

Programme

HTML, les fondamentaux (1)

- Courte introduction au web
 - Connaître la notion de client-serveur.
 - Comprendre que le HTML, c'est côté client.
- Séparation des concepts
 - Comprendre que le langage HTML sert à porter le contenu.
- Le W3C
 - Comprendre qu'il y a eu plusieurs versions et une problématique de standardisation.
 - Savoir écrire le doctype html5.
- Présentation du navigateur : Chrome/Firefox
 - Savoir visualiser le code source de la page
 - Savoir ouvrir les outils de développeur et inspecter les éléments
- Présentation de l'éditeur : SublimeText
 - Savoir ouvrir un dossier dans SublimeText
 - Savoir les principaux raccourcis d'édition de contenu : sauvegarder un fichier, couper, copier, coller un texte.
 - Connaître le raccourcis SublimeText "Goto" (ctrl + p) pour trouver un fichier en tapant son nom.
- Syntaxe, balises et attributs
 - Maîtriser la syntaxe `<balise attribut="valeur"></balise>`
 - Connaître la structure : `html, head, <meta charset="utf-8">, title body`
 - Connaître les balises de contenu : `h1, h2..., p, strong, em, a, img, div, ul, ol, object, blockquote`
 - Connaître les balises de tableau : `table, thead, tbody, tfoot, tr, th, td`
 - Connaître les balises de formulaire : `form, label, input (text, checkbox, radio), select, option, textarea, button`
- Indentation et commentaires
 - Maîtriser parfaitement l'indentation sur son propre code.
 - Savoir appréhender un code HTML d'un autre développeur et réindenter un code mal indenté.
 - Savoir écrire un commentaire HTML.
 - Comprendre l'importance pour le débogage et la collaboration.
- Ressources et lectures conseillées
 - Savoir retrouver un listing de balises dans la documentation.
 - Savoir retrouver la sémantique d'une balise dans la documentation.
- Terminologies, définitions, jargon
 - Maîtriser le jargon de base : développement web, intégration web, langage, balise, attribut, navigateur, éditeur de texte, code, HTML
 - Balises fermantes et auto-fermantes
 - Connaître la notion de balise de type `block` et `inline`

Bonus

- Ajout de style via l'attribut, pour préparer le cours du lendemain.
 - Connaître les balises de structure : `link`, `script`
 - Connaître les balises de contenu : `dl`, `abbr`, `address`, `del`, `ins`, `iframe`
 - Connaître les balises de formulaire : `input` (`file`), `fieldset`, `legend`
-

Les CSS (2)

- Les CSS
 - Savoir ajouter un fichier CSS avec la balise `<link>`.
 - Connaître l'instruction `@import` pour permettre d'importer un CSS.
 - Comprendre que l'attribut `style` servira au JavaScript pour venir modifier le style.
- Séparation des concepts
 - Comprendre que le langage CSS sert à la présentation, l'apparence, et ne sert pas à afficher du contenu (HTML).
- Arborescence de fichiers, organisation, raccourcis clavier, bonnes pratiques
 - Savoir créer un dossier comprenant un dossier `html/`, `css/`, `img/` et lier les fichiers entre eux.
- Syntaxe, sélecteurs simples, propriétés
 - Maîtriser l'écriture d'une instruction simple : `element { propriete: valeur; }`.
 - Savoir cibler dans le HTML avec `#id` et `.class`.
- Indentation et commentaires.
 - Savoir bien présenter son fichier CSS avec indentation et commentaires.
- Mise en forme typographique avec les CSS.
 - Savoir utiliser `color`, avec `#RGB`, `#RRGGBB` et `rgb(RRR, VVV, BBB)`.
 - `font-size`, `font-family`, `font-weight`.
 - `text-transform`, `text-decoration`.
- Ajout d'images en CSS
 - Savoir utiliser `background-image` et maîtriser la syntaxe `url(...)`
 - Comprendre la différence entre un contenu `` et une décoration en CSS.
- `block` et `inline`
 - Maîtriser la notion de balise de type `block` et `inline`.
 - Comprendre que le comportement peut être changé avec `display` et `float`.
- Dimensions et marges.
 - Savoir utiliser `width`, `height`.
 - Savoir utiliser `margin-top`, `margin-right`, `margin-bottom`, `margin-left`, `padding-top`, `padding-right`, `padding-bottom`, `padding-left`.
 - Comprendre le fonctionnement des propriétés complexes : `margin: <top> <right> <bottom> <left>;`, `margin: <top> <left/right> <bottom>;`, `margin: <top/bottom> <left/right>;`.
- Positionnement en CSS, premiers pas
 - Savoir utiliser `float`.
 - Savoir utiliser `clear`.
 - Savoir présenter un contenu texte en deux colonnes avec `float`.
 - Comprendre la notion de flux.

- Ressources et lectures conseillées.

Bonus

- Introduction du positionnement avancée avec `position`.
-

HTML et CSS avancés (3)

- Reset et Normalize.
 - Comprendre la différence entre les deux concepts.
 - Savoir mettre en place un reset ou un normalize.
- Sélecteurs avancés
 - Connaître les sélecteurs `>`, `+`, `~`
 - Connaître les pseudo-sélecteurs `:hover`, `:focus`, `:active`, `:first-child`
- Priorité et poids des sélecteurs.
 - Savoir déterminer précisément quel sélecteur sera prioritaire
- Positionnement CSS, solutions avancées
 - Utilisation de la propriété `position`.
- Images : notions supplémentaires
 - Utilisation du `background-repeat`
 - Utilisation du `background-position`

Bonus

- Aborder la notion de sprite CSS
-

HTML5 et CSS3 (4)

- Les nouvelles balises sémantiques HTML5
 - Savoir utiliser `header`, `footer`, `nav`, `article`, `main`, `figure`, `aside`, `section`
- CSS3
 - Savoir utiliser `box-shadow`, `opacity`, `border-radius`
 - Aborder la propriété `transform`
 - Comprendre le concept des pseudos-éléments `::before` et `::after`
 - Comprendre et savoir utiliser la technique du clearfix
 - Comprendre et savoir utiliser la technique du `box-sizing: border-box`
- Adobe Extract
 - Savoir travailler sur un PSD avec Adobe Extract pour découper des images
 - Connaître les problématiques d'optimisations des images

Bonus

- Savoir ajouter un favicon avec `<link rel="shortcut icon">`

Atelier pratique HTML5/CSS3 : integration 1 (5)

- Fonts
 - Comprendre l'instruction `@font-face`
 - Savoir aller chercher et utiliser une police de Google Fonts.

Bonus

- Aborder la valeur `linear-gradient`
-

Atelier pratique HTML5/CSS3 : correction 1 (6)

Introduction à JavaScript, partie 1 (7)

- Courte introduction au développement web.
 - Comprendre la différence entre le HTML/CSS et le développement
 - Savoir que développement = calculs, logiques, traitements...
 - Comprendre que le JS qu'ils vont voir, c'est côté client.
- Les qualités requises du développeur
 - Savoir faire preuve de rigueur : respecter la syntaxe, faire attention au faute de frappe
 - Développer une capacité d'abstraction : se représenter ce qu'il se passe lorsque le code s'exécute
- Intro à JavaScript.
 - Savoir que JavaScript = nom plus commercial de ECMAScript
 - Savoir que le langage a été développé dans le but d'ajouter des interactions aux pages
- Séparation des concepts
 - Comprendre que le langage JS sert à réaliser des interactions, et ne doit pas porter de contenu (HTML) ou de présentation (CSS).
- Les variables.
 - Savoir assigner un valeur à une variable.
 - Savoir modifier le contenu d'une variable.
- Les instructions.
 - Comprendre que les lignes de code sont lues de haut en bas.
 - Chaque instruction doit se terminer par un `;` ou un retour à la ligne.
- Les types de données.
 - Comprendre qu'un langage de programmation possède plusieurs types de données.
 - Savoir utiliser les types de base : nombre (entier et flottant), chaîne de caractère, booléen.
 - Connaître les collections de base : tableau et objet.
- Opérations sur les données.

- Savoir réaliser une addition (+), soustraction (-), multiplication (*) et division (/) de nombres.
- Savoir réaliser une addition (+) de chaîne de caractère.
- Connaître le terme technique pour l'addition de chaîne : concaténation.
- Connaître le modulo (%) : reste de la division euclidienne.
- Savoir récupérer un item d'un tableau avec `tableau[index]`.
- Savoir récupérer une propriété d'un objet avec `objet[propriete]` et `objet.propriete`.
- La console.
 - Comprendre le système de console : interpréteur au sein du navigateur.
 - Savoir lancer une instruction depuis la console.
 - Savoir afficher une information dans la console depuis le code avec `console.log()`.
- Indentation et commentaires
 - Maîtriser parfaitement l'indentation sur son propre code.
 - Savoir judicieusement utiliser les blocs de commentaires et les commentaires en ligne.
 - Savoir appréhender un code JS d'un autre développeur et réindenter un code mal indenté.
 - Comprendre l'importance d'un code bien indenté pour la relecture du code et la gestion d'erreur de syntaxe.
- Les erreurs.
 - Comprendre le mécanisme d'erreur et savoir repérer une erreur dans la console.
 - Reconnaître les erreurs de syntaxe : "Uncaught SyntaxError"
 - Reconnaître l'erreur de référence, souvent liée à une faute de frappe : "Uncaught ReferenceError"
- Les fonctions.
 - Connaître la syntaxe d'une fonction : `function bonjour() {}` ou `bonjour = function() {}`
 - Comprendre la différence entre une fonction `bonjour` et son exécution `bonjour()`
 - Savoir retourner une valeur avec `return`
- Ressources et lectures conseillées.

Bonus

- Améliorer le debug : `console.info()`, `console.error()`.
 - Introduction à la logique.
 - Introduction aux boucles.
-

Introduction à JavaScript, partie 2 (8)

- Les opérateurs logiques.
 - Maîtriser la notion de booléen
 - Connaître les opérateurs de comparaison `>`, `<`, `>=`, `<=`, `==`, `!=`, `===`, `!==`
 - Connaître les opérateurs logique `&&`, `||`, `!`
- La comparaison
 - Savoir utiliser la syntaxe `if(condition) {} else {}`.
 - Savoir combiner les opérateurs logique et de comparaison pour obtenir la condition souhaitée

- Aborder les conditions imbriquées et `else if`
- Incrémentation et arithmétique.
 - Connaître les opérateurs d'affectation : `+=`, `-=`, `*=`, `/=`, `%=`
 - Connaître l'opérateur d'incrémentation (`++`) et de décrémentation (`--`).
 - Aborder la différence entre `i++` et `++i`
- Tableaux et objets
 - Savoir récupérer un item d'un tableau avec `tableau[index]`
 - Savoir rajouter un item avec `tableau.push()`
 - Savoir ressortir un item avec `tableau.pop()`
 - Savoir récupérer la valeur d'une clé d'un objet avec `objet.cle`
- Les boucles
 - Connaître la syntaxe et le fonctionnement de la boucle `for`
 - Connaître le fonctionnement de la boucle `while`.
 - Savoir parcourir un tableau avec `for` et `while`.
 - Aborder la boucle de `for-in`.
- La portée des variables
 - Comprendre la notion de variable locale et globale.
 - Savoir utiliser le mot-clé `var`.
- Structuration de code JS.
 - Savoir regrouper son code en fonction courtes.
 - Prendre l'habitude de commenter avec des blocs de commentaires chaque partie du fichier.
 - Prendre l'habitude de commenter avec des commentaires en ligne chaque instruction qui n'est pas compréhensible immédiatement.

Bonus

- `switch`.
 - `do-while`.
 - Améliorer le debug : `debugger`.
-

Introduction a JavaScript, partie 3 (9)

- Le DOM
 - Savoir utiliser la fonction `document.getElementById`
 - Connaître les fonctions `document.getElementsByClassName` et `document.getElementsByTagName`
- Manipulation des contenus.
 - Comprendre la relation entre attribut (HTML) et propriété d'objet (DOM).
 - Savoir récupérer ou définir un attribut.
 - Savoir modifier le texte d'un élément avec `.textContent`.
 - Savoir ajouter un enfant avec `.appendChild()`.
 - Savoir combiner ces manipulations : générer des éléments, les remplir de texte et les ajouter à la page.

- Manipulation des styles.
 - Savoir modifier l'attribut `class` avec `.className`.
 - Savoir modifier le style directement avec `.style`.
- Les événements.
 - Comprendre le système d'évènements : une fonction qui est exécutée lorsqu'une action survient.
 - Connaître les principaux évènements : `click`, `mouseenter`, `mouseleave`, `keydown`, `keyup`, `scroll`, `resize`, `change`, `submit`, `DOMContentLoaded`
- Les écouteurs d'évènements.
 - Aborder l'existence de vieilles syntaxes : `<element onClick="hello()">` et `element.onClick = hello`
 - Connaître la syntaxe de `.addEventListener()`
 - Savoir exécuter une action au clic sur un élément.
 - Savoir récupérer la valeur d'un champ de saisie au submit d'un formulaire.
 - Connaître la syntaxe de `.removeEventListener()`

Bonus

- Modifier les classes avec `.classList`
 - `document.querySelector()` et `document.querySelectorAll()`
 - Bubbling d'évènements
-

Introduction a JavaScript, partie 4 (10)

- Gestion du temps
 - Comprendre la syntaxe de `setTimeout()`.
 - Savoir exécuter une fonction au bout de X secondes.
 - Comprendre la syntaxe de `setInterval()`.
 - Savoir exécuter une fonction toutes les X secondes.
 - Comprendre la syntaxe de `clearInterval()`
 - Savoir arrêter un compte à rebours.

Bonus

- Découverte de jQuery.
-

Introduction a jQuery, partie 1 (11)

- Pourquoi utiliser des librairies ?
 - Comprendre l'importance de ne pas réinventer la roue.
 - Comprendre l'importance des librairies, framework et CMS dans le monde du développement et dans le marché du travail.

- Savoir que jQuery a été créé pour permettre une compatibilité entre les navigateurs.
- Savoir que jQuery permet une écriture plus rapide, plus intuitive et plus lisible.
- Intégrer jQuery, CDN.
 - Savoir télécharger jQuery en local, et l'intégrer via `<script>`.
 - Savoir intégrer jQuery depuis un CDN.
 - Comprendre les principaux intérêts d'utiliser un CDN en production : cache et temps de réponse.
 - Comprendre l'intérêt de ne pas utiliser de CDN en développement : pouvoir travailler hors connexion.
- Les sélecteurs.
 - Savoir réutiliser les sélecteurs CSS avec jQuery.
 - Savoir utiliser les sélecteurs propres à jQuery les plus utiles : `:visible`, `:hidden`, `:first`.
 - Comprendre que la différence entre objet jQuery et objet HTML, et que `$("#id")` est différent de `document.getElementById("id")`.
 - Comprendre qu'une collection jQuery est comme un tableau d'éléments HTML, avec des méthodes en plus.
- Manipulation du DOM, et méthodes de parcours.
 - Savoir utiliser les principales fonctions de parcours : `.find()`, `.children()`, `.next()`, `.prev()`, `.parent()`, `.eq()`
 - Savoir utiliser les principales fonctions de parcours : `.find()`, `.children()`, `.next()`, `.prev()`, `.parent()`
 - Savoir aller chercher dans la documentation jQuery pour rechercher la méthode appropriée.
- Les event listeners.
 - Utiliser la syntaxe raccourci du chargement du DOM : `$(function(){ ... })`
 - Savoir utiliser `.on()`
 - Savoir utiliser `.off()` et comprendre la simplicité par rapport à `.removeEventListener()`

Bonus

- Voir davantage de fonctions de parcours : `.nextAll()`, `.prevAll()`, `.siblings()`
 - Savoir utiliser `.one()`
-

Introduction à jQuery, partie 2 (12)

- Animations jQuery.
 - Savoir faire une animation simple avec `.animate(propriétés)`
 - Connaître les possibilités et options de `.animate(propriétés, options)`
 - Savoir exécuter une fonction après la fin d'une animation.
 - Savoir enchaîner les animations et comprendre le mécanisme de queue.
 - Connaître les fonctions d'animations : `.fadeIn()`, `.fadeOut()`, `.slideDown()`, `.slideUp()`.

Bonus

- Utilisation de la fonction `.delay()`
 - Utilisation de la fonction `.stop()`
-

jQuery ui et les extensions jQuery (13)

- jQuery UI.
 - Connaître les possibilités de jQuery UI.
 - Savoir télécharger jQuery UI, en sélectionnant uniquement les composants utiles.
 - Savoir mettre en place un datepicker.
 - Savoir mettre en place un draggable.
- Plugins courants
 - Savoir réaliser une popup avec FancyBox.
 - Savoir réaliser un carousel avec FlexSlider.

Bonus

- Savoir utiliser `.sortable()` pour classer les `` d'une liste.
 - Savoir utiliser `droppable`, et exécuter une action suite à un drag & drop.
-

Révision Rattrapage HTML / CSS / JS (14)

Révision Rattrapage HTML / CSS / JS (15)

Examen JavaScript et jQuery, révision (16)

PHP : Démarrage et syntaxe (17)

- Le processus de requête du client au serveur
 - Comprendre la distinction entre client et serveur
 - Comprendre les concepts de requête et de réponse
- Les outils nécessaires
 - Connaître les différentes options de plate-forme de développement (wamp, mamp, xampp, etc.)
 - Savoir installer une plate-forme de développement
 - Connaître quelques options d'éditeurs de texte et d'IDE
- Installation/Configuration de xampp

- Savoir retrouver l'emplacement de l'exécutable PHP
- Savoir retrouver le php.ini
- Les ressources disponibles
 - Connaître l'existence de la ressource essentielle, le manuel PHP
 - Connaître quelques options de tutoriels
- Les balises PHP
 - Savoir ouvrir et fermer la balise `<?php ?>`
- Les variables, les types de données
 - Savoir créer des chaînes, des entiers, des décimales, des booléens et des tableaux
 - Savoir assigner des valeurs aux variables
 - Savoir modifier la valeur d'une variable
 - Savoir afficher la valeur d'une variable
 - Savoir distinguer les types de données par leur valeur
 - Comprendre la raison d'être de chacun des types de données
- Les instructions
 - Comprendre la notion d'instruction
 - Savoir terminer une instruction par un ;
- Le débogage, les erreurs
 - Savoir utiliser les fonctions `print_r()` et `die()`
 - Connaître les différents niveaux d'erreurs
 - Se familiariser avec les messages d'erreurs les plus courants
- Les conditions
 - Savoir écrire un `if` `elseif` `else`
- Les opérateurs, la comparaison
 - Savoir utiliser les principaux opérateurs de comparaison
- La concaténation
 - Savoir utiliser l'opérateur de concaténation `.`
 - Savoir utiliser l'opérateur combiné `.=`
 - Comprendre la distinction entre apostrophes et guillemets
- Arithmétique et incrémentation
 - Savoir utiliser les opérateurs de base `(+/-/*)`
 - Savoir incrémenter et décrémenter une variable avec `++` et `--`
 - Connaître les opérateurs combinés `+=`, `-=`, etc.

Bonus

- Historique de PHP
 - PHP vs les autres langages serveur
-

PHP : boucles et fonctions utiles (18)

- Les boucles

- Connaître l'existence des différents boucle : for, foreach, while, do while
 - Savoir utiliser une boucle for pour afficher x nombre d'éléments
 - Savoir utiliser une boucle foreach pour parcourir un tableau
 - Les fonctions internes de base
 - Savoir utiliser quelques fonctions fréquemment utilisées en PHP : date(), empty(), isset(), unset(), print_r()...
 - Étude du manuel PHP
 - Comprendre la structure d'une page type sur une fonction
 - Savoir comment chercher une fonction
 - Gestion des dates
 - Savoir utiliser la fonction date() pour obtenir la date actuelle sous différent format
 - Comprendre le concept de timestamp Unix
 - Connaître les fonctions time() et strtotime()
-

PHP : Les tableaux complexes et les formulaires (19)

- Les tableaux multidimensionnels
 - Comprendre le concept de tableaux imbriqués
 - Savoir créer un tableau multidimensionnel
 - Être capable d'accéder à une valeur imbriquée (pour la lire, la modifier...)
 - Savoir parcourir un tableau multidimensionnel avec une boucle foreach
- Les formulaires en HTML
 - Connaître les attributs essentiels pour la réception des données (action, method, name, etc.)
 - Comprendre comment sont envoyées les valeurs des différents types de champs
- Les variables superglobales : \$_GET et \$_POST
 - Comprendre dans quels cas utiliser la méthode GET vs POST
 - Comprendre comment les valeurs du formulaire sont transportées jusque dans ces tableaux
 - Savoir accéder aux valeurs de \$_GET ou \$_POST
- Stratégie de validation de données
 - Savoir tester si un formulaire est soumis
 - Savoir tester si un champs requis est vide
 - Savoir tester si une longueur de chaîne est comprise entre un minimum et un maximum
 - Savoir afficher des messages d'erreurs auprès du champ concerné
 - Savoir détecter si un formulaire est globalement valide
- Fonctions sur les chaînes
 - Comprendre comment utiliser des fonctions telles que : strlen(), strtoupper(), trim(), strip_tags(), etc.

Bonus

- Introduction à la sécurité et à l'intégrité des données

PHP : Dynamiser un site HTML (20)

- Structures de contrôle d'inclusion (include() et autres)
 - Savoir faire une inclusion d'une partie d'un fichier html
 - Comprendre la distinction entre include(), require() et les version _once()
- Affichage de balises HTML avec les boucles
 - Être en mesure d'afficher les valeurs d'un tableau multidimensionnel, en concaténation avec des balises HTML
- Les fonctions utilisateurs
 - Savoir créer en autonomie une fonction sans argument ni valeur de retour
 - Comprendre le concept d'argument
 - Connaître l'existence des valeurs de retour
- La portée des variables
 - Comprendre qu'une variable globale n'est pas accessible dans une fonction et vice-versa
 - Connaître l'existence du mot-clef global
 - Comprendre le lien entre la portée des variables, et les arguments des fonctions et des valeurs de retour

Bonus

- Fonctions internes sur les tableaux
-

Hébergements et domaines (21)

- Décryptage des offres d'hébergements mutualisés
 - Connaître quelques options d'hébergement de qualité
 - Savoir quels critères analyser lors du choix
 - Comprendre les limites des hébergements mutualisés
- FTP
 - Connaître quelques options de clients FTP
 - Savoir configurer le logiciel retenu
 - Être en mesure de se connecter à un serveur
 - Être en mesure de charger/télécharger des fichiers
 - Savoir comment changer les droits d'accès sur les fichiers et répertoires
- Gestion de nom de domaines
 - Connaître quelques options de registrar
 - Comprendre le processus d'enregistrement d'un nom de domaine
 - Comprendre la structure d'un nom de domaine et les différentes extensions
- DNS et réseaux
 - Connaître la raison d'être des adresses IP
 - Être initié au processus de requête aux serveurs DNS

- Comprendre les principaux enregistrements DNS
 - Gestion d'adresses mail
 - Connaître les principes de création et de gestion des adresses mail
-

Conception de bases de données (22)

- Utilité des bases de données
 - Comprendre l'importance des bases de données
 - Connaître plusieurs exemples d'utilisation de base de données sur le web
 - Les différents SGBD
 - Être initié aux concepts de SGBDR et de bases NoSQL
 - Être familiarisé avec les grands noms des SGBDR / NoSQL
 - Connaître l'état de MySQL aujourd'hui
 - Présentation de SQL, MySQL et PHPMyAdmin
 - Comprendre la distinction et l'interaction entre le langage, le SGBDR et le GUI
 - Savoir comment accéder à PHPMyAdmin
 - Les tables
 - Comprendre la fonction des tables
 - Savoir comment créer une table par PHPMyAdmin
 - Les types de données
 - Connaître les types de données fréquemment rencontrés
 - Être en mesure de choisir le type de données le plus approprié
 - Comprendre l'implication sur la performance
 - Les utilisateurs
 - Connaître le concept d'utilisateur et de droits sous MySQL
 - Être capable de créer / configurer des utilisateurs
 - Être sensibilisé aux règles de sécurité de base
 - Les relations
 - Être initié au concept de relations entre tables
 - Clefs primaires
 - Connaître les caractéristiques des clefs primaires
 - Comprendre l'importance des clefs primaires
 - Importer et exporter des données
 - Savoir importer et exporter des base de données de manière autonome
 - Connaître quelques options d'export utile sous PHPMyAdmin
-

SQL, et introduction à PDO (23)

- SQL
 - CRUD de base en SQL

- Connaître les règles syntaxique de base de SQL
 - Être initié aux bonnes pratiques SQL
 - Être en mesure d'exécuter des requêtes INSERT, DELETE, UPDATE et SELECT très simples
 - Comprendre la clause WHERE
 - Clauses et fonctions SQL de base
 - Savoir utiliser les opérateurs AND et OR
 - Savoir utiliser le LIMIT (à un paramètre)
 - Savoir utiliser le ORDER BY
 - Savoir utiliser la fonction NOW()
 - Les erreurs SQL
 - Être initié aux messages d'erreur courants
 - PDO
 - Connexion à la base de données
 - Comprendre les données du DSN
 - Être en mesure d'expliquer un code de "connexion à la base"
 - Les classes PDO et PDOStatement
 - Être prévenu de l'existence de 2 objets distincts
 - Exécuter des requêtes CUD
 - Savoir exécuter de manière dirigée des requêtes INSERT, UPDATE et DELETE
-

PHP et MySQL avec PDO (24)

- Exécuter des SELECT : les différents fetch
 - Être initié au fetchAll()
 - Être initié au fetch()
 - Être initié au fetchColumn()
 - Comprendre le type de données reçues en fonction du fetch utilisé
 - Les injections SQL
 - Comprendre le scénario d'attaque et les faiblesses possibles
 - Comprendre comment utiliser les paramètres nommés pour se protéger
 - Les attaques XSS
 - Comprendre le scénario d'attaque des XSS persistantes et non-persistantes
 - Connaître les fonctions utiles pour se protéger
 - Les requêtes préparées, paramètres
 - Savoir réaliser des requêtes préparées avec paramètres en semie-autonomie
-

Administration système Linux (25)

- Ligne de commande
 - Introduction à l'administration système Linux

- Décryptage des offres d'hébergements VPS et dédiés
 - Initiation à la ligne de commande
 - Chemins relatifs et absolus
 - Permissions
 - Administration système
 - Utilisation de "root"
 - Créer et modifier des fichiers texte en ligne de commande
 - Commandes utiles
 - Installation de services sous distributions basées sur Debian
 - Transfert de fichiers de serveur à serveur
-

Construire un site dynamique, partie 1 (26)

- Structure et organisation d'un site dynamique
 - Comprendre une stratégie simple d'organisation des répertoires
 - Connaître le concept DRY
 - Comprendre l'utilité des structures de contrôle d'inclusion pour éviter les répétitions
 - Comprendre l'utilité des fonctions pour éviter les répétitions
 - Planification
 - Comprendre l'importance de planifier l'organisation des fichiers
 - Savoir tracer les grandes lignes du parcours utilisateurs
 - Stratégie d'utilisation des paramètres d'URI
 - Comprendre comment les paramètres d'URI permettent de rendre une même page dynamique
 - Savoir utiliser un paramètre d'URI simple, du type `?id=125`
 - MySQL : les différents types de relations
 - Comprendre la différence entre les relations 1:1, 1:n et n:n
 - Être capable de choisir la bonne relation entre 2 tables simples
-

Construire un site dynamique, partie 2 (27)

- Composer
 - Savoir installer composer sur Windows
 - Savoir utiliser et comprendre les commandes `search`, `require`, `install` et `update`
 - Comprendre le fichier `composer.json`
 - Connaître l'existence du fichier `composer.lock`
 - Savoir intégrer le fichier `autoload` de composer
- Stratégies de système de pagination
 - Comprendre l'utilité du paramètre d'URI `?page=2`
 - Savoir utiliser les "limit" et "offset" MySQL, de manière dynamique

- Savoir comment afficher le numéro des résultats actuels (ie. articles #21 à #40 sur 1540 articles)
 - Savoir comment créer des liens vers la page suivante et précédente
-

Construire un site dynamique, partie 3 (28)

- Stratégies de système de filtre/recherche
 - Comprendre pourquoi le formulaire est en GET
 - Comprendre comment utiliser les paramètres d'URI, de type `?q=motclef&categ=3`
 - Comprendre comment rendre la requête SQL dynamique, de manière sécuritaire
 - Savoir faire une recherche SQL par mot-clef simple avec `LIKE '%motclef%'`
 - MySQL : fonctions d'agrégation
 - Comprendre le comportement de base des fonctions d'agrégation
 - Savoir faire des requêtes avec des fonctions d'agrégation simples telles que `COUNT()`, `SUM()`, `MAX()`, etc.
 - Comprendre le fonctionnement du `GROUP BY`
-

Révision Rattrapage HTML / CSS / JS (29)

Responsive Web Design (30)

- Le Responsive Design
 - Comprendre la notion de responsive : on s'adapte à l'utilisateur afin d'augmenter son expérience.
 - Comprendre les problématiques du responsive : ne se limite pas au design, mais aussi à l'ergonomie et de marketing (mise en avant d'un numéro de téléphone plutôt qu'un formulaire de contact par exemple).
 - Comprendre l'intérêt du responsive : le nombre d'appareils est très grand et encore en augmentation, et qu'il est rarement possible de faire une application native par appareil.
 - Comprendre le concept de Mobile First.
- S'adapter à l'utilisateur...
 - ...en récupérant son navigateur : connaître le principe d'user-agent.
 - ...en observant les fonctionnalités disponibles : aborder la librairie Modernizr.
 - ...en calculant la taille d'écran : savoir récupérer la largeur en js `$(window).width()`
- CSS Media Queries
 - Séparation des concepts : comprendre que ce n'est pas à JS de faire du responsive, mais à CSS.
 - Maîtriser la syntaxe `@media (conditions) { sélecteurs... }`?
 - Savoir utiliser et combien `min-width` et `max-width`.
 - Comprendre le principe de breakpoints et l'idée de réécrire le CSS pour une taille d'écran.
 - Type de média : `screen`, `print`...

- Savoir utiliser les unités `em` et `rem`.
- Savoir réaliser la version mobile d'une intégration desktop.
- Guidelines responsive
 - Savoir qu'il faut éviter les contenus Flash.
 - Savoir utiliser des polices lisibles, avec des tailles appropriées.
 - Savoir qu'il ne faut pas coller deux liens, car un pouce est moins précis qu'une souris.
 - Savoir configurer la fenêtre d'affichage avec la balise `<meta name="viewport">`.
 - Connaître l'outil PageSpeed Insights de Google pour tester son site.

Bonus

- Aborder la propriété `flexbox`.
-

Atelier pratique HTML5/CSS3 : integration 2 (31)

Bonus

- Aborder la propriété `transition`.
 - Aborder la propriété `animation`.
-

Examen PHP/SQL et révision (32)

Introduction à AJAX avec jQuery, partie 1 (33)

- AJAX
 - Aborder la notion d'`XMLHttpRequest` et d'AJAX sans jQuery.
 - Comprendre le principe d'envoyer une requête sans recharger la page.
 - Révision : comprendre la différence entre les requêtes HTTP `GET` et `POST`
- AJAX en jQuery
 - Savoir réaliser une requête simple avec `$.ajax({url: ...})`
 - Savoir exécuter une action après une requête avec `complete`
 - Savoir exécuter une action en fonction du résultat avec `success` et `error`
 - Savoir récupérer le retour du serveur avec `success: function(retour) { ... }`
 - Savoir envoyer des données au serveur avec `data`
 - Savoir faire une requête POST avec `type: "POST"`
 - Savoir soumettre un formulaire en exécutant une requête ajax dans le handler du submit.
 - Savoir utiliser la fonction `.serialize()` pour récupérer les données d'un formulaire

Bonus

- Aborder l'option `dataType`.

Introduction à AJAX avec jQuery, partie 2 (34)

- Accessibilité et ergonomie
 - Comprendre l'intérêt d'AJAX pour l'ergonomie : recharger une partie de la page sans tout recharger.
 - Comprendre le problème d'accessibilité du contenu, si tout le contenu n'est pas présent en HTML car il est chargé avec AJAX.
 - Comprendre le problème d'un site 100% ajax pour partager un lien, ouvrir un lien dans un nouvel onglet, utiliser les bouton suivant / précédant.
 - Comprendre le problème de cohérence d'un client asynchrone : quand deux onglets sont ouverts sur le même site AJAX.
 - Aborder l'utilisation d'un hash afin d'avoir des URLs différentes au sein d'un site 100% ajax : `exemple.com/#page1`, `exemple.com/#page2`
 - Comprendre les conséquences pour le SEO si tout le contenu n'est pas accessible en HTML.
 - Aborder rapidement l'ajax crawling de Google : `exemple.com/#!page1` sera référencé avec le contenu du `exemple.com/?_escaped_fragment_=page1`
- AJAX en jQuery
 - Passer des données PHP directement dans un objet javascript, avec `dataType: "json"` en JS et `json_encode()` en PHP.
 - Connaître la fonction `.ajaxSetup()` afin de simplifier son code.
 - Aborder les événements liés à AJAX : `$(document).on("ajaxStart",function() { ... })`.

Bonus

- Modifier le hash avec `location.hash`
 - Aborder la notion de `history.pushState`
-

GitHub.com / Système de fichiers, PHP7 (35)

- GitHub.com
 - Les avantages de versionner son code
 - Réaliser les dangers de ne pas versionner
 - Comprendre les problématiques de partage de codes en équipe
 - Comprendre l'intérêt de retourner dans le passé
 - Comprendre l'intérêt de pouvoir retracer l'évolution d'un projet
 - Exploration du site `github.com`
 - Se créer un compte github
 - Savoir comment faire des recherches de projets ou d'utilisateurs
 - Savoir naviguer dans un dépôt
 - Savoir télécharger un `.zip`
 - Savoir créer un nouveau dépôt

- Connaître l'existence des différentes sections du dépôt
- GitHub Desktop
- Installation et configuration
 - Savoir installer et configurer de manière autonome le logiciel
- Utilisation de base
 - Savoir créer un nouveau dépôt
 - Savoir cloner un dépôt existant sur GitHub
 - Savoir faire un commit, avec un message suivant les bonnes pratiques
 - Comprendre ce qu'est un bon commit
 - Savoir consulter l'historique (log)
 - Savoir lire les modifications apportées à un fichier (diff)
 - Savoir synchroniser un dépôt local avec un dépôt distant sur GitHub (pull / push)
- Workflow simple en équipe
 - Savoir travailler dans un dépôt distant commun (en tant que *collaborators*)
 - Savoir détecter et régler un conflit
- Système de fichiers en PHP
 - CRUD sur des fichiers
 - Savoir comment créer un fichier avec `file_put_contents()`
 - Savoir comment lire un fichier avec `file_get_contents()`
 - Savoir comment *ouvrir* un fichier avec `fopen()`
 - Comprendre les différents mode d'ouverture de fichier
 - Savoir comment écrire dans un fichier avec `fwrite()`
 - Savoir *fermer* un fichier avec `fclose()`
 - Savoir comment supprimer un fichier
 - Fonctions internes utiles
 - Être initié à `file_exists()`, `chmod()`, `is_dir()`, `is_file()`, `copy()`, `filemtime()`, etc.
- PHP7
 - Performances
 - Être initié à quelques benchmarks
 - Nouvelles fonctionnalités
 - Connaître l'existence de l'opérateur spaceship `<=>`
 - Connaître l'existence de l'opérateur `??`
 - Connaître les nouveaux typages
 - Être initié à la gestion des erreurs
 - Fonctions dépréciées
 - Savoir la disparition des fonctions `mysql_*`
 - Savoir la disparition des fonctions `ereg_*`

Bonus

- Les systèmes de gestion centralisé vs décentralisé
- Les logs

Git en ligne de commande / Upload de fichiers (36)

- Git
 - Les commandes essentielles de Git
 - Savoir utiliser de manière assez autonome : `status`, `log`, `add`, `commit -m`
 - Savoir lire l'aide de git
 - Savoir utiliser : `push` et `pull`
 - Savoir lire les messages d'erreur git
 - Connaître `reset --hard`
 - Les branches
 - Comprendre les avantages des branches basiques
 - Savoir créer une branche
 - Savoir changer de branche
 - Être initié à `merge`
- Upload de fichiers en PHP
 - Les particularités du formulaire
 - Savoir qu'il faut utiliser l'attribut `enctype`
 - Connaître les particularités du champ `file`
 - Variable superglobale `$_FILES`
 - Comprendre toutes les informations présentes dans le tableau
 - Sécurité et attaques possibles
 - Connaître un moyen fiable de tester le type de fichier chargé, type `Fileinfo`
 - Connaître un moyen de tester la taille du fichier
 - Savoir se protéger des attaques XSS sur le nom du fichier
 - Savoir comment sauvegarder sécuritairement le fichier sur le serveur

Bonus

- Études de différents workflow Git
 - Manipulation d'images en PHP
-

Authentification et autorisation, partie 1 (37)

- Inscription de l'utilisateur
 - Savoir réaliser les validations nécessaires (email déjà présent, mot de passe sécuritaire, etc.)
- Hachage de mot de passe, salt (PHP >= 5.5)
 - Comprendre les raisons du hachage, les risques de sécurité
 - Savoir utiliser la fonction `password_hash()`
 - Être initié au concept de "salt"
- Authentification de l'utilisateur
 - Savoir récupérer un utilisateur depuis la base de données

- Savoir utiliser la fonction password_verify()
 - Les sessions
 - Comprendre l'utilité générale des sessions
 - Comprendre qu'il faut utiliser session_start() partout
 - Comprendre comment manipuler la variable \$_SESSION
 - Savoir comment utiliser \$_SESSION pour authentifier un utilisateur
 - Les sessions et la sécurité
 - Connaître l'existence et l'utilité du cookie de session
 - Être initié au concept d'usurpation de session
-

Authentification et autorisation, partie 2 (38)

- Système d'oubli du mot de passe
 - Connaître un procédé sécuritaire de récupération d'un nouveau mot de passe
 - Être initié au concept de jeton (token) présent dans l'URL
 - Être capable de réaliser le système en suivant des étapes largement documentées
- Envoi d'email en PHP
 - Être initié aux enjeux des envois d'emails (limites, bonnes pratiques, options possibles)
 - Être en mesure d'envoyer un email avec une classe telle que PHPMailer et un serveur SMTP distant
 - Savoir créer un corps de message au format HTML
- Autorisation
 - Être en mesure d'utiliser les données disponibles sur l'utilisateur connecté pour autoriser l'accès aux ressources
 - Comprendre le concept de "rôle" d'un utilisateur
 - Connaissance de l'erreur 403

Bonus

- Système de sessions persistantes ("remember me")
-

Projet en équipe, partie 1 (39)

- Planification technique d'un projet
 - Comprendre l'importance de la planification d'un projet
 - Savoir analyser les besoins
 - Connaître les principaux composants des cahiers des charges
 - Connaître des techniques d'estimation du temps de travail
 - Connaître les grandes lignes des modèles conceptuel des données
 - Organisation du code
 - Division d'un projet en tâches

- Méthodologie de travail
 - Connaître des techniques de travail en équipe
 - Savoir comment répartir efficacement le travail
 - Connaître les pièges et avantages du travail en équipe
 - Connaître les outils disponibles (versionnage, communication, partage de fichiers, etc.)
-

Projet en équipe, partie 2 (40)

- Utilisation d'API
 - Comprendre la raison d'être des API et des webservices
 - Connaître l'existence des webservices les plus populaires
 - Savoir réaliser une requête à une API REST en PHP (sans CURL)
 - Savoir utiliser les données reçues
-

Projet en équipe, partie 3 (41)

- MySQL
 - Les jointures
 - Savoir faire une jointure simple entre 2 tables
 - Être en mesure d'expliquer le code SQL d'une jointure plus complexe
 - Comprendre les différents type de jointures MySQL
 - Les sous-requêtes
 - Comprendre l'utilité des sous-requêtes
 - Savoir réaliser une sous-requête simple
-

Projet en équipe, partie 4 (42)

- Les expressions rationnelles
 - Comprendre les délimiteurs, les quantifieurs, les classes de caractères et certains caractères spéciaux
 - Savoir écrire une expression rationnelle simple
 - Savoir tester ou remplacer l'expression avec `preg_match()` et `preg_replace()`
-

WordPress pour les webmasters (43)

- WordPress, un moteur de blog, mais pas que
 - Être initié au positionnement actuel de WordPress : popularité, réputation, etc.
 - Être initié au concept de CMS

- Comprendre dans quels cas le choix technologique de WordPress est pertinent
 - Le Codex, les ressources
 - Connaître l'existence de la documentation WordPress
 - Être initié à sa structure, à sa lecture
 - Connaître quelques tutoriels et références en ligne
 - Installation en local
 - Savoir installer de manière autonome un WordPress en local
 - Exploration du back-office
 - Connaître les différents types de contenus par défaut (posts, pages, media, comments)
 - Savoir créer et gérer des contenus
 - Connaître les configurations et réglages de base
 - Savoir gérer les utilisateurs
 - Exploration du front-office
 - Comprendre comment s'articulent et se présentent les différents types de contenus
 - Être initié à la barre d'administration
 - Les menus
 - Savoir gérer les menus de navigation
 - Les médias
 - Comprendre la gestion des médias dans la bibliothèque WordPress
 - Savoir créer des galeries
 - Les extensions
 - Savoir comment installer une extension
 - Savoir comment bien choisir une extension fiable
 - Connaître les dangers et limites des extensions
 - Connaître quelques extensions incontournables
 - Les thèmes (existants)
 - Savoir comment installer et activer un thème
 - Connaître quelques ressources de thèmes gratuits et payants
 - Être initié aux différentes options de customisation des thèmes
-

WordPress pour les intégrateurs (44)

- Les thèmes
 - Comprendre la structure des fichiers d'un thème existant
 - Connaître l'arborescence des fichiers (wp hierarchy.com)
 - Être initié aux contenus des fichiers (functions.php, style.css, index.php, etc.)
 - Hacker un thème avec les thèmes enfants
 - Comprendre l'utilité des thèmes enfants
 - Savoir comment créer un thème enfant simple
 - Créer un thème avec les thèmes blancs
 - Comprendre l'utilité des thèmes blancs (html5blank, bones, etc.)
 - Savoir modifier un thème blanc (directement dans le code)

- Intégration HTML/CSS sous WordPress
 - Savoir répartir le html dans les fichiers de thèmes
 - Connaître les principaux *template tags*
-

WordPress pour les développeurs (45)

- Les types de contenus personnalisés
 - Savoir ajouter et afficher un type de contenu personnalisé
 - Les champs personnalisés
 - Savoir ajouter et afficher des champs personnalisés
 - Les requêtes à la base de données
 - Savoir comment réaliser une requête personnalisée à la base de données
 - Introduction aux extensions WordPress
 - Introduction aux hooks, actions et shortcodes
 - Être en mesure de comprendre le code source d'une extension simple
 - Savoir à quel moment la création d'une extension est requise
 - Spécificités de la mise en ligne d'un site WordPress
 - Connaître la procédure et les pièges d'une mise en ligne sur un hébergement mutualisé
-

SEO – principes et bonnes pratiques (46)

- Les enjeux du SEO
 - Se concentrer sur l'augmentation du CA, et non sur les positions.
 - Comprendre le fonctionnement des moteurs de recherches.
 - Comprendre les problématiques d'audit et de suivi de conversion.
- Techniques d'optimisation on-site
 - Savoir rédiger des métas (title, description, url) optimisés
 - Savoir créer un code HTML sémantique
 - Connaître les critères de qualité d'un contenu web
 - Connaître les astuces de linkbaiting pour rédiger du contenu qui attire des liens
 - Savoir construire l'architecture et la hiérarchisation des contenus sur un site web
 - Savoir optimiser le linking interne
 - Connaître les notions d'Open-graph sociaux
 - Connaître l'algorithme Panda pour éviter les sanctions liés au contenu
- Techniques d'optimisation off-site
 - Connaître l'attribut nofollow
 - Comprendre la démarche de linkbuilding
 - Comprendre la notion de réputation / autorité
 - Obtenir de la visibilité via les réseaux sociaux
 - Trucs et astuces pour créer du lien : commentaires, annuaires, guest blog...

- Connaître l'algorithme Penguin pour éviter les sanctions liés aux mauvais liens
- Pratique
 - Savoir sélectionner ses mots clés
 - Connaître le concept de longue traîne
 - Découverte des outils Google : Analytics, Search Console, Keyword Planner...
 - Découverte d'outils d'analyse : Moz (OSE), Ahrefs, MajesticSEO...

Bonus

- Techniques d'optimisation on-site
 - Savoir utiliser les microformats pour bénéficier des rich snippets
 - Techniques d'optimisation off-site
 - Relations presse : contacter blogueurs ou journalistes pour obtenir un lien
 - Pratique
 - Connaître des outils de suivi de position : Positeo, SEO Soft, Seurank...
-

Les frameworks CSS (47)

- Les Frameworks CSS
 - Comprendre l'idée de ne pas réinventer la roue : un jeu de classes prédéfinies.
 - Comprendre l'avantage pour collaborer : un vocabulaire et des façons de faire communs.
 - Comprendre l'avantage du système de grille : positionnement et media queries prédéfinis.
 - Connaître l'existence de plusieurs solutions : Bootstrap, Foundation, Knacss, Rössti, Inuitcss.
- Présentation de Bootstrap
 - Savoir que Bootstrap est en Mobile First : les media queries se font avec `min-width`.
 - Connaître les breakpoints : `xs`, `sm`, `md`, `lg`.
 - Savoir réaliser une interface différente pour les 4 tailles, avec les classes `.col-*`.
 - Savoir combiner les classes `.container`, `.row`, `.col-*`.
 - Savoir réordonner les colonnes avec `.col-push-*` et `.col-pull-*`.
 - Connaître les classes utiles : `.hidden-*`, `.visible-*`.
 - Savoir utiliser les classes prédéfinies pour les boutons : `.btn`, `.btn-primary`, `.btn-secondary`...
 - Savoir utiliser les classes prédéfinies pour les formulaires : `.form-group`, `.form-control`...
 - Aborder les classes de typographie.

Bonus

- Aborder Bootstrap 4 et les thèmes Bootstrap.
 - Montrer que Bootstrap peut être configurable en Sass.
 - Présentation de Boilerplate.
 - Présentation d'un préprocesseur : Stylus, Sass, Less.
 - Présentation de postprocesseur : PostCSS.
-

CSS3 avancé (48)

- Compatibilité
 - Comprendre les problématiques des nouvelles technologies CSS et JS.
 - Maîtriser l'utilisation des préfixes propriétaires.
 - Connaître des ressources permettant de s'informer sur la compatibilité d'une propriété : caniuse.com, html5please.com.
- Les transitions
 - Maîtriser la syntaxe `transition`.
 - Aborder les cas d'utilisations classiques d'une transition : hover sur un lien...
- Les animations
 - Maîtriser la syntaxe `animation`.
 - Comprendre le concept d'images clés, et la syntaxe `@keyframes`.
 - Aborder les cas d'utilisations classiques d'une animation : loader CSS, effets `fadeIn` / `fadeOut`...
 - Savoir recréer une animation jQuery simple avec `animation`.
- Les transformations
 - Aborder les transformations 3D : `translate3d`, `rotateY`, `perspective`...

Bonus

- Aborder la propriété `will-change`.
-

Les objets en PHP (49)

- Les avantages de la POO
 - Savoir, sans les avoir expérimenté, les principaux avantages de la POO
- Les classes et les propriétés
 - Connaître les bonnes pratiques de la définition des classes (fichier unique, majuscule)
 - Savoir créer une classe
 - Savoir définir des propriétés publiques
- L'instanciation
 - Comprendre la distinction entre classe et instance
 - Savoir instancier un objet
 - Savoir assigner des valeurs aux propriétés publiques
- Les méthodes
 - Savoir créer des méthodes publiques dans une classe
 - Savoir appeler les méthodes
- La visibilité
 - Comprendre les visibilités `public` et `private`
 - Être initié au concept de `getter` et `setter`
- `$this`
 - Comprendre la relation entre la variable `$this` et l'instance

- Savoir faire référence à une propriété ou à une méthode dans la classe
 - Constructeur et destructeur
 - Comprendre l'utilisation et la raison d'être du constructeur
 - Connaître l'existence du destructeur
-

Introduction à la POO, partie 1 (50)

- L'héritage et le polymorphisme
 - Savoir comment hériter des propriétés et des méthodes d'une classe parente
 - Connaître quelques cas d'utilisation
 - Comprendre la visibilité protected
 - Connaître les limites de l'héritage
 - Savoir comment et pourquoi surcharger une méthode
 - Savoir comment appeler une méthode de la classe parente
- Les espaces de noms
 - Comprendre l'utilité des espaces de noms
 - Savoir comment déclarer une classe dans un namespace
 - Savoir comment faire référence à une classe par son FQCN
 - Savoir comment aliaser une classe
- Autochargement de classes
 - Savoir utiliser `spl_autoload_register()` sans paramètre, en combinaison avec les namespaces

Bonus

- PSR-0
 - Méthodes magiques
 - Le typage d'objets
-

Introduction à la POO, partie 2 (51)

- Interfaces
 - Connaître l'existence et le fonctionnement des interfaces
 - Reconnaître le mot clef implements
- Abstraction de classes
 - Connaître l'existence et le fonctionnement des classes et méthodes abstraites
- Méthodes et propriétés statiques
 - Comprendre l'opérateur de résolution de portée
 - Comprendre l'utilité des méthodes et propriétés statiques
 - Connaître par coeur l'appellation Paamayim Nekudotayim
 - Savoir utiliser des méthodes statiques
- POO : Structure, organisation

- Avoir des notions de structure de dossiers et de classes en POO

Bonus

- Design patterns célèbres
 - Les traits
-

Qualité Web, partie 1 (52)

- Notions générales
 - Présentation des bonnes pratiques
-

Qualité Web, partie 2 (53)

- Présentation des bonnes pratiques (suite)
 - Exercices
 - Certification OpQuast Website 2
-

Atelier pratique HTML5/CSS3 : exercices avancés (54)

Atelier pratique HTML5/CSS3 : portfolio (55)

AJAX, webservices et API (56)

- Comment consommer un webservice en AJAX
 - Comprendre la problématique du CORS
 - Savoir faire appel à un webservice simple
 - Savoir récupérer et manipuler des données JSON
 - Connaître les points communs des documentations de webservice
 - Connaître quelques exemples de webservice utiles en AJAX
- Google Maps API
 - Savoir créer une carte
 - Savoir utiliser l'autocomplétion d'adresse
 - Savoir recentrer une carte
 - Savoir placer un marqueur sur une carte
- Structure de code complexe en JavaScript

- Revoir les bonnes pratiques d'organisation de code JavaScript

Bonus

- Géolocalisation en JavaScript
 - JSONP
-

Architecture MVC (57)

- Les vues
 - Comprendre quels éléments d'un projet web composent les vues
 - Connaître le fonctionnement type des moteurs de template
- Les modèles
 - Comprendre quels éléments d'un projet web composent le modèle
 - Data Mapper vs Active Record : comprendre la différence entre les 2 stratégies
- Les contrôleurs
 - Comprendre le rôle des contrôleurs
- Contrôleur frontal
 - Comprendre l'utilité d'un contrôleur frontal
- La réécriture d'URL
 - Connaître les bases de la réécriture d'URL avec le .htaccess
 - Être en mesure de rediriger les requêtes vers le contrôleur frontal
 - Être capable de comprendre un code de réécriture (RewriteCond et RewriteRule)
- Le routage
 - Comprendre le concept de définition de routes et les avantages

Bonus

- Bibliothèques utiles
-

Introduction aux frameworks PHP (58)

- Frameworks vs CMS vs "from scratch"
 - Les caractéristiques des frameworks vs CMS
 - Comprendre quelle solution utiliser en fonction du projet
 - Connaître quelques CMS populaires et leurs principales caractéristiques
- Tour d'horizon des frameworks populaires
 - Connaître quelques frameworks PHP populaires et leurs principales caractéristiques
- Concepts fondamentaux et points communs
 - Être familiarisé avec la structure d'un code réalisé dans un framework
- W : prise en main
 - Présentation et installation

- Comprendre l'intérêt du framework pédagogique
- Savoir installer de manière autonome le framework
- Savoir configurer de manière autonome le framework
- Prise en main
- Savoir créer des pages simples dans le framework
- Comprendre le parcours d'une requête au serveur jusqu'à la réponse au client
- Savoir consulter la documentation
- MVC
- Comprendre quels éléments du framework occupent quels rôles

Bonus

- Savoir lire les commentaires DocBlocks
-

Framework W : Créer une application, partie 1 (59)

- Les routes
 - Comprendre l'utilité des composants d'une route
 - Être en mesure de définir des routes simples
 - Être en mesure de définir des routes avec des paramètres dynamiques
 - Les contrôleurs
 - Être en mesure de créer de nouvelles méthodes, associées à des routes
 - Savoir récupérer les éventuels paramètres d'URL
 - Être en mesure de créer un nouveau contrôleur
 - Savoir faire une redirection vers une autre route
 - Les gestionnaires
 - Être en mesure de créer un gestionnaire associé à une table MySQL
 - Être en mesure d'exécuter des requêtes de base avec le gestionnaire
 - Être en mesure de définir des requêtes personnalisées dans le gestionnaire
 - Les templates
 - Être en mesure de hériter d'un *layout*
 - Comprendre le concept de *section*
 - Savoir comment rendre disponible des données du contrôleur dans la vue avec `show()`
-

Framework W : Créer une application, partie 2 (60)

- Les utilisateurs et la sécurité
 - Authentification
 - Comprendre les configurations liées aux utilisateurs
 - Être en mesure d'enregistrer un nouvel utilisateur
 - Être en mesure de connecter et déconnecter un utilisateur

- Autorisation
- Savoir comment récupérer les données de l'utilisateur connecté (contrôleur et vue)
- Comprendre le concept de rôle
- Savoir comment autoriser l'accès à une méthode de contrôleur en fonction du rôle

Bonus

- Savoir envoyer une réponse JSON
-

Projet (61 à 70)
