|  |  |
| --- | --- |
| Materi | Nilai |
| Pembelajaran AndroidStudio Video ke  29 - 35  (Tugas semester 5 #6) | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Tools:listitem adalah perintah yang digunakan untuk memanggil atau menjalankan resource file (file XML) untuk pengaturan isi list ke dalam activity\_main.xml yang biasanya ditulis di dalam recyclerview.

Cth :

<androidx.recyclerview.widget.RecyclerView  
 tools:listitem="@layout/item\_barang"  
/>

Artinya, di dalam recycler view terdapat perintah tools:listitem yang memanggil file XML item\_barang yang ada pada folder layout.

Di dalam file item\_barang, berisi pengaturan untuk penampilan item pada recyclerview.

1. isEmpty() adalah perintah yang digunakan untuk mengecek apakah komponen yang diperiksa kosong atau tidak.
2. Equals() adalah perintah yang digunakan untuk membaca objek apakah memiliki nilai sesuai dengan nilai yang dituliskan pada equals() atau tidak.
3. Mengecek nilai pada barang, stok, dan harga kosong atau tidak menggunakan perintah isEmpty().

Cth :

if (barang.isEmpty()|| stok.isEmpty() || harga.isEmpty()){  
 pesan("Data kosong, anda wajib mengisi semua kolom!");  
}

Artinnya, terdapat pengujian menggunakan if, mengecek apakah variabel barang, stok, dan harga kosong atau tidak menggunakan perintah isEmpty().

Apabila variabel kosong dan tidak ada nilai, maka akan menjalankan function pesan, dengan isi pesan Data kosong, anda wajib mengisi semua kolom.

1. Apabila nilai yang dicek menggunakan isEmpty tidak kosong, maka akan menjalankan else yang akan membaca isi dari textview menggunakan equals().

Cth :

else {  
 if (pilihan.equals("insert")){  
 pesan("Insert Data Berhasil");  
 }else {  
 pesan("Update Data");  
 }  
}

Artinya, apabila variabel barang, stok, dan harga tidak kosong, maka akan menjalankan else, yang berisi pengujian lagi, yaitu apakah isi dari variabel pilihan adalah insert, apabila isi nya adalah insert, maka akan menampilkan pesan Insert Data Berhasil. Tetapi jika nilai dari pilihan tidak sama dengan equals(“insert”), maka akan menampilkan pesan Update Data.

1. execSQL() adalah fungsi yang digunakan untuk mengeksekusi code pada SQL. Jadi sebelum memasukkan tabel ke dalam android, maka kita harus membuat function runSQL dimana functionnya berisi execSQL yang digunakan untuk mengeksekusi code.
2. INSERT adalah pernyataan yang digunakan untuk menambahkan data. untuk menambahkan data hanya dengan menuliskan seperti ini,

INSERT INTO table\_name (column1, column2, column3, ...)  
VALUES (value1, value2, value3, ...);

1. Memasukkan data / menginsertkan data pada tblbarang ke dalam database dengan menggunakan perintah INSERT INTO.

Cth :

String sql = "INSERT INTO tblbarang (barang, stok, harga) VALUES ('"+barang+"', "+stok+", "+harga+")";  
db.runSQL(sql);  
  
pesan("Insert Data Berhasil");

Artinya, terdapat variabel sql yang berisi perintah memasukkan data (INSERT INTO) ke dalam tblbarang dengan kolom yang dimasukkan adalah barang, stok, dan harga. Lalu, values yang diambil dari variabel barang, variabel stok, dan variabel harga.

Setelah melakukan pengaturan INSERT, maka data akan dimasukkan ke dalam sql dengan menggunakan function runSQL yang menjalankan sql. Dimana db adalah variabel yang dibuat dengan memanggil class Database dan di dalam Database terdapat function runSQL untuk mengeksekusi perintah yang dimasukkan.

Kemudian, apabila berhasil untuk memasukkan data ke dalam database, maka akan menjalankan function pesan di dalam nya berisi toast yang menampilkan pesan pop up, dengan teks Insert Data Berhasil.

1. Cara penulisan values tergantung pada tipe data dari setiap kolom pada database.
2. Penulisan values apabila kolom pada database bernilai string adalah dengan : ‘ ”+variabel pengisi+” ’.

Cth :

'"+barang+"'

1. Penulisan values apabila kolom pada database bernilai bukan teks, seperti real atau integer adalah dengan : “+variabel pengisi+”.

Cth :

"+stok+", "+harga+"

1. Toast dapat digunakan untuk menampilkan apakah sebuah pengujian tersebut berhasil berjalan atau tidak.
2. Cursor digunakan untuk menyajikan hasil dari SQLiteDatabase dalam bentuk format tabel yang menyerupai database SQL. Di dalam class cursor terdapat metode yang digunakan untuk menggerakkan cursor melalui struktur data dan metode untuk mendapatkan data dari setiap kolom dan barisnya.
3. rawQuery() adalah function yang digunakan untuk menjalankan sql yang dikirimkan ke database dan mengembalikan nilai cursor.
4. Di dalam parameter pada rawQuery() berisi 2 parameter. Parameter yang pertama berisi string query dan parameter kedua berisi argument.
5. SELECT adalah pernyataan atau statement yang digunakan untuk memilih data dari database.
6. Statement SELECT bisa dituliskan menggunakan 2 cara, yaitu select all dan select sesuai dengan kolomnya,

Untuk memilih data menggunakan select all hanya dengan menuliskan seperti ini,

SELECT \* from nama\_tabel

Untuk memilih data sesuai dengan kolom hanya dengan menuliskan seperti ini,

SELECT column1, column2, ...  
FROM table\_name;

1. ORDER BY adalah perintah yang digunakan untuk mengurutkan data sesuai dengan kolom yang dipilih.
2. ASC (ascending) adalah perintah untuk mengurutkan query dari nilai terkecil ke nilai terbesar. Apabila tidak terdapat keterangan memilih urutan secara ASC atau DESC, maka secara otomatis tampilkan yang akan ditampilkan adalah menggunakan urutan ASC.
3. DESC (descending) adalah perintah untuk mengurutkan query dari nilai terbesar ke nilai yang terkecil.
4. Menampilkan data semua kolom dan menggunakan perintah select di dalam sqlitebrowser.

Cth :

**SELECT \* FROM tblbarang ORDER BY stok ASC**

Artinya, terdapat perintah SELECT dengan memilih semua data yang ada pada tblbarang dengan data yang diurutkan adalah dari nilai query terkecil ke terbesar (ASC).

1. Menampilkan data sesuai kolom yang dipilih dan menampilkannya menggunakan perintah select.

Cth :

SELECT barang FROM tblbarang ORDER BY barang DESC

Artinya, memilih data pada kolom barang menggunakan perintah SELECT yang ada pada tblbarang dengan data yang diurutkan adalah dari kolom barang dengan urutan dari nilai terbesar ke nilai yang terkecil (DESC).

1. getCount() adalah fungsi yang digunakan untuk mengembalikan jumlah data dimana getCount() ini diperlukan untuk mengetahui jumlah data yang akan ditampilkan.
2. Viewholder berisi objek dimana setiap item yang ditampilkan pada recyclerview adalah objek ViewHolder. Kita perlu menyediakan layout XML untuk setiap item yang akan ditampilkan dimana semua komponen yang ada di layout XML ini akan digunakan di objek ViewHolder.
3. Adapter digunakan sebagai perantara untuk menghubungkan data dengan tampilan. Adapter menerima atau mengambil data, melakukan pekerjaan yang diperlukan agar data dapat ditampilkan.
4. Setiap viewholder menampung satu rangkaian data dan adapter menambahkan data ke viewholder untuk ditampilkan oleh pengelola layout.
5. Recyclerview adalah komponen dasar yang digunakan untuk menampilkan data berupa list dan dapat digunakan untuk menampung ribuan bahkan ratusan ribu data tanpa mengalami lag.
6. Recyclerview memiliki 3 komponen utama, yaitu layout, viewholder, dan adapter.
7. Jadi alur untuk membuat dan menampilkan recycler view adalah, mengisi data > data dimasukkan ke dalam adapter > membuat adapter > isi dari adapter akan dimasukkan ke dalam viewholder > ditampilkan ke dalam layout.
8. Model pada android studio digunakan untuk menghubungkan database dan adapter. Untuk masuk ke adapter harus membuat model dulu agar bisa terhubung.
9. Constructor adalah method / function khusus yang akan dijalankan ketika pembuatan objek dan digunakan untuk mempersiapkan data untuk objek.
10. Set() adalah fungsi yang digunakan untuk mengatur / mengisi data yang akan dikirimkan.
11. Get() adalah fungsi yang digunakan untuk mengambil nilai data.
12. Nama dari method setter dan getter pada android studio harus diawali dengan set untuk setter dan diawali dengan get untuk getter.
13. Cara untuk membuat method setter dan getter adalah, klik kanan > generate > setter and getter > pilih variabel apa yang akan ditambahkan method setter and getter > finish.
14. itemView adalah tag yang digunakan ketika pemanggilan item menggunakan RecyclerView.
15. Mendeklarasikan variabel activity yang tidak diketahui yang mana yang akan digunakan dapat menggunakan variabel context yang sudah disediakan oleh android studio. Jadi ketika activity dipanggil, maka akan otomatis ditambahkan.
16. Membuat private variabel untuk activity apa yang dipanggil dan data apa yang dimasukkan, dan membuat constructor lalu private variabel dimasukkan ke dalam constructor. Jadi, ketika Barang Adapter dijalankan, maka dia akan mencari siapa yang menggunakan dan data apa yang dimasukkan, dimana data – data yang dibutuhkan disimpan ke dalam constructor.

Cth :

public class BarangAdapter extends RecyclerView.Adapter<BarangAdapter.ViewHolder> {  
  
 Context context; -> Variabel siapa yang akan menggunakan  
 List<Barang> barangList; -> Variabel data apa yang dimasukkan  
  
 public BarangAdapter(Context context, List<Barang> barangList) {  
 this.context = context;  
 this.barangList = barangList;  
 }

Artinya, ketika Adapter BarangAdapter dijalankan, maka adapter akan membutuhkan 2 data, yaitu siapa yang akan menggunakan dimana data yang disimoan ke dalam variabel context dan data apa yang dimasukkan disimpan ke dalam variabel barangList.

Setelah itu, variabel context dan barangList disimpan ke dalam constructor BarangAdapter.

Jadi, ketika BarangAdapter berjalan, maka constructor akan memanggil 2 data yang sudah disimpan ke dalam constructor.

1. Function onCreateVIewHolder digunakan untuk membuat objek viewholder dan akan menentukan layout XML yang akan digunakan untuk setiap item.
2. Function onBindViewHolder digunakan untuk menghubungkan data yang ada dengan objek ViewHolder.
3. Function getItemCount digunakan untuk melihat / menghitung ada berapa banyak data pada list di dalam barangList. Dan untuk menampilkan jumlah barang di dalam getItemCount diberi fungsi size() setelah barangList dipanggil.
4. LayoutInflater adalah library yang digunakan untuk memasukkan dan mengconvert semua data file xml yang ada ke dalam recyclerview.
5. getContext() digunakan untuk mengambil nilai dari context yang akan dipanggil saat proses pemanggilan layout.
6. Inflate() adalah fungsi dari LayoutInflater yang digunakan untuk mengisi view baru dari spesifik xml resource. Parameter dari inflate() ini berisi resource, root, attached to root.
7. Resource berisi id dari xml layout resource yang akan di load.
8. Root breisi view parent yang akan dipanggil.
9. Attachtoroot berisi jika bernilai salah, root hanya digunakan untuk membuat subclass yang sesuai dengan LayoutParams untuk root di dalam view pada xml.
10. setLayoutManager() adalah perintah yang digunakan untuk mengatur layoutmanager dari recyclerview yang akan digunakan.
11. LinearLayoutManager() adalah perintah di dalam LayoutManager yang digunakan untuk membuat vertical linearlayoutmanager.
12. setHasFixedSize() adalah perintah yang digunakan untuk membuat ukuran dari recyclerview tidak berubah. Jadi fungsi ini akan memastikan bahwa ukuran akan tetap seperti yang dibuat tidak berubah walaupun ada item yang ditambahkan atau item yang dihapus.

Cth :

rcvbarang.setHasFixedSize(true);

Artinya, ukuran di rcvbarang akan tetap sama walaupun isi item ada penambahan atau pengurangan, karena menggunakan fungsi setHasFixed dengan nilai true.

1. notifyDataSetChanged() adalah perintah yang digunakan untuk menampilkan data baru ketika dilakukan perubahan.
2. setAdapter() adalah perintah yang digunakan untuk mengatur adapter apa yang dijalankan untuk menyimpan data yang diambil.
3. moveToFirst() adalah perintah yang digunakan untuk menggerakkan cursor ke dalam pertama.
4. moveToNext() adalah perintah yang digunakan untuk menggerakkan cursor ke data berikutnya.
5. moveToLast() adalah peirntah yang digunakan untuk menggerakkan cursor ke data terakhir.
6. moveToPrevious() adalah perintah yang digunakan untuk menggerakkan cursor ke data sebelumnya.
7. Clear() adalah fungsi yang digunakan untuk menghapus data pada queue.
8. Popup menu adalah aksi – aksi yang dipasangkan pada sebuah view dan dapat diakses oleh user dengan mengklik view tersebut.
9. onMenuItemClick adalah fungsi yang digunakan untuk menjalankan aksi pada saat menu diklik, dimana di dalam function ini menggunakan statement switch untuk menentukan kejadian yang ada pada masing – masing opsi dengan mendapatkan id pada setiap menu item.
10. getItemId() adalah perintah yang digunakan untuk mendapatkan id dari item.
11. Show() adalah perintah yang digunakan untuk menampilkan data yang sudah dibuat.
12. Membuat popup menu dan menampilkan sesuatu ketika setiap item menu di klik.

Cth :

ublic void onClick(View v) {  
 PopupMenu popupMenu = new PopupMenu(context, viewHolder.tvmenu);  
 popupMenu.inflate(R.menu.*menu\_item*);  
  
 popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {  
 @Override

public boolean onMenuItemClick(MenuItem item) {  
 switch (item.getItemId()){  
 case R.id.*ubah*:  
 Toast.*makeText*(context, "Ubah Data", Toast.*LENGTH\_SHORT*).show();  
 break;  
  
 case R.id.*hapus*:  
 Toast.*makeText*(context, "Hapus Data", Toast.*LENGTH\_SHORT*).show();  
 break;  
 }  
  
 return false;  
 }  
 });  
  
 popupMenu.show();  
}

Artinya, apabila tvmenu di klik, maka akan menampilkan popup menu dengan membuat constructor yang berisi context -> popup menu dijalankan dimana, viewHolder.tvmenu -> tempat dimana popup menu ditampilkan.

Lalu, popupmenu mengambil nilai menu dari menu\_item menggunakan perintah inflate.

Setelah itu, pada setiap menu item yang di klik, akan menjalankan menuitemclicklistener dan pada setiap item akan menjalankan aksi menggunakan statement switch dan dengan mendapatkan nilai id dari setiap item menu.

Switch yang pertama yaitu, apabila case id adalah ubah, maka kejadian yang ditampilkan adalah pesan popup menggunakan Toast dengan isi, Ubah data.

Switch yang kedua yaitu, apabila case id adalah hapus, maka kejadian yang akan ditampilkan adalah pesan popup menggunakan Toast dengan isi, Hapus data.

Kemudian, popupmenu ditampilkan menggunakan perintah show().

1. DELETE adalah perintah untuk melakukan penghapusan data yang ada di tabel. Untuk menghapus data, hanya dengan menuliskan pernyataan seperti ini,

DELETE FROM table\_name WHERE condition;

1. Melakukann delete apabila item tersebut di klik hapus.

Cth :

Di dalam file MainActivity :

public void deleteData(String id){  
 String idBarang = id;  
 String sql = "DELETE FROM tblbarang WHERE idBarang = "+idBarang+" ";  
  
 if (db.runSQL(sql)){  
 pesan("Data sudah dihapus");  
  
 selectData();  
 }else{  
 pesan("Data gagal dihapus");  
 }  
  
}

Artinya, pada function deleteData yang berisi parameter id, terdapat variabel idBarang yang mengambil nilai dari id yang dimasukkan. Lalu terdapat variabel sql yang berisi perintah sql untuk menghapus data dari tblbarang dengan memilih data yang dihapus sesuai dengan idBarang yang dimasukkan.

Lalu terdapat pengujian yaitu apabila runSQL berhasil menjalankan variabel sql, maka akan menampilkan pesan Data sudah dihapus. Kemudian akan melakukan refresh data menggunakan function selectData();

Apabila data gagal dihapus, maka akan menampilkan pesan, Data gagal dihapus.

Di dalam BarangAdapter.java :

case R.id.*hapus*:  
 ((MainActivity)context).deleteData(barangList.get(i).getIdBarang());  
 break;

Artinya, pada case yang mengambil id hapus, akan menjalankan perintah untuk mengambil data dari MainActivity yang dijalankan pada case ini dengan context, lalu fungsi yang diambil adalah deleteData dengan mengisi parameter yang mengambil nilai id dari barangList dengan menggunakan perintah getIdBarang().

**Saya Belum Mengerti**