

Examen Analyste Programmeur (Banque) – 8 Axes – Tech Lead (Questions + Réponses)

UML

Q1. Qu'est-ce qu'un stéréotype UML (<>, <>...) ? (niveau tech lead)

Réponse : Extension sémantique pour préciser le rôle d'un élément selon un profil (ex. UML pour logiciel). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q2. Association vs composition ? (niveau tech lead)

Réponse : Association = lien logique (cycle de vie indépendant). Composition = relation tout/partie forte (la partie dépend du tout, cycle de vie lié). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q3. Héritage vs interface (réalisation) en UML ? (niveau tech lead)

Réponse : Héritage = généralisation (est-un). Interface = contrat de comportements sans implémentation (selon langage). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q4. À quoi sert la multiplicité (0..1, 1..*, etc.) ? (niveau tech lead)

Réponse : Elle constraint le nombre d'instances participantes d'une association (cardinalités), utile pour règles métier et validation. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q5. Quand utiliser un diagramme d'activités ? (niveau tech lead)

Réponse : Pour modéliser un workflow/processus (décisions, parallélisme), ex. ouverture de compte, traitement de virement. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q6. Diagramme de composants : intérêt ? (niveau tech lead)

Réponse : Visualiser modules/services, dépendances, interfaces exposées, utile en architecture (microservices, APIs). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q7. Agrégation : à quoi sert-elle ? (niveau tech lead)

Réponse : Relation tout/partie faible : la partie peut exister sans le tout (ex. Banque agrège des Agences). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Networking

Q8. Qu'est-ce que le MTU et pourquoi c'est important ? (niveau tech lead)

Réponse : Taille max d'un paquet IP sur un lien. MTU mal aligné ⇒ fragmentation/perf; MSS TCP dépend du MTU. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q9. TCP vs UDP : principales différences ? (niveau tech lead)

Réponse : TCP est orienté connexion, fiable (ACK, retransmission, ordre). UDP est sans connexion, plus léger, pas de garantie d'ordre/fiabilité. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q10. DNS : rôle et mécanisme haut niveau ? (niveau tech lead)

Réponse : Résolution nom→IP via hiérarchie (root/TLD/autoritatifs) et cache (TTL). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q11. Load balancer L4 vs L7 ? (niveau tech lead)

Réponse : L4 route sur IP/port (TCP/UDP). L7 comprend HTTP (URL, headers), permet routage avancé, WAF, etc. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q12. CIDR : que signifie /24 ? (niveau tech lead)

Réponse : Préfixe réseau de 24 bits; en IPv4 cela correspond à 256 adresses (dont 254 utilisables en pratique). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q13. TLS/HTTPS : que protège-t-on ? (niveau tech lead)

Réponse : Confidentialité (chiffrement), intégrité (MAC), authentification (certificats). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q14. Qu'est-ce qu'un reverse proxy ? (niveau tech lead)

Réponse : Proxy côté serveur, termine TLS, fait cache, routage, protection, masquant les services internes. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

OOP

Q15. Encapsulation : bénéfice concret ? (niveau tech lead)

Réponse : Masquer l'état interne, contrôler invariants via méthodes, réduire couplage. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q16. SOLID : que signifie le 'D' ? (niveau tech lead)

Réponse : Dependency Inversion: dépendre d'abstractions, pas de concrétions. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q17. Qu'est-ce qu'un objet valeur (Value Object) ? (niveau tech lead)

Réponse : Objet identifié par ses valeurs (equals/hashCode), immutable; ex. IBAN, Money. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q18. Inversion of Control : idée ? (niveau tech lead)

Réponse : Le framework orchestre la création/injection (DI), l'application fournit des composants. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q19. Héritage vs composition : règle pratique ? (niveau tech lead)

Réponse : Préférer la composition (plus flexible) sauf vraie relation 'est-un' stable. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q20. 4 piliers de la POO ? (niveau tech lead)

Réponse : Encapsulation, Abstraction, Héritage, Polymorphisme. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q21. Polymorphisme : exemple ? (niveau tech lead)

Réponse : Une variable de type Interface peut référencer différentes implémentations; dispatch dynamique à l'exécution. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

DSA

Q22. Deux pointeurs : exemple classique ? (niveau tech lead)

Réponse : Déetecter paire somme cible dans tableau trié, ou retirer doublons in-place. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q23. HashMap : complexité moyenne insertion/recherche ? (niveau tech lead)

Réponse : Moyenne $O(1)$, pire $O(n)$ (collisions/pathologiques), atténué par resizing/arbres (Java). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q24. Tableau vs liste chaînée : trade-offs ? (niveau tech lead)

Réponse : Tableau: accès $O(1)$ mais insertion milieu coûteuse. Liste: insert/delete $O(1)$ avec pointeur mais accès $O(n)$. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q25. Tri rapide : moyenne vs pire ? (niveau tech lead)

Réponse : Moyenne $O(n \log n)$, pire $O(n^2)$ si pivot mauvais; randomisation réduit risque. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q26. Trie : pour quoi faire ? (niveau tech lead)

Réponse : Recherche préfixe/auto-complétion en $O(L)$ (longueur du mot). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q27. Fenêtre glissante : pour quoi ? (niveau tech lead)

Réponse : Sous-tableau/sous-chaîne optimale en $O(n)$ (ex. somme max longueur k). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q28. Arbre équilibré (AVL/RB) : intérêt ? (niveau tech lead)

Réponse : Garantir hauteur $O(\log n) \Rightarrow$ opérations $O(\log n)$ stables. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Design Patterns

Q29. Strategy : cas d'usage bancaire ? (niveau tech lead)

Réponse : Sélection de calcul de frais/commission selon type client/produit. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q30. Decorator : bénéfice ? (niveau tech lead)

Réponse : Ajouter responsabilités dynamiquement sans modifier la classe (ex. logging, métriques). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q31. CQRS : principe ? (niveau tech lead)

Réponse : Séparer modèles Command (écriture) et Query (lecture) pour scalabilité/perf. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q32. Adapter : quand ? (niveau tech lead)

Réponse : Intégrer API legacy/tiers avec interface différente (ex. switch de paiement). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q33. Template Method : idée ? (niveau tech lead)

Réponse : Squelette d'algorithme dans une classe mère, étapes personnalisées dans sous-classes. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q34. Singleton : pourquoi souvent critiqué ? (niveau tech lead)

Réponse : Cache dépendances, rend tests difficiles, état global, problèmes concurrence/ordre d'init. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q35. Command : cas d'usage ? (niveau tech lead)

Réponse : Encapsuler une requête (virement) pour queue/retry/audit. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Backend Patterns

Q36. AuthN/AuthZ : différence ? (niveau tech lead)

Réponse : AuthN = prouver identité; AuthZ = droits/permissions. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q37. REST : contraintes clés ? (niveau tech lead)

Réponse : Client-server, stateless, cacheable, interface uniforme, layered system. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q38. Idempotence : exemple en banque ? (niveau tech lead)

Réponse : POST virement avec idempotency-key : répéter la requête ne double pas l'opération. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q39. Saga : pourquoi ? (niveau tech lead)

Réponse : Gérer transactions distribuées via compensations (ex. réservation + débit + confirmation). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q40. Gestion erreurs API : bonnes pratiques ? (niveau tech lead)

Réponse : Codes HTTP, message clair, trace-id, pas de fuite sensible, mapping exceptions. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q41. Observabilité : 3 piliers ? (niveau tech lead)

Réponse : Logs, métriques, traces distribuées (OpenTelemetry). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q42. Caching : où et quoi ? (niveau tech lead)

Réponse : CDN/reverse proxy/app/DB; cache des lectures idempotentes, invalidation/TTL. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

SQL & Bases de données

Q43. Transactions : ACID ? (niveau tech lead)

Réponse : Atomicité, Cohérence, Isolation, Durabilité. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q44. INNER JOIN vs LEFT JOIN ? (niveau tech lead)

Réponse : INNER: intersection. LEFT: toutes lignes de gauche + NULL si pas de match à droite. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q45. EXPLAIN : pourquoi ? (niveau tech lead)

Réponse : Voir plan d'exécution (index scan, seq scan), diagnostiquer lenteurs. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q46. Niveaux isolation : exemple d'anomalie ? (niveau tech lead)

Réponse : READ COMMITTED évite dirty reads mais pas non-repeatable/phantom reads. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q47. Partitionnement : quand ? (niveau tech lead)

Réponse : Très grandes tables (transactions), améliore maintenance/perf par pruning. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q48. DELETE vs TRUNCATE ? (niveau tech lead)

Réponse : DELETE journalise ligne par ligne (WHERE possible). TRUNCATE vide la table rapidement, souvent DDL, réinitialise identity. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q49. Clé primaire vs unique ? (niveau tech lead)

Réponse : PK identifie ligne (non NULL, 1 par table). UNIQUE impose unicité (peut être multiple, NULL selon SGBD). (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Frontend Patterns

Q50. Props vs state ? (niveau tech lead)

Réponse : Props = données reçues (immutables côté enfant). State = état interne composant. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q51. Pattern MVVM : idée ? (niveau tech lead)

Réponse : ViewModel expose state/commands; data-binding entre View et ViewModel. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q52. Accessibility : exemple ? (niveau tech lead)

Réponse : Labels associés aux inputs, ARIA quand nécessaire, navigation clavier. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q53. Performance : re-render inutile ? (niveau tech lead)

Réponse : Memoization, keys stables, découper composants, éviter state global abusif. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q54. Gestion formulaires : bonnes pratiques ? (niveau tech lead)

Réponse : Validation côté client + serveur, messages accessibles, prévention CSRF, masquage données sensibles. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q55. Sécurité front : XSS ? (niveau tech lead)

Réponse : Injection de script; éviter innerHTML, échapper/sanitizer, CSP. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Q56. Lazy loading : bénéfice ? (niveau tech lead)

Réponse : Réduire bundle initial, améliorer performance (LCP) en chargeant à la demande. (attendu : décisions d'architecture, risques, gouvernance, observabilité).

Questions bonus

Q57. [Bonus] Transactions : ACID ? (Tech Lead)

Réponse : Atomicité, Cohérence, Isolation, Durabilité.

Q58. [Bonus] Caching : où et quoi ? (Tech Lead)

Réponse : CDN/reverse proxy/app/DB; cache des lectures idempotentes, invalidation/TTL.

Q59. [Bonus] NAT : pourquoi l'utiliser ? (Tech Lead)

Réponse : Traduction d'adresses privées vers publique, économise IPv4 et masque topologie interne.

Q60. [Bonus] Adapter : quand ? (Tech Lead)

Réponse : Intégrer API legacy/tiers avec interface différente (ex. switch de paiement).