# Cheat Sheet: Customized ListView

## Steps:

1. Create class to represent the data entity
2. Create row layout XML file
3. Drag ListView to activity layout
4. Create ArrayAdapter subclass
5. Instantiate ArrayAdapter subclass in ListActivity
6. SetListAdapter() in ListActivity

## Detailed Steps:

### Create class to represent the data entity

|  |  |
| --- | --- |
| 1 2 3 4 5 6  7 8 9 10 11 12 13 14  15  16  17  18 | // Change ClassName to suitable class like “Place”  public class ClassName {  // Create the attributes for your class  private String attr1;  private boolean attr2;  // Create the constructor to give your attributes values  public ClassName(String attr1, boolean attr2) {  this.attr1 = attr1;  this.attr2 = attr2;  }  // Create the setters for all your attributes  public String getAttr1() {  return attr1;  }  public boolean isAttr2() {  return attr2;  }  } |

### Create row layout XML file

This row blueprint is what every row is built from.

|  |  |  |
| --- | --- | --- |
|  |  |  |

### Drag ListView to activity layout

Remember to change its ID to something helpful @+id/lvFood

### Create ArrayAdapter subclass

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39 | public class ClassNameArrayAdapter extends ArrayAdapter<ClassName>{  // Create ArrayList of objects  private ArrayList<ClassName> objects;  // To hold the context object  private Context context;  // Create the UI objects to hold the UI elements in row layout  private TextView tv1;    public ClassNameArrayAdapter(Context context, int resource,  ArrayList<ClassName> objects) {  super(context, resource, objects);  // Store the ArrayList of objects passed to this adapter  this.objects = objects;  // Store Context object as we would need to use it later  this.context = context;  }  // getView() is called every time for every row  @Override  public View getView(int pos, View convertView, ViewGroup parent)  {  // "inflate" the XML file into a View object  LayoutInflater inflater = (LayoutInflater) context  .getSystemService(Context.*LAYOUT\_INFLATER\_SERVICE*);  // Change R.layout.rowxyz if file is rowxyz.xml .  View rowView = inflater.inflate(R.layout.*row*, parent,  false);  // Get the TextView object  tv1 = (TextView)  rowView.findViewById(R.id.*textView1*);  // Parameter "pos" is the index of the  // row ListView is requesting.  // We get back the object at the same index.  ClassName object = objects.get(pos);  // Set the TextView to show the object info  tv1.setText(object.getAttr1());  // Return this row that is being populated.  return rowView;  }  } |

### Instantiate ArrayAdapter subclass in Activity

### setListAdapter() in ListActivity

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11  12 | @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*main*);  // objects can be retrieved from database  ArrayList<ClassName> objects = ...;  // Create the ArrayAdapter object.  ClassNameArrayAdapter adapter =  new ClassNameArrayAdapter(this, R.layout.*row*, objects);  ListView lv = (ListView) findViewById(R.id.lvFood);  lv.setAdapter(adapter);  } |