

D2AM

Encore l'équipe marketing

Les “marketeux” ont eu une nouvelle idée. Ils ont remarqué que certains utilisateurs avaient beaucoup trop de tâches en état “todo” alors qu’il est recommandé d’en avoir maximum 3.

Ils ont alors rédigé un magnifique tutoriel à envoyer par mail pour aider nos utilisateurs dans le besoin.

Extrayez la liste des emails d'utilisateur, possédant + de 3 tâche à l'état “todo”.

L'équipe de test

Nos superadmin testeurs ont créés plein de tâches pour effectuer différents tests, mais aimerais que vous fassiez un petit coup de nettoyage.

Supprimez toutes les tâches appartenant à des utilisateurs possédant le rôle “superadmin”



Vous pouvez faire vos tests avec UPDATE, ou garder dans un vs code un dataset prêt à insérer pour éviter de devoir refaire un dataset complet en cas d'erreur.

Nouvelle feature: les workspace

Une nouvelle demande est arrivée chez votre chef de projet.

Il vous est demandé de mettre en place une feature “workspace” permettant à nos utilisateurs de collaborer.

Voici donc les modifications demandés.

- Ajout d'une table “workspaces”

- id: PK AI (primary key, auto increment)
- title
- created_by: FK user.id

- Modification de la table “todos”

- + workspace_id PK workspace.id

Bonus: ajout d'une contrainte dans la table todos pour obliger une todo à avoir soit user_id de renseigné, soit workspace_id. Mais pas les 2 en même temps.

Ajoutez un dataset de test à cette nouvelle table.

L'application devient payante

Bon, il est temps, notre application devient payante pour nos utilisateurs qui sont dans un workspace de 3 personnes et +.

Extrayez la liste des emails des propriétaires de ces workspaces (workspaces.created_by) pour que l'équipe commerciale puisse les prévenir rapidement.

Ajoutez une colonne “forfait” à la table workspace (ENUM(“free”, “premium”))

Des problèmes de visualisation 1

Il serait plus pratique pour les gestionnaires de plusieurs workspace, de pouvoir visualiser en un seul endroit la liste des workspaces, ainsi que le nombre d'utilisateurs contenu dans ces workspace.

Créez une fonction qui liste tous les workspaces d'un utilisateur, et compte le nombre d'autres utilisateurs dans ces workspace.

ex:

title	users_count
workspace1	3
workspace2	4
workspace3	1

Des problèmes de visualisation 2

Nos utilisateurs se plaignent d'un problème d'UX. Il est impossible de savoir avant de rentrer dans un workspace combien de todos sont encore à l'état "todo". Ce qui implique de la navigation inutile.

Créez une requête SQL qui agrège le nombre de tâche restantes à effectuer, groupées par workspace.

Préparation pour la suite

Modifiez la structure de la table `todos` afin d'y ajouter une colonne `deadline` de type `DATE`.

Insérez des données de test possédant cette colonne `deadline` (par INSERT ou UPDATE).

En retard ?

Nos utilisateurs aimeraient pouvoir visualiser au même endroit toutes les tâches en retard.

Créez une requête pour récupérer toutes les tâches en retard pour un workspace donné (id).



Réfléchissez au contexte de l'application, à ce que les utilisateurs souhaitent. Des fois dans la demande tous les filtres ne sont pas énoncés clairement. A vous de faire preuve de bon sens.

Presque en retard ?

Même principe que pour l'exercice précédent, mais cette fois la demande formulée concerne les tâches qui sont bientôt expirées (1 jour avant la deadline)

Des problèmes de visualisation 3

Modifiez la requête de l'exercice "Des problèmes de visualisation 2" afin d'ajouter une colonne `overdue`

Cette nouvelle colonne prendra les valeurs suivantes en fonction de la situation:

- "overdue": si au moins une tâche est en retard
- "in time": si aucune tâche n'est en retard.

Nos nouveaux utilisateurs

Le CTO a eu une nouvelle idée, il sera pertinent à l'inscription d'un utilisateur, de lui créer un workspace associé. Ainsi que quelques tâches.

On peut se dire "ok simple, on n'a qu'à faire un INSERT et terminé". Sauf qu'imaginons que l'inscription d'un utilisateur échoue à cause de la clause UNIQUE de l'adresse email. Notre workspace sera quand même créé et rattaché à un utilisateur qui n'existe pas !! Également pour les todos.

Créez la transaction qui règle ce problème.