

# Widget/Selection Widget

ListView

Spinner

GridView

AutoComplete  
TextView

# Widget

(when to use)

Các Radio Button và Check Button  
được dùng để chọn lựa trong một tập  
nhỏ các lựa chọn

Khi tập lựa chọn lớn hơn, người ta thường  
dùng các view/widget như Listbox,  
ComboBox, DropDown List, Picture Gallery

# Widget

Data adapter

Array adapter

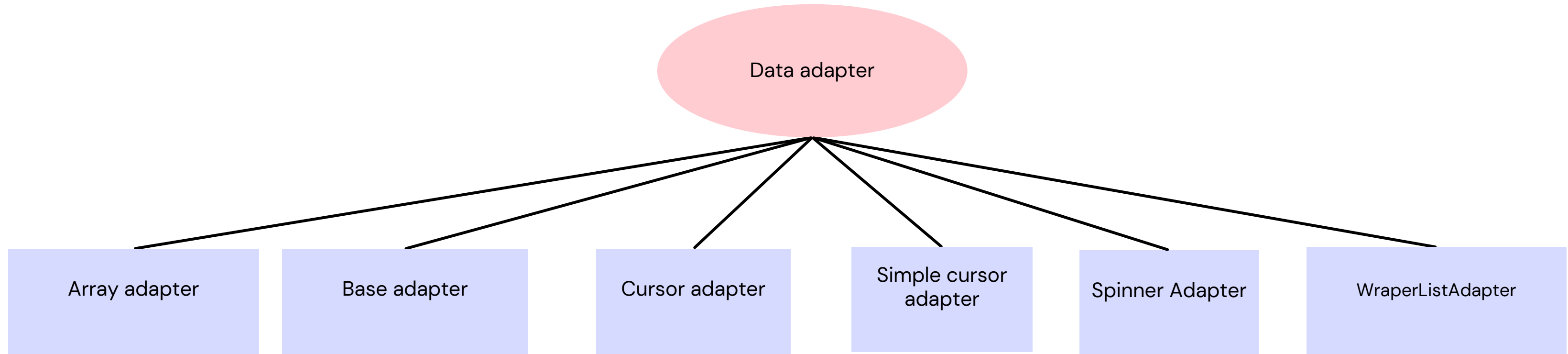
Base adapter

Cursor adapter

Simple cursor  
adapter

Spinner Adapter

WraperListAdapter



# Spinner

## Tính chất

- Trong Android, Spinner tương đương với DropDown
- ta có thể cung cấp một Adapter để kết nối dữ liệu danh sách lựa chọn bằng cách sử dụng setAdapter()

## Sử dụng listener

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    selection = (TextView) findViewById(R.id.selection);
    Spinner spin = (Spinner) findViewById(R.id.spinner);
    spin.setOnItemSelectedListener(this);
    // set a clickable right push-button comboBox to show items
    ArrayAdapter<String> aa = new ArrayAdapter<String>(
        this, android.R.layout.simple_spinner_item, items);
    // provide a particular design for the drop-down lines
    aa.setDropDownViewResource(
        android.R.layout.simple_spinner_dropdown_item);
    // associate GUI spinner and adapter
    spin.setAdapter(aa);
}

////////////////////////////////////

public void onItemSelected(
    AdapterView<?> parent, View v, int position, long id) {
    selection.setText(items[position]);
}

public void onNothingSelected(AdapterView<?> parent) {
    selection.setText("");
}
}
```

# GridView

Tính chất	Thuộc tính
<ul style="list-style-type: none"><li>GridView là một ViewGroup hiển thị thông tin theo 2 chiều dưới dạng lưới có thể cuộn được</li><li>Các mục trong danh mục dạng lưới có thể tự động chèn vào bằng cách sử dụng ListAdapter</li></ul>	<ul style="list-style-type: none"><li>android:id : ID duy nhất của layout</li><li>android:columnWidth: Xác định độ rộng cho một cột</li><li>android:gravity: Xác định gravity bên trong mỗi ô. Giá trị có thể là top, bottom, left, right, center, center_vertical, center_horizontal ...</li><li>android:horizontalSpacing: Xác định khoảng cách mặc định theo chiều ngang giữa các cột. Có thể là px, dp, sp, in, hoặc mm</li><li>android:numColumns: Xác định có bao nhiêu cột để hiển thị. Có thể là một giá trị nguyên, như "100" hoặc auto_fit nghĩa là hiển thị bao nhiêu cột để có thể điền vào không gian có sẵn</li><li>android:stretchMode: Xác định cách các cột nên dẫn ra để điền vào không gian còn trống. Nó phải là một trong các giá trị: -none: Vô hiệu hóa việc mở rộng, spacingWidth: Khoảng cách giữa mỗi cột được dẫn ra, columnWidth: Mỗi cột được kéo dẫn ra như nhau, spacingWidthUniform: Khoảng cách kéo dẫn ra là đồng nhất</li><li>android:verticalSpacing: Định nghĩa khoảng cách mặc định theo chiều dọc giữa các hàng.</li></ul>

# AutoCompleteTextView

## Tính chất

- AutoCompleteTextView hỗ trợ cho người dùng những gợi ý liên quan khi nhập vào trường EditText. Những gợi ý đó sẽ được hiển thị trong một menu thả xuống từ đó người dùng có thể chọn một mục để thay thế cho nội dung của mình vừa nhập vào.

## Sử dụng listener

```
AutoCompleteTextView(R.layout.activity_main,
    selection = (TextView)
findViewById(R.id.selection);
    edit = (AutoCompleteTextView)
findViewById(R.id.edit);
    edit.addTextChangedListener(this);
    edit.setAdapter(new ArrayAdapter<String>(this,
android.R.layout.simple_dropdown_item_1line, items));
}
    public void onTextChanged(CharSequence s, int start,
int before, int count) {
        selection.setText(edit.getText());
    }
    public void beforeTextChanged(CharSequence s, int
start,
                                int count, int after) {
    }
    public void afterTextChanged(Editable s) {
    }
```