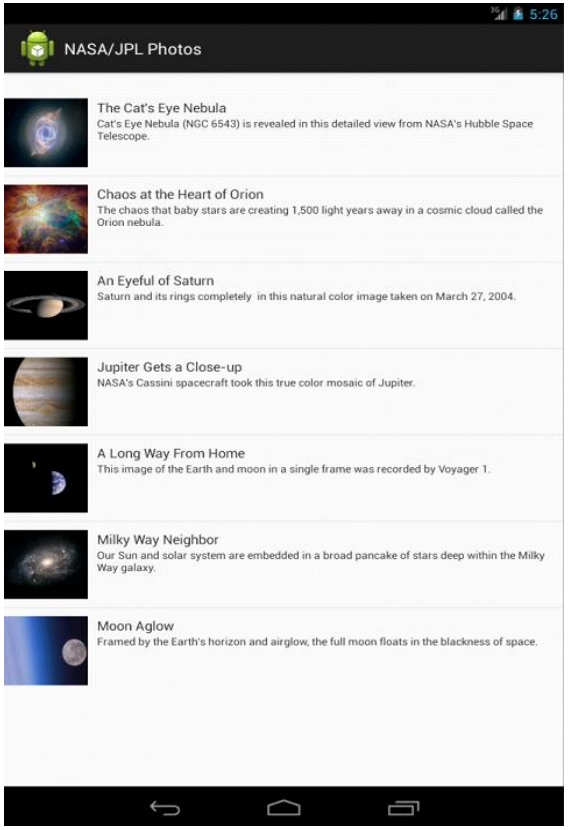
# P82 DÉVELOPPEMENT D'APPLICATIONS POUR MOBILE - ANDROID

Composants : ListView et ListAdapter

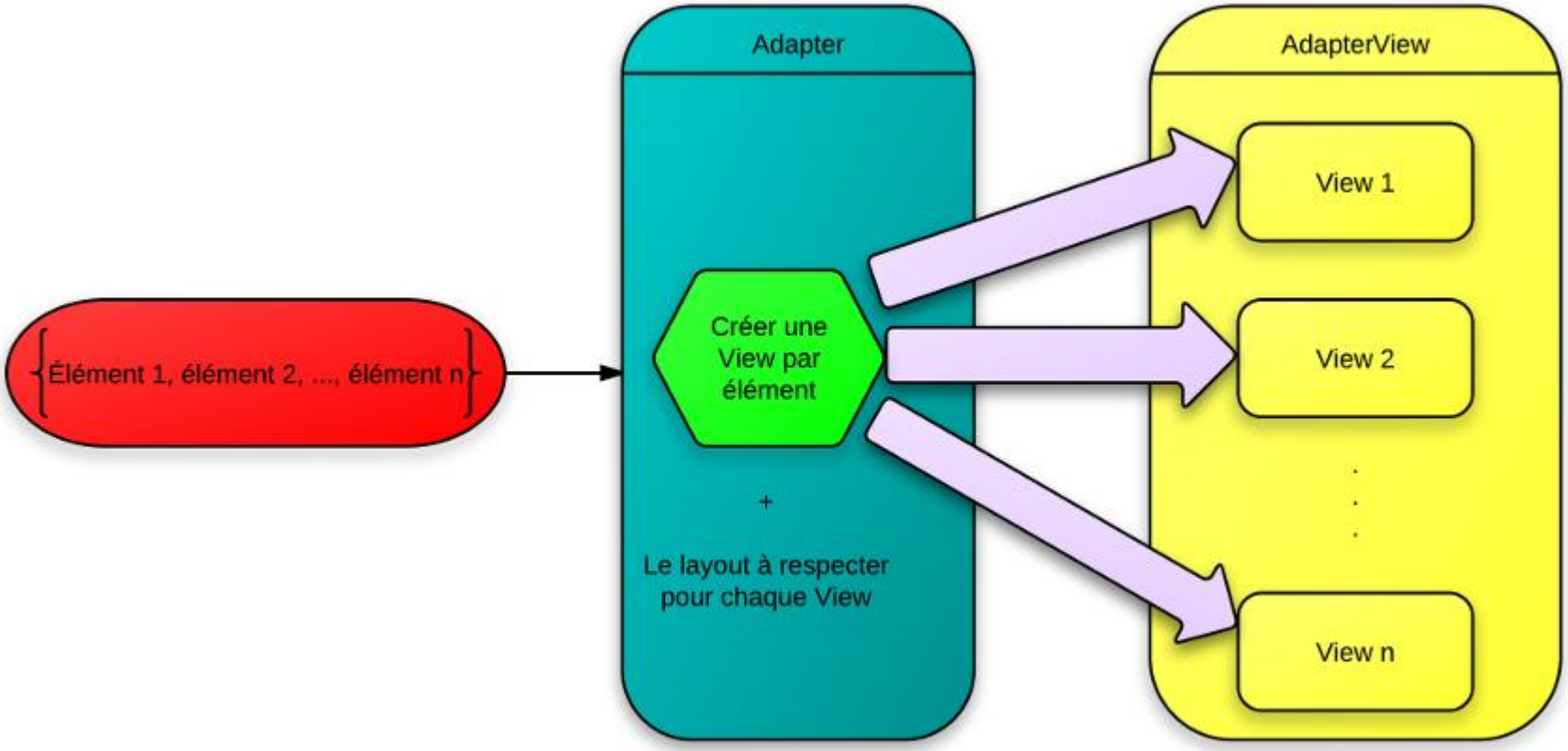
# ListView - Adapter

- ViewGroup qui a pour but d'afficher une liste d'items avec défilement si nécessaire.




- On utilise un Adapter pour mettre les items dans la ListView
- Données fournies à Adapter
  - Simple Array de données
  - Depuis une base de données
- Adapter va convertir les données en View et les ajouter dans une liste

# ListView - Adapter



# Adapter

- Dans l'utilisation 3 Adapters principaux
  - ArrayAdapter

    Information simple
  - SimpleAdapter

    Peut avoir plusieurs informations par item
  - CursorAdapter

    Pour info d'une base de données

# ArrayAdapter

- public ArrayAdapter (Context contexte, int idLayout, List<T> objects).
  - Contexte
  - Le layout que va prendre la liste (déjà existant mais pourrait être créé)
  - Les données à afficher (Attend un Object donc Integer[], Double[]….)

# ArrayAdapter exemple

```
<string-array name="listEtudiant">
        <item>Oumou Dily</item>
        <item>Omar</item>
        <item>Jorel</item>
        <item>Ruben</item>
        <item>Nicolas</item>
        <item>Moussa</item>
        <item>Meng</item>
    </string-array>
```

# ArrayAdapter exemple

- String[] etudiant = getResources().getStringArray(R.array.*listEtudiant* );
- ArrayAdapter lapdapt = new ArrayAdapter<String>(this, android.R.layout.*simple_list_item_1,etudiant*);
  - android.R.layout.*simple_list_item_1 : layout fournit par android*
- lv.setAdapter(lapdapt);

à la place de setContentView

Lv est un ListView provenant d un layout

# ArrayAdapter

- Meilleur solution utiliser la class ListActivity
  - Contient déjà un ListView
- Mais dois absolument utiliser un ListAdapater à la place d'un ArrayAdapter
- Méthode à redéfinir
  - protected void onListItemClick(ListView l, View v, int position, long id)
    - Position est la position de l item cliqué

# SimpleAdapter

- On va pouvoir mettre plusieurs lignes dans un item
- public SimpleAdapter (Context context, List<? ext ends Map<String, ?>> data, int resource)
  - Data : est un Map ou chaque entrée correspond a une ligne
  - Ressource : comment on va afficher les informations
- https://openclassrooms.com/courses/creez-des-applications-pour-android/des-widgets-plus-avances-et-des-boites-de-dialogue

# Adapter personnalisé

- Premiere chose se faire le layout en XML

```
<?xml version="1.0" encoding="utf-8"?>
    <LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">


        <ImageView android:id="@+id/imgEtu"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            />
        <TextView android:id="@+id/lblNom"
            android:layout_marginLeft="10dp"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            />
    </LinearLayout>
```

# Se créer un entités Etudiant

```
public class Etudiant {
        String nom;
        int img;

        public Etudiant(String nom, int img){
            this.nom = nom;
            this.img = img;
        }
// getter setter
}
```

# Redéfinir ArrayAdapter<Etudiant>

- Méthode:

public View getView(int position, View convertView, ViewGroup parent)

appelée pour chaque item du ArrayAdapter

-> création de la vue

Ce qui sera afficher sera convertView

# Constructeur

```
public class EtudiantListe extends ArrayAdapter<Etudiant> {
        Context ctx;
        int idLayout;
        int nbEtu;


public EtudiantListe(Context context, int resource, List<Etudiant> le) {
            super(context, 0);
            ctx = context;
             idLayout = ressource;

    //ajoute les étudiants dans l ArrayAdapteur
            for(Etudiant e:le){
                this.add(e);
                nbEtu++;
            }
        }
    }
```

# Redéfinir la méthode getView()

```
@Override
    public View getView(int position, View convertView, ViewGroup parent) {
     // récupère l étudiant depuis  l'adapater
       Etudiant e = getItem(position);

    // on va redéfinir la vue de l item
        if(convertView == null){
            LayoutInflater li = LayoutInflater.from(ctx);
            convertView = li.inflate(R.layout.list_etudiant,null);
        }

    // redéfinir les éléments dans le convertView
        ImageView img = (ImageView) convertView.findViewById(R.id.imgEtu);
            img.setImageResource(e.getImg());
        TextView tv = (TextView) convertView.findViewById(R.id.lblNom);
            tv.setText(e.getNom());

        return convertView;
    }
```

# Utiliser notre ListView avec les items selon notre layout

```
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        lv = (ListView) findViewById(R.id.etudiantListView);


        String[] etudiant= getResources().getStringArray(R.array.listEtudiant);

        ArrayList<Etudiant> listEtu = new ArrayList<>();
        for (String e : etudiant){
            Etudiant etu = new Etudiant(e,R.drawable.etudiantVide);
            listEtu.add(etu);
        }

        EtudiantListe etu = new
    EtudiantListe(this,R.layout.list_etudiant,listEtu);

        lv.setAdapter(etu);
    }
```

# Liste des étudiants

- Nom étudiant et nom de l image doivent être les mêmes

```
private List<Etudiant> getEtudiant(){
        String[] etu = getResources().getStringArray(R.array.listEtudiant);
        List<Etudiant> listEtu = new ArrayList<>();
        Field fs[] =R.drawable.class.getFields();

        int cpt = 0;
            // permet de faire une réflexion sur la class R
        for (Field f : fs){
            if(f.getName() == etu[cpt]){

                try {
                    Etudiant e = new Etudiant(etu[cpt], f.getInt(null));
                    listEtu.add(e);
                    cpt++;
                } catch (IllegalAccessException e1) {
                    e1.printStackTrace();
                }
            }
        }
     return  listEtu;
    }
```