# Programme EVALUATION

Fichiers de sortie : index.html et style.css

Fichier d’entrée : questions.qs

Exécution du code (enregistrer l’heure des réponses)

## Déroulement de l’évaluation :

### Partie 1 – Préparation a un projet de développement :

#### Ligne de commande + github

1. Créer un dossier évaluation\_20170103 à la racine du serveur
2. Initialiser le depot github
3. Alimenter le readme avec :

* Author :
* Title :
* Date :
* Language :
* Description : (empty)

1. Pousser « initialisation du projet » sur la branche master.
2. Créer un fichier Evaluation (dont l’extension dépend du langage choisi) qui correspondra au « main »
3. Pousser « mise en place environnement » sur la branche master
4. Créer une branche style
   1. Créer un dossier resultat
   2. Créer un fichier style.css
   3. Modifiez le readme pour rendre compte de la branche
   4. Pousser « ajout du style » sur la branche style
   5. Codez le css afin d’avoir a minima le rendu sur le fichier screenshot.jpg
   6. Pousser les changements
5. Créer une branche questionnaire
   1. Copier le fichier questions.qs à la racine du projet
   2. Modifiez le readme pour rendre compte de la branche
   3. Pousser « partie application » sur la branche questionnaire
   4. Coder l’application Evaluation.
   5. Pousser les changements avec des commits cohérents
6. Merger la branche questionnaire avec la branche master
7. Merger la branche style avec la branche master
8. Effectuer le questionnaire
9. Verifier le resultat à l’adresse : <http://localhost/évaluation_20170103/resultat>

### Partie 2 – Développement d’une application Questionnaire d’évaluation :

#### Choix libre du langage (php,java,python,ruby,c)

Fonctionnement du programme :

Le programme va lire le fichier questions.qs lignes par lignes :

##nom de la thématique : correspond au changement de thématique

#Question du bout du monde ? : correspond à la question à laquelle il faut répondre

Si c’est un QCM les lignes des choix sont sous cette forme :

-choix un

-choix deux

#Question suivante

Vous pouvez éditer le fichier pour voir sa forme.

Une fois récupérées, stockées dans un tableau et débarrassées des caractères inutiles

Le programme va poser les questions à l’utilisateur

Il va stocker les réponses dans une liste prévu à cet effet.

Pour chacune des réponses le programme va stocker l’heure (timestamp) de la réponse.

A la fin du Questionnaire, le programme va enregistrer les réponses dans un fichier html dont la forme est fournie plus bas.

Tips :

Utilisez un maximum de variable (vous débutez, n’hésitez pas)

Utiliser au maximum les fonctions pour rendre votre code plus clair

Utilisez les fonctionnalités des langages afin de vous faciliter la tâche (genre la fonction sort()).

Si vous avez opté pour un langage objet, profitez de la puissance que procure ce concept.

Exemple d’approche :

Vous pouvez séparer votre code en 3 grandes fonctionnalités :

* Partie récupération des questions
* Partie Evaluation
* Partie création du fichier de résultat.

Declaration des variables :

* String sortie = « <html>…</h2> » par exemple
* String debutTheme = « <article>…</h3>
* String finTheme = « </article>
* Etc. (je pense que vous voyez)
* List questions (pensez que certaines questions sont composées d’une liste de choix)
* List responses

Vous trouverez en annexe des exemples de codes dans différents langages pour lire et écrire dans un fichier correctement

Comme tout est à base de boucle, de conditions et de tableaux, normalement ça devrait le faire.

### Partie 3 – Couche présentation / webdesign :

#### Mettre en œuvre le css afin de correspondre à la maquette proposée.

Vous devez coder le fichier css afin de donner un peu « volume » au rendu Il est obligatoire de justifier ses choix (mise en place des classes, sélecteurs avancés, pseudo classes, etc.) en commentaire dans le code. Placer du responsive design n’est pas nécessaire car l’idée est surtout d’être sûre de savoir comment fonctionne le css et comment se comporte certaines balises avec certaines propriétés.

Programme Evaluations :

Fichiers de sortie :

* index.html :

<!--

index.html

Fichier créé par "nom" le "date"

-->

<html>

<head>

<meta http-equiv="content-type" content="text/html; charset=utf-8" />

<title>evaluations du 03 janvier 2017 – ville</title>

<link rel="stylesheet" type="text/css" href="style.css">

</head>

<body>

<h1>evaluations du 03 janvier 2017 – ville</h1>

<h2>nom de l'apprenant</h2>

<article>

<h3>Ecosystème</h3>

<section>

<h4><span>numero de question - </span>question ? <span>heure</span></h4>

<p> reponse sous forme de texte </p>

</section>

<section>

<h4><span>numero de question - </span>question (QCM)?<span>heure</span></h4>

<ol>

<li>reponse 1</li>

<li>reponse 2</li>

<li>reponse 3</li>

</ol>

<p> numero de la reponse </p>

</section>

</article>

<article>

<h3>HTML / CSS</h3>

<section>

<h4><span>numero de question - </span>question ?<span>heure</span></h4>

<p> reponse sous forme de texte </p>

</section>

<section>

<h4><span>numero de question - </span>question (QCM)?<span>heure</span></h4>

<ol>

<li>reponse 1</li>

<li>reponse 2</li>

<li>reponse 3</li>

</ol>

<p> numero de la reponse </p>

</section>

</article>

<article>

<h3>ALGO</h3>

<section>

<h4><span>numero de question - </span>question ?<span>heure</span></h4>

<p> reponse sous forme de texte </p>

</section>

<section>

<h4><span>numero de question - </span>question (QCM)?<span>heure</span></h4>

<ol>

<li>reponse 1</li>

<li>reponse 2</li>

<li>reponse 3</li>

</ol>

<p> numero de la reponse </p>

</section>

</article>

</body>

</html>

* style.css :

A minima vous devez coder le ficher css pour ce rendu ;)

* questions.qs : (exemple pour la forme du fichier)

##HTML/CSS

#quel est la valeur par défaut de la propriété display pour une balise DIV ?

-inline

-inline-block

-table

-block

#peut-on mettre une balise NAV dans une balise SECTION ? Pourquoi ?

##ECOSYSTEME

#par quelle expérience du quotidien peut-on expliquer le modèle client-serveur ? Expliquez ?