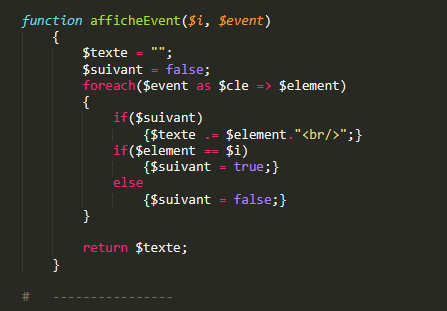


Supprevent.php

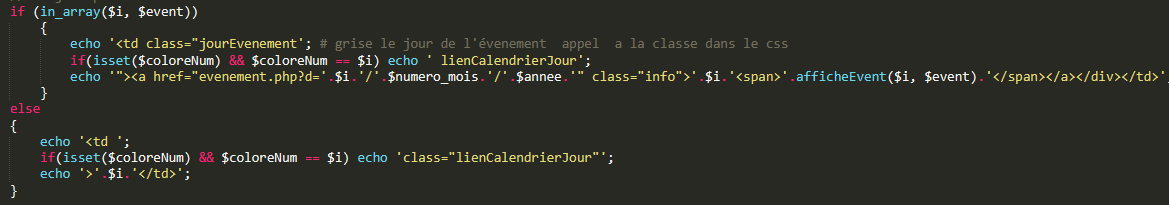


Lorsque l'on clique sur le bouton suppression, on envoie alors une requête SQL afin de supprimer l'événement dont l'id a été sélectionné.

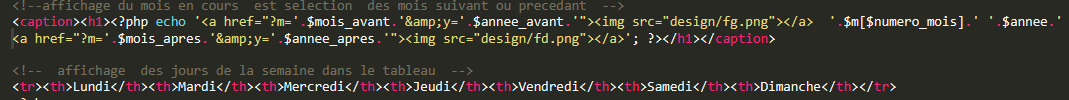


Fonction récupérant les dates des évènements grâce à requête en SQL.

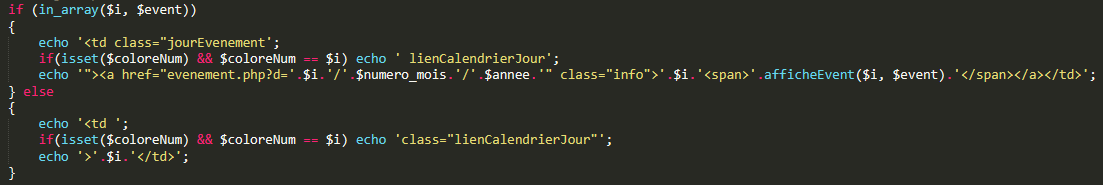
Fonction recherchant tous les évènements.



Appel de la fonction précédente pour afficher les événements en pop-up.



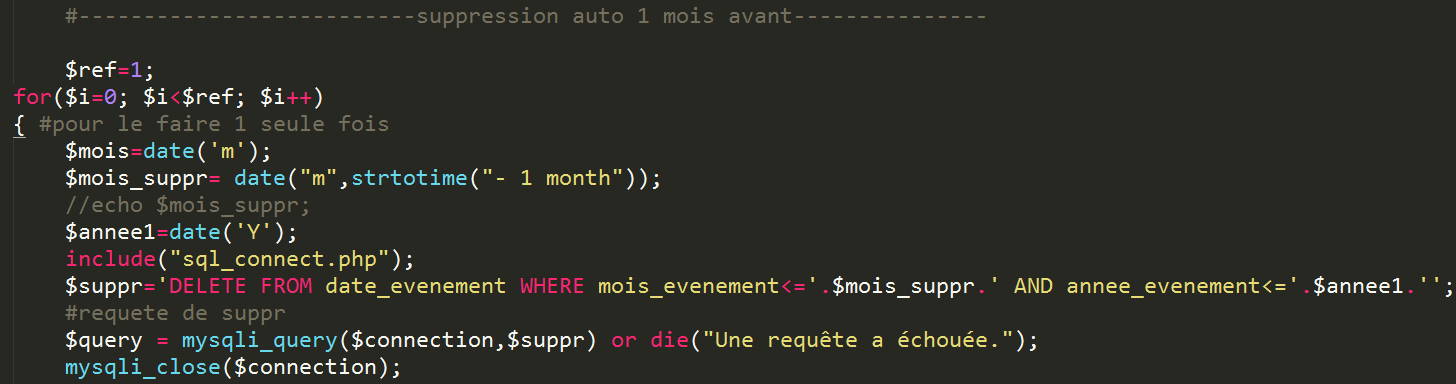
Affichage des jours ; des mois en cours, et un bouton pour faire précédant et suivant avec l’année.



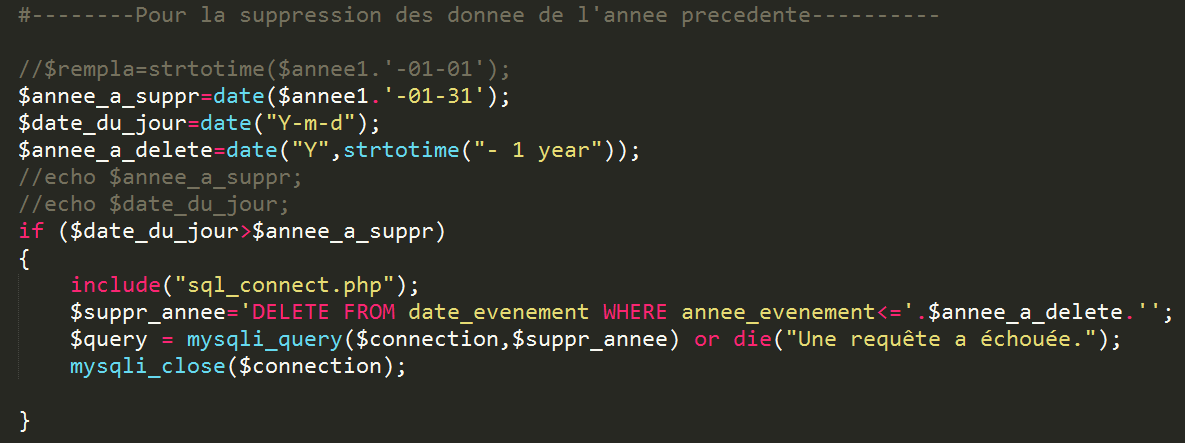


Colorie la case s’il y a un événement avec un appel de fonction css.

Présentation de la suppression automatique:

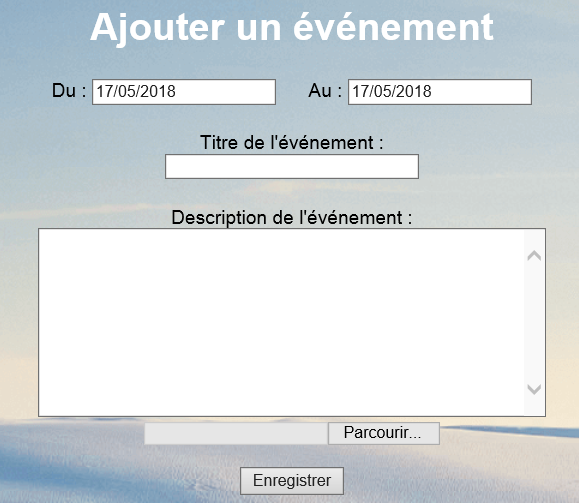


On utilise une boucle for afin de ne pas faire tourner ce programme en boucle. On créé une variable de temps à J-1mois et pour tout événements inférieurs à cette date, on les supprime. Cependant cette commande ne marche que pour les variables de l'année en cours. Pour pallier à cela, on fait une autre fonction :



Ici on efface tous les événements des années précédant l'année en cours.

Ajoutevent.php

 Le programme ajoutevent.php permet la saisie et l’enregistrement de nouveaux évènements :

* Dates
* Titre
* Description
* Image

Tous ces paramètres seront enregistrer dans

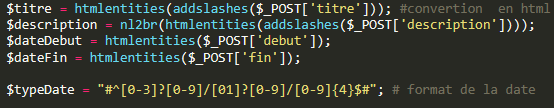
la base de donnée MySQL.

 Gestion des boutons compris dans la balise « form » (formulaire) par la méthode « POST » qui récupère les différentes données saisies dans les zones textes.

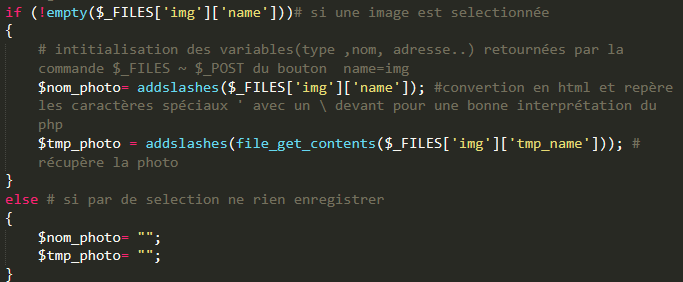
Lorsqu’un bouton est actionner celui-ci permet le retour au début de la page « boucle » (action="ajoutevent.php»).



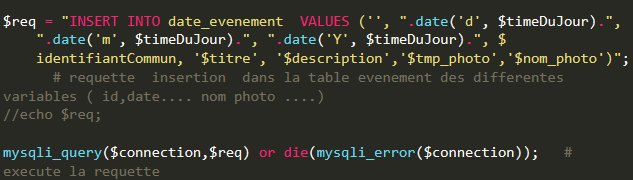
Si un évènement (titre…) est saisi. On exécute la boucle si dessous.



Récupération des valeurs saisies : titre, description …. Grace à la commande « $\_POST ».

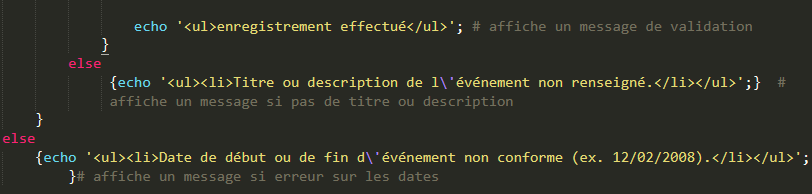


Si le nom de l’image existe, on récupère son nom dans « $nom\_photo » et toute les caractéristiques de la photo. Sinon on n’enregistre rien.

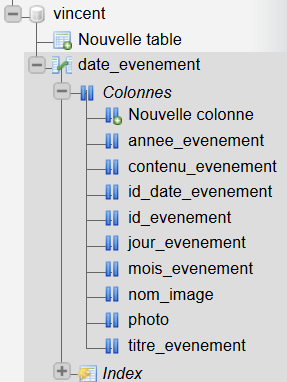
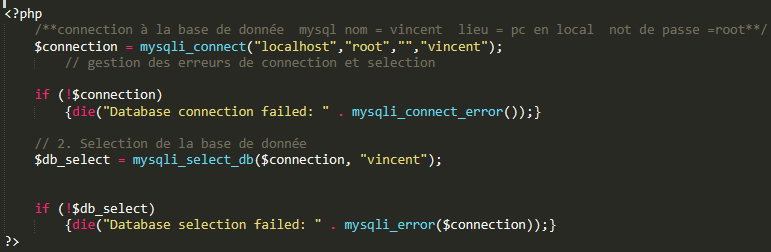


Requête permettant l’enregistrement de la date, titre, image… dans la base MySQL.

La commande « mysqli\_query » exécuté cette requête si aucune erreur de connection avec la base de données est détectée.

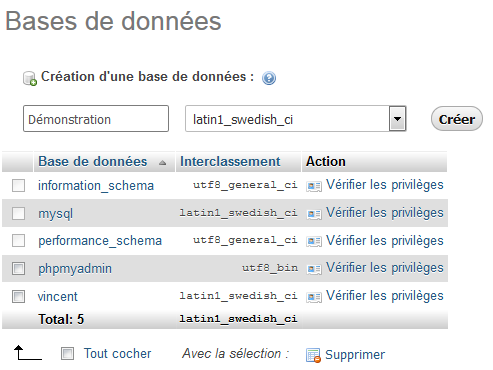


Affiche des informations pour l’utilisateur sur le résultat de ses actions (retour des conditions « if …. ».

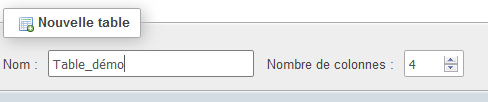
Sql\_connect.php

Connection entre la base de donnée SQL et le programme en PHP.la fonction msqli\_connect nous permet de relier le programme en se connectant à la base de données nommé vincent ou root est le nom d’utilisateur de base et localhost le nom de l’hebergeur.

Création de la Base donnée

Tout d'abord on créé la base de données depuis phpmyadmin:

On créé à l'intérieur une table avec un nombre de colonnes souhaitées qui vont permettre d'y stockés les valeurs ou données voulues.

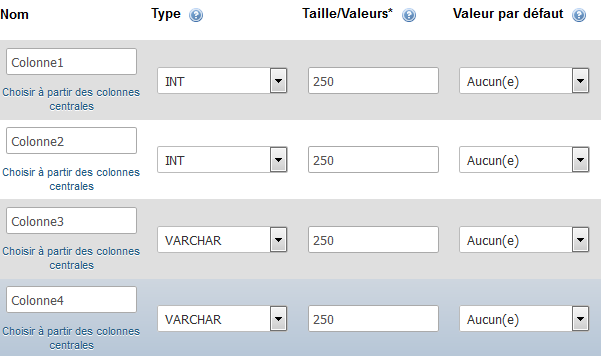


On nomme ensuite les colonnes et choisissons de quelles types elles sont, pou notre calendrier nous utiliserons le type:

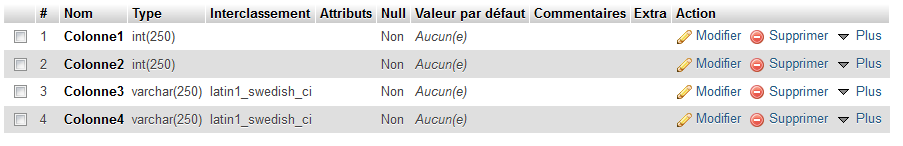
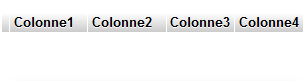
-INT (il est utilisé pour les nombres entiers);

-VARCHAR(sert à stocker un texte);

-LONGBLOB(stocke en binaire de grosse données, on l'utilise pour stocké nos photos);

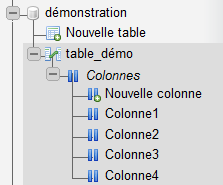


Nos colonnes sont créées (gauche), on peut y voir le type de valeur qu'elles stockent, leurs noms et leurs stockage maximale. On peut voir à droite le contenu des colonnes.



Zone où s'affiche les données stockées.

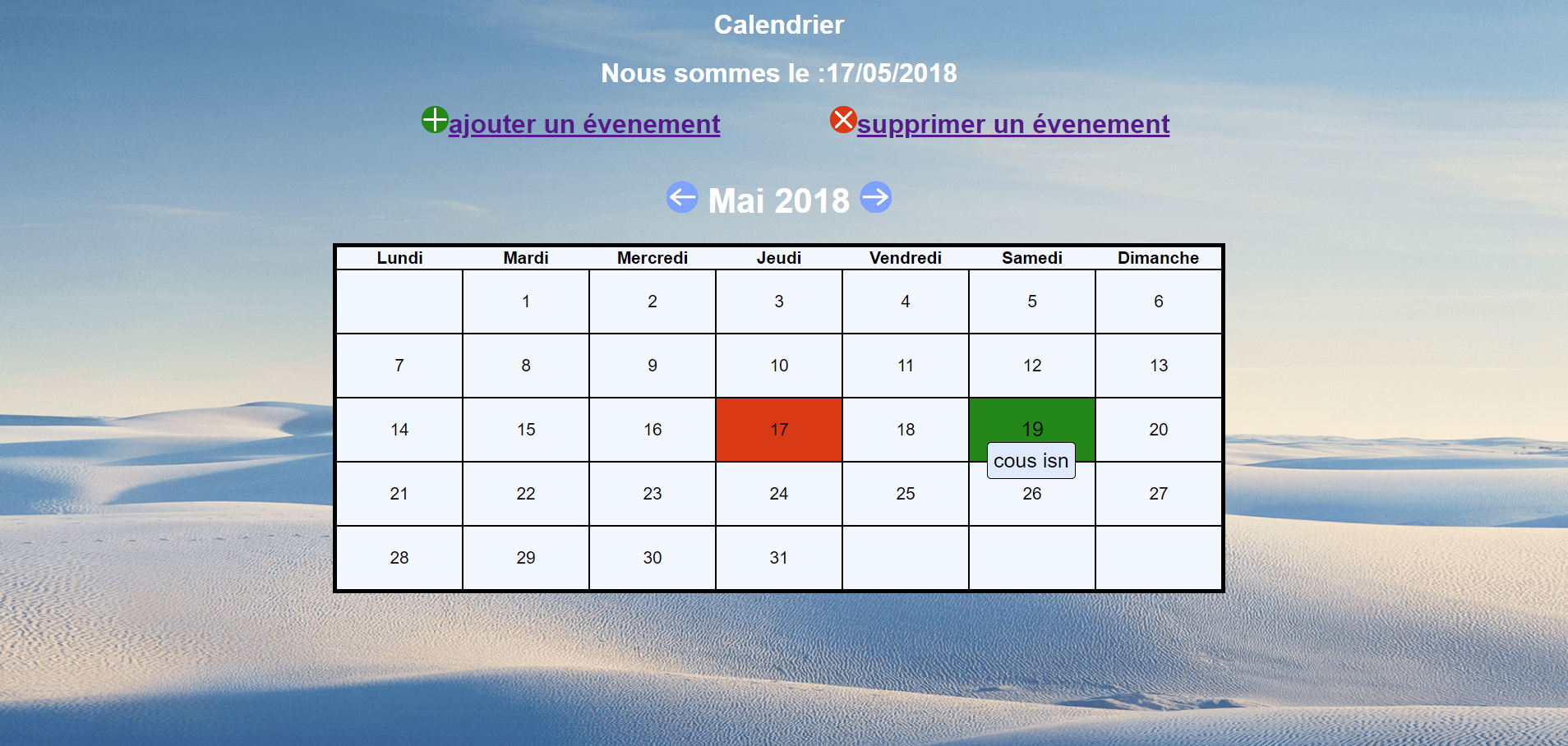
La base de données créée se présente sous cette forme:



Calendrier.css

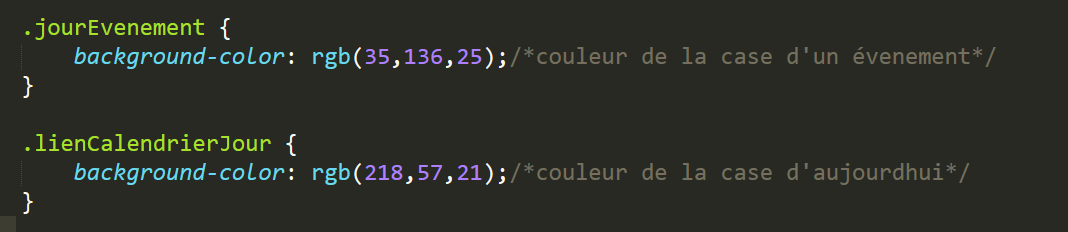
L’apparence

Le rendu du programme est important pour l’utilisateur, il lui permet d’interagir avec celui-ci. Dans la vraie vie un calendrier est simple d’utilisation, permettant la prise de notes rapide. Par conséquent un programme dédié à la réalisation d’un calendrier doit avoir une apparence simple et accessible par tout le monde. C’est pourquoi nous avons conçu pour notre calendrier une interface ergonomique, intuitive et sobre, le rendant à la fois élégant et pratique. Le programme comprend 4 pages.

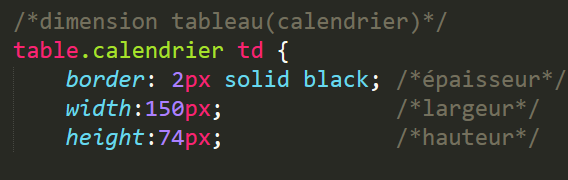


La page d’accueil : Sur celle-ci est affiché un tableau qui est le calendrier. Le jour actuel est de couleur rouge tandis que les jours assignés à un évènement apparaissent en verts. Les jours creux prennent une couleur s’approchant du blanc.

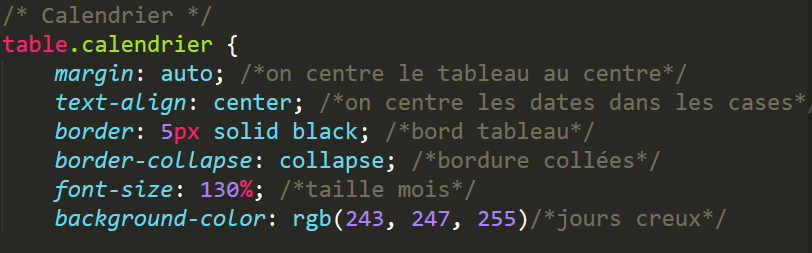
Couleurs cases :



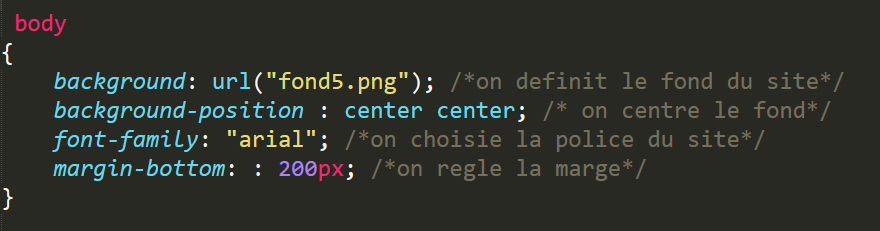
Les dimensions du tableau :



Les propriétés du tableau :

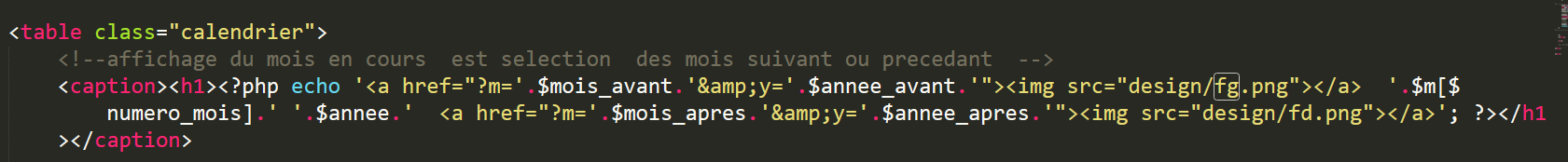


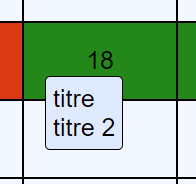
Définition de la police fond d’écran et de leur positionnement :



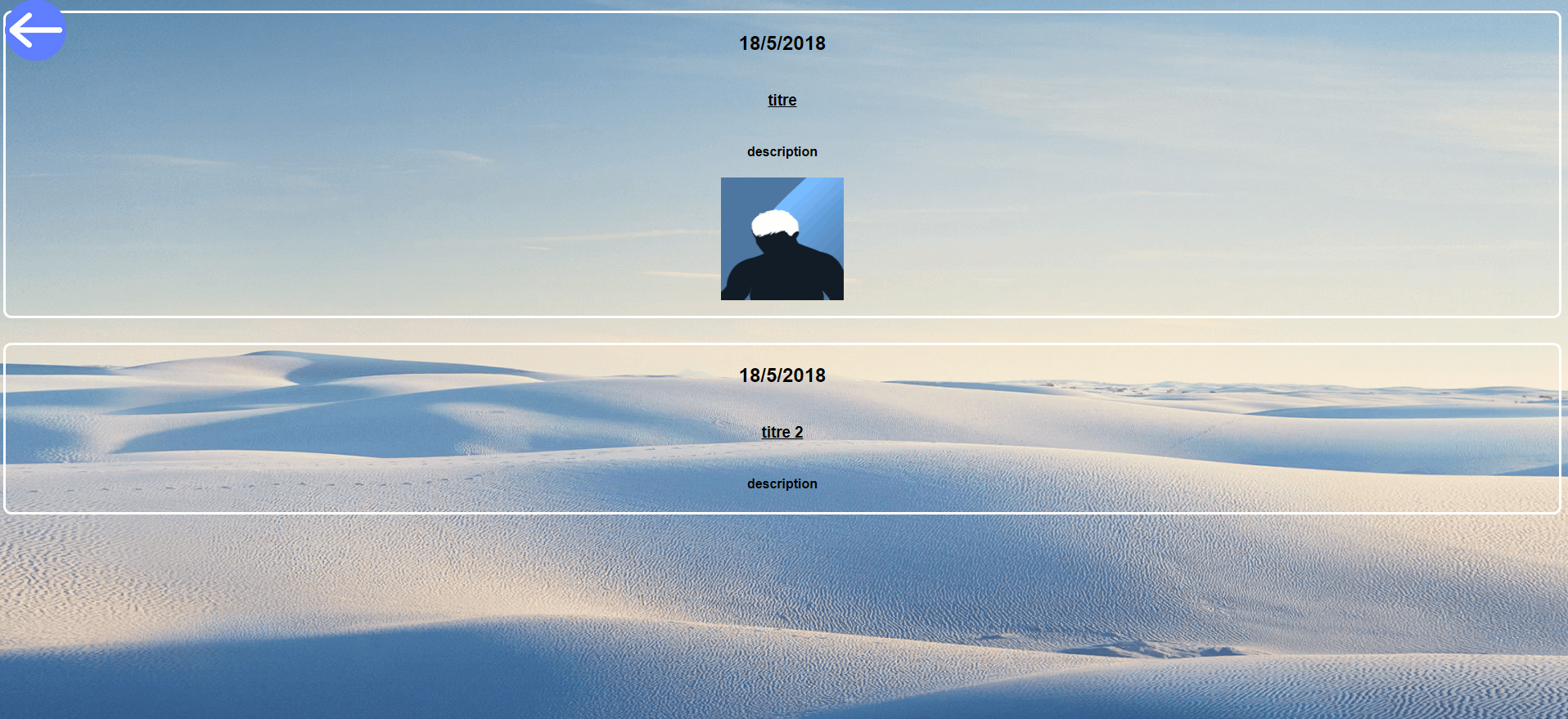
Au-dessus du tableau est positionné le mois et deux flèches à ses côtés. Celles-ci permettent à l’utilisateur de changer de mois en jouant sur les variables «mois avant» et « mois après»

Appel des images « fg » et « fd » pour changer de mois (php) :



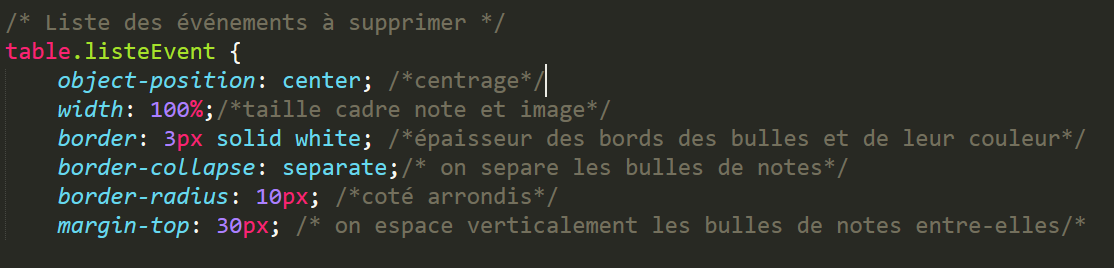


Il est possible de visualiser les évènements (du moins leur titre) en glissant sur une case. Cette action ouvrira une infobulle proche de la case du jour dans laquelle seront affichées les tâches. Puis en cliquant sur la case vous accédez à une fenêtre avec tous les évènements alignés les uns en dessous des autres.



Page de la liste des évènements : cette page affiche les évènements se passant dans la même journée. Chaque note est centrée dans une bulle avec une hiérarchie de ces éléments (dates, titre de l’évènement, contenu et photo). Ces bulles sont séparées par un espace facilitant la lecture à l’utilisateur.

Mise en forme des bulles et des notes :



En revenant sur la page d’accueil nous avons la possibilité de créer ou de supprimer un évènement.



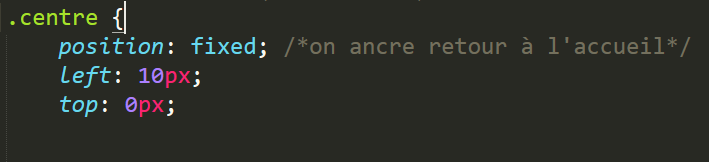
En cliquant sur le lien «ajouter un évènement » nous accédons à la fenêtre « ajouter un évènement ».



Page pour ajouter un évènement : cette page permet de créer un évènement et de lui assigner une photo. Cette page a une apparence sobre pour laisser place à l’essentiel. Elle est directement reliée au code « ajoutevent.php » mais elle prend l’apparence définie par « calendrier.css ».

Ensuite en revenant en arrière nous avons accès à la page « supprimer un évènement »voici à quoi elle ressemble :

Page de suppression des évènements : dans cette page l’utilisateur à la possibilité de supprimer manuellement des évènements. La mise en page est quasi la même que celle de la page « liste des évènements », car leurs mises en formes dépendent toutes les deux du fichier « calendrier.css ».Cependant nous avons l’ajout de la fonction de suppression. Celle-ci permet de supprimer un évènement. Du point de vue de la mise en page elle est exécutée lorsque que l’on clique sur l’icône rouge positionnée à gauche de la page.



La touche de navigation : cette touche « retour » est présente dans les fenêtres «supprimer un évènement », « ajouter un évènement », « liste des évènements ».Celle-ci nous permet de revenir à la page d’accueil lorsque l’on se situe dans une de ces fenêtres. Elle est ancrée dans la page, c’est-à-dire que si l’utilisateur fait défiler une page, la touche ne défilera pas avec le reste de la page. Elle restera fixe à l’écran, facilitant la navigation dans le calendrier.

Problèmes rencontré

- la suppression auto ne marchait que pour les mois de l’année en cours

- erreur quand l’utilisateur ne choisit pas d’image

- récupération des données date, évènement et image s’ils n’étaient pas dans la même table

- bug de langage de programmation ( ; : , «  ‘ ……)

- difficulté pour rentrer une image dans la base de données (LONGBLOB)

- affichages et créations des requêtes

- réalisation du calendrier en lui-même…

Conclusion :

Ce projet nous a permis de réaliser un programme faisant appel à différente forme de langages tels que le PHP, HTML, CSS et la gestion d’une base de données MySQL.

La cohésion d’équipe a été indispensable et enrichissante pour mener à bien ce projet et surmonter les problèmes rencontrés.

Aujourd’hui on se rend compte du travail à fournir pour réaliser une application du quotidien. Cependant nous sommes conscients de toutes les améliorations que nous pourrions apporter à ce programme dans l’avenir.

