

Examen pratique 2 Programmation mobile

Question 1

À faire dans l'activité de la question 1 :

1. Créer un recyclerView qui contiendra des offres d'emplois selon la méthode vue en classe.
 - a. Créez le layout pour afficher un élément de la liste qui représentera une offre d'emploi et possèdera les éléments suivants :
 - i. Le titre du poste
 1. Le titre doit avoir une taille relativement grande
 2. Le titre doit être en haut à gauche
 - ii. Une description de l'emploi
 1. Un texte en taille normal
 2. Sous le titre
 - iii. Le salaire
 1. Le salaire doit s'afficher à droite du titre
 - iv. Un bouton « Retirer »
 - b. Créer une classe objet « jobOfferModel »
 - i. La classe doit contenir les attributs : titre du poste, description et le salaire
 - c. Créez un adapter pour le recyclerView qui prend dans son constructeur une liste d'offre d'emploi
 - d. Ajouter le code nécessaire dans l'activité de la question1 pour le bon fonctionnement du recyclerView
 - i. Dans la méthode « onCreate » créez une liste d'offre d'emploi
 1. Ajouter au moins 3 emplois
 2. Utilisez cette liste dans votre recyclerView
 - e. Ajouter le code nécessaire pour que lorsque le bouton « Retirer » d'un élément de la liste est appuyé, cet élément soit retiré de la liste.
 - i. Le recyclerView doit se mettre à jour.

Question 2

À faire dans l'activité de la question 2:

1. Créez un layout pour un dialog qui doit contenir
 - a. Un message qui dit : « Pleine puissance monsieur Sulu »
 - b. Un bouton qui dit « Aye, Captain »
 - c. Un bouton qui dit « Annuler »
2. Dans l'activité de la question2, créez un bouton pour afficher votre dialog.
3. Concernant le dialog

- a. Appuyer le bouton « Aye, Captain » affiche un toast avec le contenu « Vroom »
- b. Appuyer sur le bouton « Annuler » affiche un toast avec le contenu « It's treason then »

Question 3

À faire dans l'activité de la question 3:

1. Créer une CustomView qui extend View
 - a. Reproduire le comportement dans la vidéo incluse
2. Dans votre classe CustomView, ajoutez
 - a. Une interface OnSuccessListener avec la fonction onSuccessPassword;
 - b. Une méthode setInput(String input)
 - c. Une méthode setOnSuccessListener

À savoir

- Être capable de détecter un « touch gestures » de différent type sur une View
- Être capable de créer une boîte de dialogue
- Être capable d'implémenter un RecyclerView et tout ce qui s'y rattache
- Être capable de créer une « CustomView » et d'en faire l'affichage ainsi que la programmation des comportements reliés