

1. Quelle est la différence entre margin et padding dans un layout Android?

Margin est l'espace supplémentaire entourant l'extérieur d'une vue, sorte de comme une barrière invisible autour de la vue.

Padding est l'espace supplémentaire à l'intérieur d'une vue; Entre le contenu de la vue et sa frontière.

2. Qu'est-ce qu'un layout ?

Les Layouts définissent la structure visuelle pour une interface utilisateur.

3. Différence entre wrap_content et match_parent ?

Match_Parent : signifie que la vue peut être aussi grande que son parent moins le padding du parent

Wrap_Content : signifie que la vue peut être juste assez grande pour s'adapter à son propre contenu interne, en tenant compte de son propre padding

4. Java build tools ? en android?

L'automatisation de la compilation consiste à automatiser la création d'une compilation de logiciels et des processus associés, notamment: la compilation du code source de l'ordinateur en code binaire, l'emballage du code binaire et l'exécution de tests automatisés.

Java Build Tools sont principalement utilisés pour compiler et construire une image logicielle utilisable à partir de votre code source. Cette image logicielle peut être une application Web, une application de bureau, une bibliothèque pour d'autres bases de code à utiliser ou même un produit complet.

Dans un projet Android, Gradle est un système de build évolué qui permet l'automatisation de compilation, de construction, des tests, des publications et du déploiement d'un projet.

5. RelativeLayout vs LinearLayout?

LinearLayout : Layout qui repose sur une idée de boîte, c'est-à-dire que les conteneurs ou les widgets appartenant à ce Layout sont soit alignés selon une orientation horizontale ou verticale.

RelativeLayout vous permet de spécifier comment les vues enfants sont positionnées l'une par rapport à l'autre.

La position de chaque vue peut être spécifiée par rapport aux éléments de frère ou par rapport au parent.

6. Est-il possible de trouver activité sans IU dans android?

Généralement Non. La fonctionnalité principale de l'activité est d'afficher la vue à laquelle elle est rattachée et gérer les interactions utilisateur interface.

7. comment adapter les views de notre application avec différents tailles de terminaux mobiles ?

utilisation des pourcentage et des constantes de positionnement dynamiques tel que wrap_content

8. les permissions ? emplacement de declaration ? exemple

Pour assurer la sécurité du système et des utilisateurs, Android demande aux applications de demander une autorisation avant que les applications puissent utiliser certaines données et fonctionnalités système. En fonction de la sensibilité de la zone, le système peut accorder l'autorisation automatiquement ou il peut demander à l'utilisateur d'approuver la demande. Déclaration des permissions dans le fichier manifest.

Exemple : permission d'utilisation d'internet

9. layout qui met ses enfants en rangs et en colonnes

LinearLayout

10. dans quel fichier on trouve les identifiants des composants graphiques ???

Fichier R.java

11. gravity ? weight ?

layout_weight : LinearLayout prend également en charge l'attribution d'un poids à chaque enfant.

Cet attribut attribue une valeur d'importance à une vue en termes de la quantité d'espace qu'elle doit occuper sur l'écran.

Layout_gravity : Constante de gravité qu'un enfant fournit à son parent.

Définit la façon dont l'enfant doit être positionné, à la fois sur les axes X et Y, dans sa disposition jointe.

12. peut-on combiner plus d'un layout dans une vue?

Oui

13. VersionName

Représente la version de sortie du code d'application, comme il doit être montré aux utilisateurs.

14. android runtime???

Dalvik et ART sont des machines virtuelles, c'est-à-dire des émulateurs, qui permettent aux applications de tourner sur des appareils, indépendamment des différences matérielles. En d'autres termes, Dalvik et ART permettent de faire tourner les applications sur différents smartphones Android, quel que soit le modèle. Dalvik a été développé notamment pour permettre aux appareils peu puissants de faire tourner plusieurs applications simultanément.

Le code d'exécution du programme, appelé bytecode, est traduit pour pouvoir être interprété par la machine virtuelle. Il est "traduit" à la volée, selon la méthode dite **Just-In-Time** (juste à temps), à chaque lancement d'une application.

Ce que ART va changer ?

La machine virtuelle ART, ou Android Run Time, change fondamentalement la donne, notamment car elle ne fonctionnera plus en Just-In-Time, mais en **Ahead-Of-Time** (avant le temps). À la différence de Dalvik, le code fonctionnant avec la machine virtuelle est traduit lors de l'installation des applications, une seule fois. ART est déjà intégré à KitKat, mais pas

activé par défaut. La version intégrée à Android L est différente, et Android 5.0 Lollipop devrait également voir son lot de changements concernant la machine virtuelle.

15. Android est basé sur quel noyau?

Linux Kernel

16. ndk ? question recherche