

Quizz KMP/Android

1. En Kotlin, quel est le mot clef pour déclarer une fonction ?

- ☐ (A) function
- ☐ (B) define
- ☐ (C) fun
- ☐ (D) declare

2. En Kotlin, la fin d'une instruction se termine par un point virgule (;) ?

- ☐ (T) True
- ☐ (F) False

3. En Kotlin, comment créer une variable de type numérique de valeur 5 ?

- ☐ (A) num = 5
- ☐ (B) val num = 5
- ☐ (C) int num = 5
- ☐ (D) num = 5 int
- ☐ (E) num : Int = 5

4. En Kotlin, val toto = "Hello" est modifiable ?

- ☐ (T) True
- ☐ (F) False

5. En Kotlin, var toto = "Hello" est modifiable ?

- ☐ (A) Vrai
- ☐ (B) Faux

6. En Kotlin, quel opérateur est utilisé pour fusionner 2 chaînes de caractères ?

- ☐ (A) le signe +
- ☐ (B) le signe &
- ☐ (C) le signe *
- ☐ (D) le mot clef "add"
- ☐ (E) la fonction ".plus()" de la class String

7. En Kotlin, comment peut-on créer une liste ?

- ☐ (A) `myList:List {"value1","value2","value3"}`
- ☐ (B) `listOf()` function
- ☐ (C) `mutableListOf()` function
- ☐ (D) `mylist["value1","value2","value3"]`
- ☐ (E) none of those answers

8. En Kotlin, comment créer une plage de chiffres entre 5 et 15 ?

- ☐ (A) `for (5.. 15)`
- ☐ (B) `for (5 to 15)`
- ☐ (C) `for (x in 5 to 15)`
- ☐ (D) `for (x in 5..15)`

9. En Kotlin, est-il possible d'hériter des propriétés et fonctions d'une classe depuis une autre ?

- ☐ (T) True
- ☐ (F) False

10. En Kotlin, que signifie "fname" dans le prototype de fonction suivant : `"fun myFunction(fname: String)"`

- ☐ (A) Le parametre
- ☐ (B) La valeur de retour
- ☐ (C) Le type d'objet
- ☐ (D) aucune de ces propositions

11. En Kotlin, comment écrire correctement les paramètres d'entrée d'une fonction ?

- ☐ (A) `fun sum(Int: ValueA, Int: valueB)`
- ☐ (B) `fun sum(Int ValueA, Int valueB)`
- ☐ (C) `fun sum(valueA: Int, valueB: Int)`
- ☐ (D) `fun sum(valueA Int, valueB Int)`

12. En Kotlin, une fonction qui accepte en paramètre ou en retour une fonction est appelée ...

- ☐ (A) une extension
- ☐ (B) une lambda
- ☐ (C) Fonction d'ordre supérieur
- ☐ (D) Nullable receiver

13. En Kotlin, pour qu'une valeur ou une fonction d'une class puisse être hérité par une autre class il faut ajouter le mof clef ...

- ☐ (A) `public`
- ☐ (B) `extends`
- ☐ (C) `open`
- ☐ (D) `final`

14. En Kotlin, quelle est la syntaxe correcte pour convertir une chaine de caractères "42" en un Long

- ☐ (A) `val l : Long = "42".toLong()`
- ☐ (B) `val l : Long = (Long)"42"`
- ☐ (C) `val l : Long = <Long>"42"`
- ☐ (D) `val l : Long = Long.parseLong("42")`

15. En Kotlin, what is the difference between
`var a: String? = "Kotlin"`
`var b: String = "Kotlin"`

- ☐ (A) b est en final est ne peut pas être changé
- ☐ (B) a est volatile comme en java
- ☐ (C) a est en final est ne peut pas être changé
- ☐ (D) b ne peut jamais être null

16. En Kotlin, quelle(s) sont la/les déclarations de fonction valide(s) ?

- ☐ (A) `function sum(a: Int, b: Int) : Int`
- ☐ (B) `fun sum(a: Int, b: Int): Int`
- ☐ (C) `int sum(a: Int, b: Int)`
- ☐ (D) `int sum(int a, int b)`

17. En Kotlin, que fait l'opérateur "!!" ?

- ☐ (A) Il convertit n'importe quelle valeur en type non-null and renvoie une exception si la valeur est en fait "null"
- ☐ (B) Il compare si 2 valeurs sont identiques plutôt qu'égales
- ☐ (C) il renvoie l'opérande de gauche si elle n'est pas null sinon il renvoie l'opérande de droite.
- ☐ (D) C'est l'opérateur modulo en Kotlin, similaire à % en Java

18. En Kotlin, que ne propose pas la "data class" ?

- ☐ (A) Génération automatique des fonctions `hashCode()` et `equals()`
- ☐ (B) Génération automatique de la fonction `toString()`
- ☐ (C) Conversion automatique de/vers JSON
- ☐ (D) Génération automatique de la fonction `copy(...)`, pour créer des copies d'instances

19. Quelle(s) mécanisme(s) permet aux applications Kotlin/Android de créer des tâches asynchrones (accès à une BDD, à une API Rest...) ?

- ☐ (A) une "Coroutine"
- ☐ (B) un Thread
- ☐ (C) un Fragment

20. Which of the following best describes the primary advantage of using Ktor's `HttpClient`?

- ☐ (A) Synchronous processing of HTTP requests
- ☐ (B) Integration only with Android platforms
- ☐ (C) Asynchronous and non-blocking by default
- ☐ (D) Exclusive support for RESTful APIs

21. How do you typically perform a GET request using Ktor's HttpClient?

- ☐ (A) `get()`
- ☐ (B) `sendRequest(HttpMethod.Get)`
- ☐ (C) `performGET()`
- ☐ (D) `executeRequest(Method.GET)`

22. What is the purpose of `MutableState` in Jetpack Compose?

- ☐ (A) To manage global application state
- ☐ (B) To retain state within a composable and trigger recomposition
- ☐ (C) To handle lifecycle events in composables
- ☐ (D) To store read-only state

23. Which function is used to create and manage mutable state in Jetpack Compose?

- ☐ (A) `stateOf()`
- ☐ (B) `mutableStateOf()`
- ☐ (C) `composeState()`
- ☐ (D) `createState()`

24. What is Kotlin Flows

- ☐ (A) Flows is a multithreading framework library for kotlin
- ☐ (B) Flows are built on Kotlin coroutines, supporting sequential asynchronous processing
- ☐ (C) Flows are synchronous
- ☐ (D) Flows are limited to Android development only

25. Which operator is commonly used to transform data emitted by a Kotlin Flow?

- ☐ (A) `map()`
- ☐ (B) `switchMap()`
- ☐ (C) `flatMap()`
- ☐ (D) `transform()`

26. What role does `CoroutineScope` play in managing coroutines?

- ☐ (A) It determines the Thread used for asynchronous execution thanks to Dispatcher
- ☐ (B) It defines the lifecycle of the application
- ☐ (C) It specifies the UI thread for coroutine execution

27. Le dossier "res" d'un projet Android contient ...

- ☐ (A) Toutes les classes réseaux de l'application
- ☐ (B) Toutes les ressources que mon projet utilise (images, layout xml...)
- ☐ (C) Toutes les classes kotlin (activity, adapter, fragments...)

28. Une activité est un composant d'une application android qui ...

- ☐ (A) est un écran/une interface visuelle permettant d'interagir avec un utilisateur pour réaliser une action
- ☐ (B) se lance dans le thread principal de l'application
- ☐ (C) doit être déclarée dans le manifest

29. Dans le pattern d'architecture MVC, l'activité android est...

- ☐ (A) Le modèle
- ☐ (B) Le contrôleur
- ☐ (C) La vue
- ☐ (D) Aucune de ces propositions

30. Dans le cycle de vie d'une application, quelle est la callback initiale que le système appelle lorsqu'une activité est appelée ?

- ☐ (A) onCreate()
- ☐ (B) onStart()
- ☐ (C) onResume()
- ☐ (D) onCreate(savedInstanceState: Bundle?)

31. Quels sont les dossiers principaux d'un projet Android ?

- ☐ (A) src, res, values
- ☐ (B) src, images, screen , designs
- ☐ (C) code,icon,lib,admin
- ☐ (D) none of those answers

32. Comment sont alignées les vues dans un "LinearLayout" ?

- ☐ (A) n'importe ou
- ☐ (B) dans plusieurs directions
- ☐ (C) ajustées entre la vue parente et ses vues filles
- ☐ (D) dans une direction : horizontale ou verticale

33. Une application "A" peut par défaut accéder aux activités d'une autre application "B" ?

- ☐ T True
- ☐ F False

34. Sur quel système d'exploitation (OS) Android est basé ?

- ☐ A DOS
- ☐ B Windows
- ☐ C Linux
- ☐ D macOS

35. Il faut une permission pour pouvoir créer une requête HTTP depuis mon application

- ☐ T True
- ☐ F False

36. Une application "A" peut accéder directement à la base de données d'une application "B" ?

- ☐ T True
- ☐ F False

37. Où la base de données locale spécifique à une application est stockée ?

- ☐ A Dans un espace de stockage privé dans la mémoire interne du téléphone.
- ☐ B Par défaut dans la carte SD du téléphone.
- ☐ C Je sélectionne à la création de mon application un serveur sur AWS, Azure, GCP ...
- ☐ D Dans un espace partagé à toutes les applications de la mémoire interne de mon téléphone

38. Quelle(s) mécanisme(s) permet aux applications Kotlin/Android de créer des tâches asynchrones (accès à une BDD, à une API Rest...) ?

- ☐ A une "Coroutine"
- ☐ B un Thread
- ☐ C un Fragment

39. Pour réaliser de longues opérations, je dois utiliser le Main thread ?

- ☐ T True
- ☐ F False

40. Sélectionnez des contraintes du développement mobile Android :

- ☐ **A** La sécurité
- ☐ **B** La gestion de la batterie
- ☐ **C** la gestion de l'ubiquité
- ☐ **D** L'utilisation intensive de capteurs
- ☐ **E** L'expérience tactile et la diversité des tailles d'écrans