

Structure et Architecture Données

1. Quel est le principe de la méthode FIFO ?

- a) Le premier élément à entrer est le dernier à sortir.
- b) Le premier élément à entrer est le premier à sortir.
- c) Le dernier élément à entrer est le premier à sortir.
- d) Tous les éléments sortent en même temps.

Réponse :

1. **b)** - Le premier élément à entrer est le premier à sortir.

2 . Quel est un avantage clé de l'architecture basée sur les microservices ?

- a) Chaque microservice peut être développé, déployé et mis à jour indépendamment.
- b) Les microservices doivent être écrits dans le même langage de programmation pour garantir la compatibilité.
- c) Les microservices sont déployés sur un seul serveur central pour améliorer les performances.
- d) Tous les services sont regroupés dans une seule base de code pour faciliter la gestion.

Réponse :

2. **a)** - Chaque microservice peut être développé, déployé et mis à jour indépendamment.

Laravel: Arborescence, MVC et Configuration

3. Quelle est la structure d'arborescence par défaut dans Laravel pour un projet ?

- a) config, public, vendor, routes, controllers
- b) controllers, views, models, migrations, config
- c) app, config, database, public, routes, storage, vendor
- d) app, database, migrations, controllers, config, resources

Réponse :

3. **c)** - app, config, database, public, routes, storage, vendor

4. Dans l'architecture MVC de Laravel, où se trouvent les contrôleurs par défaut ?

- a) app/Controllers
- b) app/Http/Controllers
- c) resources/views/controllers
- d) routes/controllers

Réponse :

4. **b)** - `app/Http/Controllers`

5. Quel est le rôle du fichier `.env` dans Laravel ?

- a) Définir les variables d'environnement, comme la configuration de la base de données.
- b) Stocker les fichiers de configuration.
- c) Permettre un accès simplifié aux variables d'environnement.
- d) Gérer les migrations de base de données.

Réponse :

- 5. **a)** - Définir les variables d'environnement, comme la configuration de la base de données.
- c)** - Permettre un accès simplifié aux variables d'environnement.

6. Quel dossier contient les modèles dans Laravel par défaut ?

- a) App/models
- b) resources/models
- c) app/Models
- d) app/Http/Models

Réponse :

6. **c)** - `app/Models`

7. Quelle commande Artisan permet de générer un contrôleur ?

- a) `php artisan make:controller`
- b) `php artisan make:model`
- c) `php artisan create:controller`
- d) `php artisan generate:controller`

Réponse :

7. **a)** - `php artisan make:controller`

8. Le modèle de base de données dans Laravel est configuré dans quel fichier ?

- a) `app/database.php`
- b) `config/app.php`
- c) `routes/web.php`
- d) `config/database.php`

Réponse :

8. **d)** - config/database.php

9. Comment accède-t-on à la configuration d'une base de données dans Laravel ?

- a) Via le fichier `database/config.php`
- b) Via le fichier `config/database.php`
- c) Via le fichier `.env`
- d) En configurant le fichier `app/Database.php`

Réponse :

9. **b) et c)** - Via le fichier `.env` et `config/database.php`

10. Où se trouve le fichier de configuration des routes web dans Laravel ?

- a) `app/routes/web.php`
- b) `config/web.php`
- c) `resources/routes/web.php`
- d) `routes/web.php`

Réponse :

10. **d)** - routes/web.php

11. Quel fichier contient les informations de configuration de l'application dans Laravel ?

- a) app/config.php
- b) config/app.php
- c) .env
- d) routes/web.php

Réponse :

11. **b)** - config/app.php

12. Comment exécuter une migration de base de données dans Laravel ?

- a) `php artisan migrate:run`
- b) `php artisan db:migrate`
- c) `php artisan migrate`
- d) `php artisan make:migration`

Réponse :

12. c) - php artisan migrate

Git - Opérations courantes

13. Quelle commande Git permet de cloner un dépôt distant ?

- a) `git clone`
- b) `git pull`
- c) `git remote clone`
- d) `git fetch`

Réponse :

13. **a)** - git clone

14. Quelle commande est utilisée pour ajouter un fichier à l'index (staging area) avant de le valider (commit) ?

- a) `git push`
- b) `git stage`
- c) `git add`
- d) `git commit`

Réponse :

14. **c)** - `git add`

15. Quelle commande permet d'afficher l'historique des commits dans un dépôt ?

- a) `git show`
- b) `git history`
- c) `git status`
- d) `git log`

Réponse :

15. **d)** - `git log`

16. Que fait la commande `git pull` ?

- a) Elle télécharge les modifications sans les fusionner.
- b) Elle récupère les modifications d'un dépôt distant.
- c) Elle envoie les modifications locales vers un dépôt distant.
- d) Elle fusionne les modifications distantes avec la branche actuelle.

Réponse :

- 16. **b)** - Elle récupère les modifications d'un dépôt distant.
- d)** - Elle fusionne les modifications distantes avec la branche actuelle.

Laravel Artisan, Tinker, Migration et Seeder

17. Quelle commande Artisan permet de créer une nouvelle migration ?

- a) `php artisan create:migration`
- b) `php artisan make:migration`
- c) `php artisan generate:migration`
- d) `php artisan migrate:new`

Réponse :

17. **b)** - `php artisan make:migration`

18. Que permet de faire Laravel Tinker ?

- a) Déployer une application sur un serveur.
- b) Exécuter des commandes Artisan en arrière-plan.
- c) Lancer un shell interactif pour exécuter du code PHP et interagir avec l'application.
- d) Créer des contrôleurs et des modèles via des lignes de commande.

Réponse :

18. **c)** - Lancer un shell interactif pour exécuter du code PHP et interagir avec l'application.

19. Comment lancer un seeder dans Laravel ?

- a) `php artisan db:seed`
- b) `php artisan seed:run`
- c) `php artisan migrate:seed`
- d) `php artisan seeder:db`

Réponse :

19. **a)** - php artisan db:seed

20. **Quelle commande Artisan est utilisée pour exécuter toutes les migrations ?**

- a) `php artisan db:migrate`
- b) `php artisan migrate`
- c) `php artisan run:migrations`
- d) `php artisan migrate:run`

Réponse :

20. **b)** - php artisan migrate