

Ce glossaire est prévu pour valider les connaissances après chaque exercice clé de votre formation. L'idée est de réussir à définir les différentes notions en une à 3 phrases maximum et d'adopter le vocabulaire technique approprié. Ce document vous servira comme base de révisions.

A chaque fois qu'un acronyme est cité dans le glossaire, il sera impératif de donner la signification de chaque lettre initiale.

- Général
- Front-end
- UX / UI
- Programmation orientée objet
- Architecture
- Modélisation / base de données
- Symfony
- Sécurité
- RGPD
- SEO
- Gestion de projets / DevOps
- English

I. Général

1. Quel est l'environnement à installer pour exécuter un script PHP ? Citer 2 exemples de logiciels permettant ce contexte
2. Qu'est-ce qu'un algorithme ?
3. Qu'est-ce qu'une variable ? Par quel symbole est préfixée une variable en PHP ?
4. Qu'est-ce que la portée d'une variable ?
5. Qu'est-ce qu'une constante ? Quelle est la différence avec une variable ?
6. Qu'est-ce qu'une superglobale, combien en existent-ils et donner un exemple d'utilisation
7. Quels sont les différents types (primitifs) que l'on peut associer à une variable en PHP ? Les citer et en donner des exemples (ne pas oublier le type d'une variable sans valeur)
8. Existe-t-il plusieurs types de tableaux en PHP, si oui lesquels ?

9. Quelles sont les différentes structures de contrôles qu'il existe en algorithmie ? Donner un exemple pour chacune d'entre elles
10. Quelle est la fonction PHP permettant de demander la longueur d'une chaîne de caractères ?
11. Qu'est-ce qu'une session ? Quelle fonction permet de démarrer une session en PHP ? Donner un exemple d'utilisation en PHP
12. Qu'est-ce qu'un cookie ? Donner un exemple d'utilisation en PHP
13. Quelle est la différence entre les instructions « require » et « include » en PHP
14. Comment effectuer une redirection en PHP ?
15. Définir la partie « front-end » et « back-end » d'une application
16. Définir le contrôle de version ? Qu'est-ce que Git ?
17. Qu'est-ce qu'un CMS ? Citer au moins 2 exemples

II. Front-end

18. Définir HTML
19. Définir CSS
20. Définir Javascript
21. Définir JSON. Dans quel contexte ce format est-il utilisé ?
22. Peut-on interpréter du Javascript côté serveur ? Si oui, comment ?
23. Qu'est-ce qu'un sélecteur CSS ?
24. Quelle balise HTML permet de créer un lien hypertexte ?
25. Qu'est-ce qu'une requête AJAX ?
26. Quel sélecteur CSS permet de sélectionner tous les éléments d'une classe spécifique ? D'un identifiant spécifique ?
27. Définir le responsive design
28. Qu'est-ce que le templating ?
29. Qu'est-ce qu'une fonction anonyme en Javascript ?
30. Quelle méthode JavaScript est utilisée pour ajouter un élément à la fin d'un tableau ?

- 31. Qu'est-ce qu'un « media query » ?
- 32. Qu'est-ce qu'un pseudo élément en CSS ?
- 33. Qu'est-ce que Bootstrap ? Donner d'autres exemples équivalent
- 34. Quand un formulaire HTML est créé, quelles sont les 2 méthodes qui peuvent lui être associées ? Donner la différence entre ces 2 méthodes

III. UX / UI

- 35. Quelle est la différence entre UX Design et UI Design ?
- 36. Qu'est-ce qu'un wireframe ?
- 37. Qu'est-ce qu'un prototype ?
- 38. Qu'est-ce que la hiérarchie visuelle en UI Design ?
- 39. Qu'est-ce que l'accessibilité en UX Design ?
- 40. Qu'est-ce qu'une grille de mise en page ?
- 41. Qu'est-ce que la notion d'affordance en UX Design ?
- 42. Qu'est-ce qu'un « mobile first design » ?

IV. Programmation orientée objet (POO)

- 43. Donner une définition de la programmation orientée objet
- 44. Qu'est-ce qu'une classe ? Comment la déclare-t-on ?
- 45. Qu'est-ce qu'un objet ?
- 46. Définir la notion de propriété / attribut / méthode
- 47. Qu'est-ce que la visibilité d'une propriété ou d'une méthode ? Citer les différents types de visibilité
- 48. Quelle est la méthode spécifique utilisée pour créer un nouvel objet à partir d'une classe ?
- 49. Qu'est-ce que l'encapsulation ?
- 50. Que signifie « étendre une classe » ? Quelle est le concept clé mis en œuvre ? Donner un exemple
- 51. Définir l'opérateur de résolution de portée

- 52. Définir une méthode / propriété statique
- 53. Définir le polymorphisme en POO
- 54. Définir une méthode / classe abstraite ?
- 55. Définir le chaînage de méthodes
- 56. Qu'est-ce que la méthode __toString() ? Existe-t-il d'autres méthodes « magiques »
- 57. Qu'est-ce qu'un « autoload » ?
- 58. Comment appelle-t-on en français les « getters » et les « setters » ?
- 59. Qu'est-ce que la sérialisation en PHP ?

V. Architecture

- 60. Qu'est-ce que l'architecture client / serveur ? Grâce à quel type de requête peut-on interroger le serveur. Définir l'acronyme de ce type de requête. Si on ajoute un « S » à cet acronyme, expliquer la différence
- 61. Donner la définition d'un design pattern. Citer au moins 3 exemples de design pattern
- 62. Qu'est-ce que l'architecture MVC ?
- 63. Quel est le rôle de chaque couche du design pattern MVC : Model, View, Controller ?
- 64. Quels sont les avantages de l'architecture MVC ?
- 65. Existe-t-il des variantes à l'architecture MVC ?
- 66. Qu'est-ce qu'une API ? Définir l'architecture REST

VI. Modélisation / Base de données

- 67. Qu'est-ce que la modélisation de données ? Définir la méthode Merise
- 68. Quelles sont les 3 étapes principales de la méthode Merise ?
 - a. Analyse, conception et réalisation
 - b. Planification, exécution et contrôle
 - c. Création, modification et suppression
- 69. Qu'est-ce qu'un modèle conceptuel de données (MCD) en Merise ?

70. Qu'est-ce qu'un modèle logique de données (MLD) en Merise ?

71. Donner la définition des mots suivants :

- a. Entité
- b. Relation
- c. Cardinalité
- d. Clé primaire / clé étrangère

72. Que devient une relation de type « Many To Many » dans le modèle logique de données ?

73. Qu'est-ce qu'une base de données ?

74. Définir les notions suivantes :

- a. SQL
- b. MySQL
- c. SGBD (donner 2 exemples de SGBD)

75. Dans une base de données, les données sont stockées dans des _____. Celles-ci sont constituées de lignes appelées _____ et de colonnes appelées _____

76. Quelle est la différence entre une base de données relationnelle et non relationnelle ?

77. Qu'est-ce qu'une jointure dans une base de données ? En existe-t-il plusieurs ? Si oui lesquelles ?

78. A quoi sert une vue dans une base de données ?

79. Qu'est-ce que l'intégrité référentielle dans une base de données ?

80. Quelles sont les fonctions d'agrégation en SQL ?

81. Qu'est-ce qu'un CRUD dans le contexte d'une base de données ?

82. Quelles sont les clauses qui permettent de :

- a. Insérer un nouvel enregistrement dans une table
- b. Modifier un enregistrement dans une table
- c. Supprimer un enregistrement dans une table
- d. Supprimer la base de données

- e. Filtrer les résultats d'une requête SQL
- f. Trier les résultats d'une requête SELECT
- g. Regrouper les résultats d'une requête SELECT en fonction d'une colonne spécifique
- h. Concaténer 2 chaînes de caractères

83. Comment se connecter à une base de données en PHP ? Quelle est la classe native utilisée ?

VII. Symfony

- 84. Qu'est-ce que Symfony ?
- 85. Sur quel langage de programmation et design pattern repose Symfony ?
- 86. Quelle est la dernière version en date de Symfony ?
- 87. Qu'est-ce qu'un bundle ?
- 88. Quel est le moteur de template utilisé par défaut dans Symfony ?
- 89. Qu'est-ce qu'un ORM ? Quel est son utilité et comment s'appelle-t-il au sein de Symfony ?
- 90. Qu'est-ce que l'injection de dépendances ? Quel est l'outil utilisé dans ce contexte et quel fichier contient l'intégralité des dépendances du projet ?
- 91. Que permet le bundle Maker au sein de Symfony ?
- 92. Quel est le langage de requêtage exploité au sein d'un projet Symfony ?
- 93. Quel est le composant qui garantit l'authentification et l'autorisation des utilisateurs ?

VIII. Sécurité

- 94. Qu'est-ce que l'injection SQL ? Comment s'en prémunir ?
- 95. Qu'est-ce que la faille XSS ? Comment s'en prémunir ?
- 96. Qu'est-ce que la faille CSRF ? Comment s'en prémunir ?
- 97. Définir l'attaque par force brute et l'attaque par dictionnaire
- 98. Existe-t-il d'autres failles de sécurité ? Citer celles-ci et expliquer simplement leur comportement
- 99. A quoi servent l'authentification et l'autorisation dans un contexte d'application web ?

- 100. Définir la notion de hachage d'un mot de passe et citer des algorithmes de hachage
- 101. Qu'est-ce qu'une politique de mots de passe forts ?
- 102. Qu'est-ce que l'hameçonnage ?
- 103. Définir la « validation des entrées »

IX. RGPD

- 104. Qu'est-ce que le RGPD ?
- 105. Quel est son objectif principal ?
- 106. Quelle est la date d'entrée en vigueur du RGPD ?
- 107. Quelles sont les sanctions possibles en cas de non-respect du RGPD ?
- 108. En France, quel est l'autorité administrative qui s'occupe de faire appliquer le RGPD ?
- 109. Quel est le consentement valide selon le RGPD ?
- 110. Qu'est-ce qu'une politique de confidentialité ?
- 111. Quelle est la durée de conservation maximale des données personnelles selon le RGPD ?
- 112. Quels sont les droits des utilisateurs selon le RGPD ?
- 113. Qu'est-ce que le principe de minimisation des données selon le RGPD ?

X. SEO

- 114. Qu'est-ce que le SEO ?
- 115. Quel est l'objectif principal du SEO ?
- 116. Existe-t-il plusieurs types de référencement ? Lesquels ?
- 117. Qu'est-ce que la densité de mots-clés en SEO ?
- 118. Qu'est-ce qu'une balise « alt » ?
- 119. Qu'est-ce que la balise « meta description » ?
- 120. Qu'est-ce que le « nofollow » en SEO ?
- 121. Quelle est l'importance du contenu de qualité pour le référencement d'un site web ?
- 122. Pourquoi est-il important d'utiliser des balises de titre (h1, h2, h3, etc.) de manière structurée ?
- 123. Quelle est la recommandation pour les URL d'un site web bien référencé ?
- 124. Qu'est-ce que le maillage interne et pourquoi est-il important pour le référencement ?
- 125. Qu'est-ce que l'optimisation des images pour le référencement ?
- 126. Qu'est-ce qu'un plan de site (sitemap) et pourquoi est-il important pour le référencement ?

XI. Gestion de projets / DevOps

- 127. Qu'est-ce que la gestion de projet ?
- 128. Qu'est-ce qu'une méthode Agile de gestion de projet ?
- 129. Expliquer la méthode MoSCoW en quelques lignes et citer ses avantages
- 130. A quoi sert la méthodologie MVP ? Citer les caractéristiques clés
- 131. Qu'est-ce que la planification itérative ?
- 132. Citer 3 méthodes Agiles dans le cadre d'un projet informatique
- 133. Qu'est-ce qu'une réunion de revue de projet ?
- 134. Qu'est-ce qu'un livrable dans un projet ?

GLOSSAIRE ~ QUIZZ

- 135. Quels sont les 3 piliers SCRUM ? Définir chacun d'entre eux
- 136. Qu'est-ce que le DevOps et quel est son objectif principal ?
- 137. Qu'est-ce que l'intégration continue ?
- 138. Qu'est-ce que Docker ? Et en quoi est-il utile dans le cadre du DevOps ?
- 139. Qu'est-ce qu'un test unitaire ?
- 140. Quelle est l'unité de code testée lors d'un test unitaire ?
- 141. Quelles sont les caractéristiques d'un bon test unitaire ?
- 142. Qu'est-ce qu'une assertion dans un test unitaire ?



XII. English

- 1) **What does JavaScript enable you to do on a website ?**
 - a. Add interactive behavior and dynamic content
 - b. Define the layout and design of web pages
 - c. Handle server-side operations
- 2) **Which programming language is primarily used for server-side web development ?**
 - a. PHP
 - b. JavaScript
 - c. HTML
- 3) **What is the purpose of a web browser ?**
 - a. To render and display web pages
 - b. To execute server-side code
 - c. To manage databases
- 4) **What is the difference between GET and POST methods in HTTP ?**
 - a. GET retrieves data from a server, while POST submits data to a server
 - b. GET submits data to a server, while POST retrieves data from a server
 - c. GET and POST methods are interchangeable
- 5) **What is the purpose of version control systems (e.g., Git) in web development ?**
 - a. To track changes and manage collaborative development
 - b. To optimize website loading speed
 - c. To handle server-side scripting
- 6) **What is the purpose of a framework in web development ?**
 - a. To provide a structured environment for building web applications
 - b. To handle network protocols and data transfer
 - c. To create visual designs and layouts for websites
- 7) **What does NoSQL stand for ?**
 - a. Not Only SQL
 - b. Non-Structured Query Language
 - c. New Object-Oriented Language
- 8) **Which of the following is a characteristic of NoSQL databases ?**
 - a. Strict schema enforcement
 - b. Support for complex transactions
 - c. Scalability and flexible data models