

TD 03 - Todo List

List View

Communication descendante de données Parent / Enfant

Objectifs du TD

- Définissez un screen **AllTasks** intégrant le composant **TasksMaster**.

Le screen **AllTasks** importe les données statiques de type **Task**, les communique au composant **TasksMaster** via son constructeur, puis **TasksMaster** crée autant de **TaskPreview** que nécessaire et transmet chaque modèle **Task** au constructeur de **TaskPreview** en tant que paramètre.

Ainsi, les données statiques transitent de façon "*descendante*" à travers différents widgets imbriqués,

- Dans le screen **AllTasks**, importez les données statiques créées dans le TD précédent :

```
import 'package:todolist/data/tasks.dart' as data;
```

data.tasks correspond à l'objet **tasks** de type **List<Task>** contenant les objets de type **Task**, créé dans le fichier `./lib/data/tasks.dart`.

- Définissez un composant **TasksMaster** permettant de gérer l'affichage de la liste des **Tasks**.

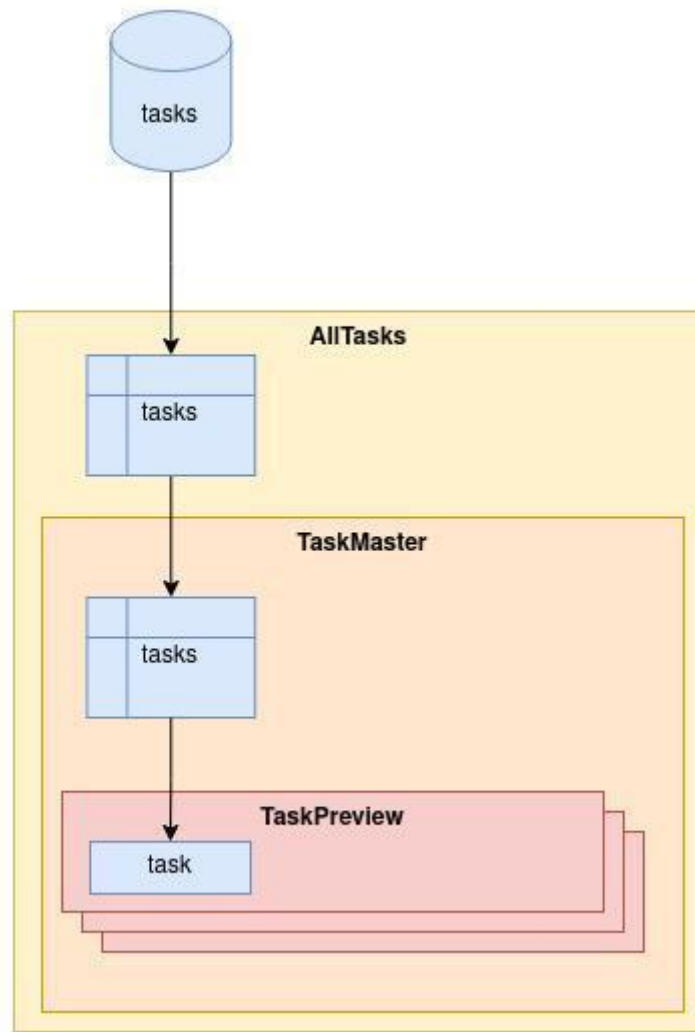
TasksMaster contiendra une **ListView** permettant d'afficher une liste verticale scrollable, composée d'autant de composants **TaskPreview** qu'il y a d'objets de type **Task** dans la source de données statiques.

Utilisez **data.tasks** en tant que paramètre de l'objet **TasksMaster**.

- Définissez un composant **TaskPreview** permettant de gérer l'affichage partiel de 1 **Task**. Ce composant doit recevoir en paramètre de son constructeur, une donnée de type **Task**, afin d'en afficher le contenu et le statut (attributs *content* et *completed*),

Développement Mobile

- Dans le composant TaskPreview, adaptez un élément graphique de votre choix (ex: couleur d'arrière-plan, icône...), afin de distinguer visuellement les Tasks dont la valeur de completed est true ou false.



Communication de données descendante "Parent / Enfant"