

Bases de données NoSQL

Durée : 30 minutes

Aucun document ou matériel électronique n'est autorisé.

Le barème est donné à titre indicatif.

Prénom NOM _____

Cet examen comporte 50 questions, pour un total de $61\frac{1}{2}$ points et 3 points bonus. L'examen contient 16 pages, avant de commencer, veuillez vérifier que vous avez l'examen dans sa totalité.

QCM

La première partie de l'examen est un questionnaire à choix multiples, les questions peuvent avoir une ou plusieurs bonnes réponses. Les questions sont indépendantes les uns des autres. Renseigner votre réponse en noircissant la case correspondante.

Questions ouvertes

La seconde partie de l'examen est composé de question(s) ouverte(s), répondez directement sur le sujet. Si vous n'avez pas assez de place, vous pouvez écrire au recto de la dernière page.

Bon courage ! ☺

Question	Points	Score	Question	Points	Score
1	$\frac{1}{2}$		27	1	
2	$\frac{1}{2}$		28	0	
3	$\frac{1}{2}$		29	$1\frac{1}{2}$	
4	$\frac{1}{2}$		30	$1\frac{1}{2}$	
5	$\frac{1}{2}$		31	$1\frac{1}{2}$	
6	1		32	1	
7	1		33	1	
8	$\frac{1}{2}$		34	1	
9	$\frac{1}{2}$		35	1	
10	1		36	$\frac{1}{2}$	
11	1		37	1	
12	1		38	1	
13	1		39	1	
14	1		40	1	
15	1		41	1	
16	1		42	1	
17	1		43	1	
18	1		44	1	
19	1		45	1	
20	1		46	5	
21	1		47	5	
22	1		48	5	
23	1		49	5	
24	1		50	0	
25	1				
26	1		Total:	$61\frac{1}{2}$	

QCM

1. ($\frac{1}{2}$ point) Qu'est-ce qu'une base de données NoSQL ?
 - ☐ Une base de données relationnelle
 - **Une base de données non relationnelle**
 - ☐ Une base de données en lecture seule
 - ☐ Une base de données exclusivement utilisée par les grandes entreprises
2. ($\frac{1}{2}$ point) Parmi les bases de données ci-dessous, laquelle est une base de données NoSQL ?
 - **Redis**
 - ☐ MySQL
 - ☐ PostgreSQL
 - ☐ Oracle
3. ($\frac{1}{2}$ point) Quels sont les quatre principaux types de bases de données NoSQL ?
 - ☐ Colonne, Clé-Valeur, XML, Graph
 - ☐ Clé-Valeur, Table, Document, SQL
 - ☐ JSON, Colonne, Document, Graph
 - **Clé-Valeur, Colonne, Document, Graph**
4. ($\frac{1}{2}$ point) Quel type de base de données NoSQL est le mieux adapté pour stocker des données hiérarchiques ?
 - ☐ Clé-Valeur
 - ☐ Colonne
 - **Graph**
 - ☐ Document
5. ($\frac{1}{2}$ point) Dans une base de données de type document, sous quelle forme les données sont-elles stockées ?
 - ☐ Tables
 - ☐ Clé-valeur
 - **JSON ou BSON**
 - ☐ Colonnes
6. (1 point) Quelle(s) est (sont) la (les) caractéristique(s) principale(s) des bases de données NoSQL ?
 - **Propriétés BASE**
 - ☐ Transactions ACID
 - ☐ Langage SQL
 - ☐ ODBC

7. (1 point) Quelle(s) base(s) de données NoSQL utilise(nt) une structure en tableau de hachage distribué ?
- ☐ Cassandra
 - ☐ MongoDB
 - **Redis**
 - ☐ Neo4j
8. ($\frac{1}{2}$ point) De quel type de base de données NoSQL est Cassandra ?
- ☐ Clé-Valeur
 - **Colonne**
 - ☐ Graph
 - ☐ Document
9. ($\frac{1}{2}$ point) Quelle base de données NoSQL est conçue pour les requêtes de graphe complexe ?
- ☐ Redis
 - ☐ Cassandra
 - **Neo4j**
 - ☐ MongoDB
10. (1 point) Quel(s) mécanisme(s) est (sont) souvent utilisé(s) pour assurer la disponibilité et la partition tolérance dans les bases de données NoSQL ?
- ☐ Indexation
 - **Sharding**
 - ☐ Joins
 - ☐ Transactions
11. (1 point) Dans une base de données NoSQL de type clé-valeur, que représente la **clé** ?
- ☐ Une table entière
 - ☐ Une colonne de la base de données
 - **Un identifiant unique pour accéder à la valeur**
 - ☐ Une fonction d'indexation
12. (1 point) Quelle est une limitation courante des bases de données NoSQL comparée aux bases de données SQL traditionnelles ?
- ☐ Flexibilité du schéma
 - ☐ Disponibilité
 - ☐ Scalabilité
 - **Gestion des transactions complexes**

13. (1 point) Comment les bases de données NoSQL assurent-elles généralement la consistance des données ?
- ☐ Par des transactions ACID
 - ☐ Par des triggers
 - **Par des mécanismes de versionnage et de synchronisation**
 - ☐ Par des procédures stockées
14. (1 point) Quel type de base de données NoSQL est idéal pour les systèmes de recommandation comme ceux utilisés par les réseaux sociaux ?
- ☐ Colonne
 - ☐ Clé-Valeur
 - **Graph**
 - ☐ Document
15. (1 point) Qu'est-ce que la partition tolérance dans le contexte des bases de données NoSQL ?
- **La capacité à supporter une perte de communication entre les nœuds sans perdre de données**
 - ☐ La capacité à exécuter des transactions ACID
 - ☐ La capacité à exécuter des requêtes SQL complexes
 - ☐ La capacité à maintenir des connexions simultanées
16. (1 point) En utilisant MongoDB, comment créer un index pour améliorer les performances des requêtes sur le champ name dans une collection users ?
- ☐ `db.users.ensureIndex({name:1})`
 - `db.users.createIndex({name:1})`
 - ☐ `db.users.index({name:1})`
 - ☐ `db.users.addIndex({name:1})`
17. (1 point) Quelle technique est souvent utilisée dans Redis pour gérer les expirations de clés et optimiser la mémoire ?
- **Time to Live**
 - ☐ Lazy deletion
 - ☐ Garbage collection
 - ☐ Snapshotting
18. (1 point) Quel est le langage de requêtage utilisé par Neo4j ?
- ☐ SQL
 - **Cypher**
 - ☐ CQL
 - ☐ Gremlin

19. (1 point) Comment les bases de données NoSQL gèrent-elles généralement les mises à jour simultanées pour éviter les conflits ?
- ☐ En utilisant des verrous (locks) sur les enregistrements
 - ☐ En utilisant des transactions ACID
 - **En appliquant des techniques de versionnage et de contrôle d'accès optimiste**
 - ☐ En appliquant des procédures stockées
20. (1 point) Dans MongoDB, quelle stratégie de sharding permet de distribuer les données de manière uniforme pour équilibrer la charge entre les nœuds ?
- ☐ Sharding par plage
 - **Sharding par hash**
 - ☐ Sharding par liste
 - ☐ Sharding par partition
21. (1 point) Comment Cassandra assure-t-elle la tolérance aux pannes lorsqu'un nœud devient indisponible ?
- ☐ En utilisant le partitionnement par hachage
 - ☐ En utilisant des transactions distribuées
 - **En répliquant les données sur plusieurs nœuds avec une stratégie de réplication**
 - ☐ En stockant toutes les données sur un seul nœud principal
22. (1 point) Quelle méthode utilise Redis pour persister les données en mémoire sur le disque ?
- ☐ Logging
 - ☐ Redis Database Folder
 - ☐ Write-ahead logging
 - ☐ Checkpointing
 - **Append-Only File et Redis Database File**
23. (1 point) Dans un système de bases de données NoSQL distribuées, quel est l'impact du théorème CAP sur le design des systèmes ?
- ☐ Il permet de choisir entre consistance et disponibilité tout en garantissant la tolérance au partitionnement.
 - ☐ Il impose de choisir entre disponibilité et tolérance au partitionnement, mais garantit toujours la consistance.
 - ☐ Il force les systèmes à sacrifier la tolérance au partitionnement pour obtenir consistance et disponibilité.
 - **Il impose des compromis entre consistance, disponibilité, et tolérance au partitionnement, mais ne permet pas d'obtenir les trois propriétés simultanément.**

24. (1 point) Comment fonctionne la réplication maître-esclave dans une base de données NoSQL comme MongoDB ?
- ☐ Les données sont écrites sur les esclaves et répliquées de manière synchrone sur le maître.
 - ☒ **Les données sont écrites sur le maître et répliquées de manière asynchrone sur les esclaves.**
 - ☐ Les données sont écrites sur les esclaves et répliquées de manière asynchrone sur le maître.
 - ☐ Les données sont écrites sur le maître et répliquées de manière synchrone sur les esclaves.
25. (1 point) Quel(s) avantage(s) majeur(s) offre le modèle de données orienté colonnes ?
- ☐ Meilleure gestion des relations complexes entre les données.
 - ☒ **Meilleure performance pour les écritures massives et les lectures par lot.**
 - ☐ Meilleure compatibilité avec les systèmes SQL traditionnels.
 - ☐ Simplification du schéma de la base de données.
26. (1 point) Dans une base de données orientée document, comment est-il possible de garantir l'atomicité des opérations sur plusieurs documents ?
- ☐ En utilisant des transactions ACID sur plusieurs documents.
 - ☐ En utilisant des transactions multi-documents.
 - ☐ En utilisant des techniques de contrôle de version et des verrous de ligne.
 - ☒ **Cela n'est généralement pas possible ; les opérations sont atomiques au niveau du document unique.**
27. (1 point) Quel est le rôle des réplicas dans une base de données NoSQL distribuée ?
- ☒ **Assurer la redondance des données et améliorer la tolérance aux pannes.**
 - ☐ Réduire la latence des requêtes en mettant en cache les réponses.
 - ☐ Améliorer la performance des écritures en les distribuant sur plusieurs nœuds.
 - ☐ Simplifier la gestion des transactions distribuées.
28. (1 point bonus) Quelle est une des principales différences entre les bases de données NoSQL et les bases de données NewSQL ?
- ☐ NoSQL est orienté document tandis que NewSQL est orienté colonne.
 - ☒ **NoSQL se concentre sur la scalabilité horizontale tandis que NewSQL vise à combiner la scalabilité avec les propriétés ACID des bases de données relationnelles.**
 - ☐ NoSQL utilise SQL pour les requêtes tandis que NewSQL utilise des API personnalisées.
 - ☐ NoSQL ne supporte pas la scalabilité horizontale, contrairement à NewSQL.

29. (1½ points) Quel(s) facteur(s) doit (doivent) être pris en compte pour choisir entre une base de données NoSQL de type Document et une base de données NoSQL de type Graph ?
- **La nature des relations entre les données**
 - ☐ La taille et la complexité des documents
 - **La nécessité de requêtes complexes et de traversées rapides des relations**
 - **La fréquence des mises à jour des documents**
30. (1½ points) Quel(s) est (sont) le(s) mécanisme(s) utilisé(s) par les bases de données NoSQL pour assurer la disponibilité et la partition tolérance dans un environnement distribué ?
- **Réplication des données**
 - **Sharding**
 - ☐ Transactions ACID strictes
 - **Cohérence éventuelle**
31. (1½ points) Lors de la modélisation des données dans une base de données NoSQL, quelle(s) est (sont) la (les) meilleure(s) pratique(s) pour optimiser les performances des requêtes ?
- **Utilisation de structures de données imbriquées pour réduire le nombre de requêtes**
 - **Création d'index sur les champs fréquemment interrogés**
 - ☐ Normalisation des données pour minimiser la redondance
 - **Dé-normalisation des données pour réduire le besoin de jointures complexes**
32. (1 point) Que signifient les initiales du théorème CAP ?
- ☐ Complexity Accuracy Performance
 - ☐ Centralized Access Protocol
 - **Consistency Availability Partition Tolerance**
 - ☐ Consistency Availability Performance
 - ☐ Conformity Availability Partition Tolerance
33. (1 point) Quel(s) est (sont) le(s) principal(ux) compromi(s) à considérer lors de l'implémentation de la cohérence éventuelle dans une base de données NoSQL ?
- **Temps de latence des mises à jour**
 - ☐ Facilité de mise à jour des transactions complexes
 - **Disponibilité du système en cas de partition réseau**
 - ☐ Simplicité de la récupération des données cohérentes
34. (1 point) Dans quel(s) scénario(s) spécifique(s) l'utilisation de Neo4j serait-elle plus avantageuse que l'utilisation de MongoDB ?
- **Analyse de réseaux sociaux pour découvrir des relations complexes**
 - ☐ Gestion de données hiérarchiques avec de nombreux niveaux imbriqués

- **Application nécessitant des recommandations personnalisées basées sur les connexions entre utilisateurs**
 - Stockage de gros volumes de documents JSON semi-structurés
- 35. (1 point) Quel(s) est (sont) le(s) avantage(s) des bases de données NoSQL par rapport aux bases de données relationnelles traditionnelles ?
 - **Meilleure scalabilité horizontale**
 - **Flexibilité du schéma**
 - **Performance améliorée pour certaines opérations**
 - **Traitement rapide de grandes quantités de données distribuées**
- 36. ($\frac{1}{2}$ point) Quels types de bases de données NoSQL existent ?
 - **Clé-Valeur**
 - **Document**
 - **Graph**
 - Relationnel
 - **Orienté colonnes**
- 37. (1 point) Quel(s) est (sont) le(s) défi(s) courant(s) de la gestion des bases de données NoSQL ?
 - **Gestion de la cohérence des données**
 - **Modélisation des données sans schéma fixe**
 - **Support limité pour les transactions complexes**
 - **Sécurité des données**
- 38. (1 point) Quel(s) mécanisme(s) les bases de données NoSQL utilise(nt)-elles pour assurer la scalabilité ?
 - **Sharding**
 - **Réplication**
 - **Partitionnement**
 - **Balancement de charge**
- 39. (1 point) Quels sont les cas d'utilisation appropriés pour les bases de données NoSQL ?
 - **Applications nécessitant une grande scalabilité horizontale**
 - **Systèmes de gestion de contenu**
 - **Réseaux sociaux**
 - **Big Data**
- 40. (1 point) Alice a une application de commerce électronique où elle doit gérer des produits avec des attributs variés comme le nom, la description, les prix fluctuants et les stocks en temps réel. Quelle base de données NoSQL serait la plus appropriée pour Alice ?

■ **MongoDB pour sa flexibilité du schéma et sa capacité à gérer des données semi-structurées comme les variations de prix.**

- ☐ Redis pour la rapidité d'accès aux données spécifiques comme les niveaux de stock.
- ☐ Neo4j pour modéliser les relations entre les produits et les utilisateurs.
- ☐ Cassandra pour gérer efficacement les statistiques de vente.

41. (1 point) Bob développe une application de réseau social où les utilisateurs peuvent se connecter et interagir avec d'autres utilisateurs ainsi qu'avec du contenu généré par les utilisateurs. Quelle base de données NoSQL serait la plus adaptée pour gérer ces relations complexes ?

- ☐ Amazon DocumentDB pour stocker des informations d'utilisateur comme les préférences de contenu.

■ **Neo4j pour sa capacité à modéliser efficacement les réseaux sociaux et les relations complexes entre utilisateurs et contenu.**

- ☐ Amazon DynamoDB pour la rapidité d'accès aux données spécifiques comme les informations de profil utilisateur.
- ☐ MongoDB pour gérer les flux d'activité des utilisateurs.

42. (1 point) Sophia gère une plateforme de streaming vidéo où des millions d'utilisateurs regardent des vidéos en continu. Tous les jours, de nombreuses vidéos sont publiées sur la plateforme. Quelle base de données serait la plus appropriée pour gérer le catalogue de vidéos et les préférences des utilisateurs ?

- ☐ Neo4j pour modéliser les relations entre les utilisateurs et les vidéos visionnées.
- ☐ Couchbase pour la gestion des vidéos et des métadonnées avec une bonne performance en lecture.

■ **Cassandra pour gérer de très grandes quantités de données avec une haute disponibilité et une tolérance aux pannes.**

- ☐ Google Bigtable pour gérer efficacement les métadonnées des vidéos.

43. (1 point) Charlie a une application de jeu en ligne massivement multijoueur où des milliers de joueurs interagissent simultanément dans un monde virtuel. Quelle base de données NoSQL serait la plus adaptée pour gérer l'état du monde virtuel et les interactions des joueurs ?

- ☐ Memcached pour la rapidité d'accès aux données spécifiques comme les scores des joueurs.
- ☐ Neo4j pour modéliser les relations entre les joueurs et les alliances.
- ☐ Cassandra pour gérer efficacement les données de connexion des joueurs.

■ **Google Firestore pour stocker des entités de jeu complexes comme les personnages avec des attributs variés et évolutifs.**

44. (1 point) Jean développe une application de suivi de livraisons où les utilisateurs peuvent suivre en temps réel la localisation des colis et recevoir des mises à jour sur l'état de leurs commandes. Quelle base de données NoSQL serait la plus appropriée pour gérer ces informations en temps réel ?

■ **Amazon DynamoDB pour la rapidité d'accès aux informations spécifiques de suivi des colis et des commandes.**

- ☐ CouchDB pour stocker les détails des commandes avec des informations de suivi mises à jour en temps réel.
- ☐ Amazon Neptune pour modéliser les itinéraires de livraison des colis.
- ☐ Cassandra pour gérer efficacement les statuts de livraison des colis.

45. (1 point) Emma est en train de développer une plateforme de commerce électronique qui doit pouvoir traiter des milliards de commandes en une fraction de seconde, surtout lors des périodes de pointe comme les vacances. La redondance des données et la capacité à faire face à la perte de nœuds de stockage sont essentielles pour assurer une disponibilité continue du service. Quelle base de données NoSQL serait la plus adaptée pour répondre à ces exigences spécifiques ?

- ☐ MongoDB pour sa flexibilité du schéma et sa capacité à gérer des données semi-structurées à grande échelle.
- ☐ Neo4j pour modéliser efficacement les relations entre les produits, les utilisateurs et les commandes.

■ **Apache HBase pour sa capacité à gérer de vastes volumes de données non structurées avec une faible latence et une grande scalabilité.**

- ☐ Riak pour sa capacité à offrir une haute disponibilité et une résilience élevée face aux pannes tout en gérant des données distribuées.

48. (5 points) Quels sont les principaux défis associés à la gestion des bases de données NoSQL ?

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Solution: La gestion des bases de données NoSQL pose plusieurs défis, notamment la complexité de la modélisation des données sans schéma fixe, la nécessité de gérer la cohérence des données dans des environnements distribués, et les difficultés de mise en œuvre de transactions complexes. De plus, chaque type de base de données NoSQL a ses propres spécificités et limitations, nécessitant une expertise technique pour tirer pleinement parti de leurs capacités. Enfin, la sécurité des données et les stratégies de sauvegarde et de restauration peuvent également être plus complexes à mettre en œuvre que dans les bases de données relationnelles traditionnelles.

49. (5 points) Comment les bases de données graph, facilitent-elles l'analyse des relations complexes

nant bases de données SQL et NoSQL ?

This image shows a single sheet of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Solution: Une architecture hybride combinant bases de données SQL et NoSQL est préférable dans les scénarios où différentes parties de l'application ont des exigences de stockage et de requête distinctes. Par exemple, une application de commerce électronique peut utiliser une base de données relationnelle pour gérer les transactions financières et les informations des clients, assurant ainsi la cohérence et l'intégrité des données, tout en utilisant une base de données NoSQL pour stocker et analyser les données de navigation et les comportements des utilisateurs en temps réel. Cette approche permet de tirer parti des points forts de chaque type de base de données, optimisant ainsi les performances et la scalabilité de l'application.