

# Corso di Laboratorio di Programmazione per Sistemi Mobile e Tablet

Docente: Dr. Mauro Dragoni

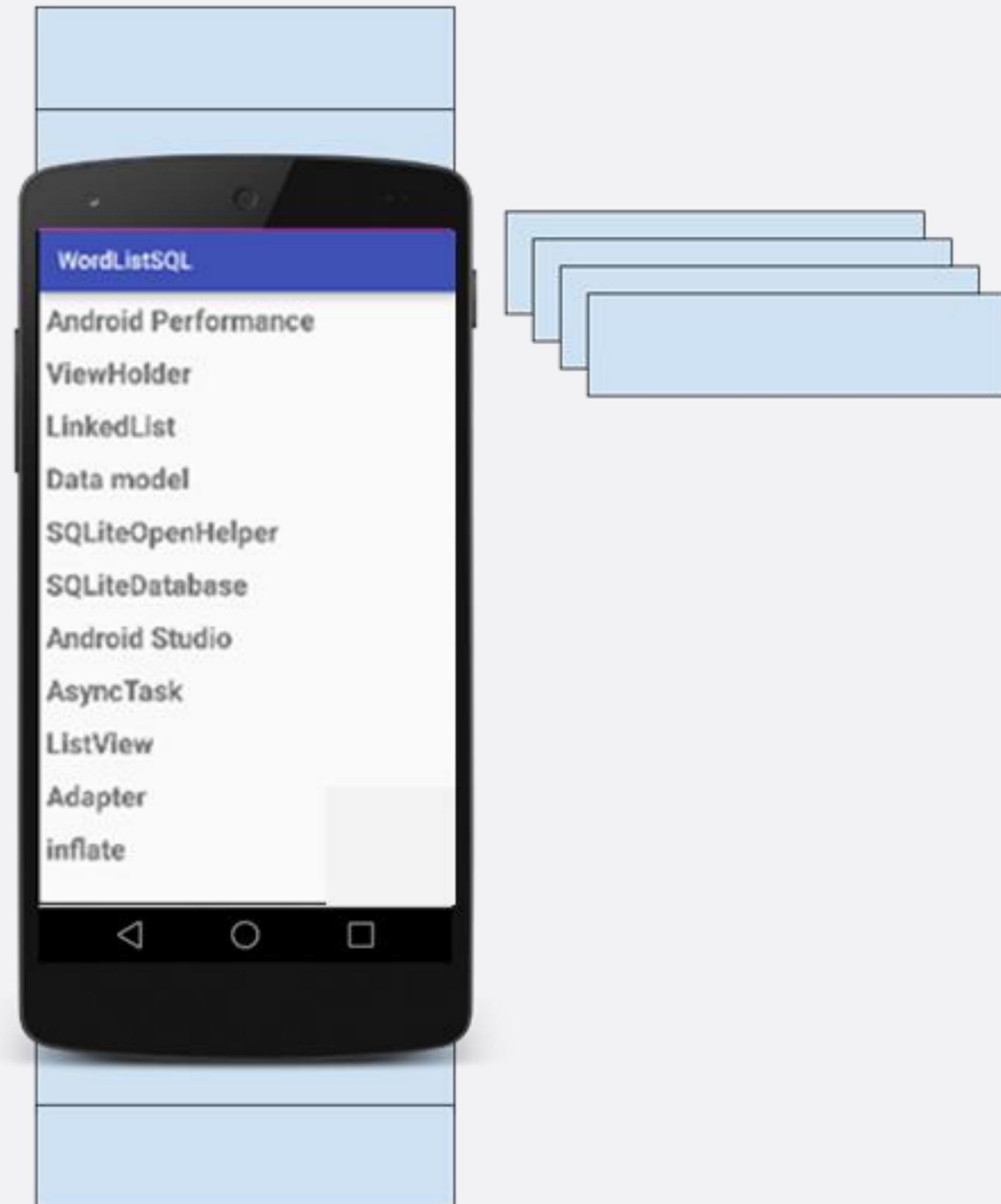
Dipartimento di Ingegneria e Scienze dell'Informazione  
Anno Accademico 2021/2022

# in questa lezione...

- RecyclerView
  - Componenti
  - Implementazione

# RecyclerView

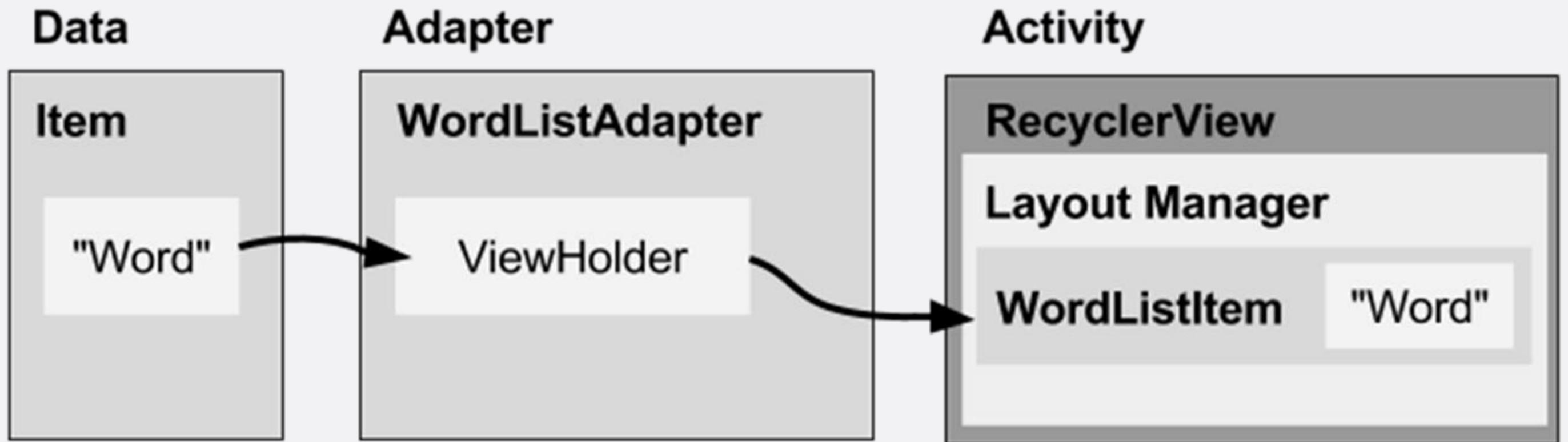
- RecyclerView e' un contenitore scorrevole per la visualizzazione dei grandi datasets.
- Principali caratteristiche:
  - Efficiente
  - Usa e ri-usa un numero limitato di View elements.
  - Aggiornamento in real-time molto veloce rispetto agli eventi.



# RecyclerView - ingredienti

- Dati
- RecyclerView scrolling list per visualizzare ciascun elemento.
- Layout da associare a ciascun elemento della lista.
- Layout manager per gestire tutti gli elementi della UI.
- Adapter per collegare i dati con la RecyclerView.
- ViewHolder contenente le informazioni per visualizzare ciascun item.

# RecyclerView - ingredienti



# RecyclerView - LayoutManager

- Ogni ViewGroup ha un layout manager.
- E' utilizzato per posizionare i View items dentro una RecyclerView.
- Ri-utilizza i View items che non sono piu' visibili agli utenti.
- Built-in layout managers:
  - LinearLayoutManager
  - GridLayoutManager
  - StaggeredGridLayoutManager
- Estendono RecyclerView.LayoutManager

# RecyclerView - Adapter

- Permette ad interfacce incompatibili tra loro di lavorare assieme.
- Esempio: prelevare dei dati da un database e preparare le stringhe da mostrare all'interno di una View.
- Giocano il ruolo di intermediario tra i dati e le View.
- Gestiscono la creazione, aggiornamento, aggiunta, e cancellazione dei dati presenti nelle View.
- RecyclerView.Adapter

# RecyclerView - ViewHolder

- Sfruttato dall'adapter per preparare il popolamento di una View con il corrispondente dato da contenere.
- Layout specificato in un file XML apposito.
- Può contenere elementi cliccabili.
- È posizionato dal LayoutManager
- RecyclerView.ViewHolder

# RecyclerView – gestire gli aggiornamenti

- Reinizializzare l'oggetto contenente i dati non funziona!  
es.: `contacts = Contact.createContactsList(5);`
- Usare i metodi dedicati:

Metodo	Descrizione
<code>notifyItemChanged(int pos)</code>	Notifica che l'item di quella posizione e' stato modificato.
<code>notifyItemInserted(int pos)</code>	Notifica che l'item in quella posizione e' un nuovo elemento.
<code>notifyItemRemoved(int pos)</code>	Notifica che l'item in quella posizione e' stato rimosso.
<code>notifyDataSetChanged()</code>	Notifica che il dataset e' cambiato.

```
contacts.add(0, new Contact("Barney", true));  
adapter.notifyItemInserted(0);
```

# | Approfondimenti

Sul sito ufficiale:

- [RecyclerView](#)