|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Android Studio vid 29 s/d 35 | 90 |

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

1. **Penggunaan tools:Listitem pada recycler view.** Digunakan agar dapat melihat review atau contoh hasil yang akan tampil ketika data sudah terisi.
2. **Menginsert data dari main activity ke dalam database.** **Pada MainActivity mendeklarasikan nama id** yang digunakan untuk mengisi data. **Pada function load mendaftarkan id** yang tadi sudah di deklarasikan ke dalam function load ini dengan menggunakan findViewById. **Membuat function untuk event onClick pada button** yang digunakan untuk menginsert data, kemudian isi dengan id yang sudah di deklarasikan tadi dan **menambahkannya dengan variable** agar lebih mudah yang digunakan untuk mengambil datanya dengan menggunakan **getText() dan toString()**, kemudian **membuat function if** yang digunakan untuk **mencegah adanya kekurangan data** yang dibutuhkan dengan cara menggunakan variable tadi dengan menggunakan **perintah isEmpty()**, kemudian **isikan pesan yang diinginkan sebagai penanda kurangnya data yang dibutuhkan**, lalu jika data yang dibutuhkan sudah lengkap menggunakan **function else dan masukkan lagi function if untuk menginput/menginsert data** dengan mengisinya dengan variable textview kemudian equals untuk menunjukkan pada keadaan apa function tersebut berjalan, kemudian memasukkan perintah insert untuk sqlnya dengan cara **“INSERT INTO tblbarang(nama tabel yang di dalam database yang ingin diisi datanya) (barang, stok, harga(nama kolom yang ada di dalam tabelnya)) VALUES(yang diisikan kedalamnya) (‘”+barang+”’(karena data berbentuk string), “+stok+”(karena data berbentuk real), “+harga+”(karena data berbentuk real))”**, kemudian jalankan/run sqlnya dengan menggunakan perintah **runSQL**, kemudian tuliskan function pesan yang timbul jika data sudah terpenuhi semua dan perintah insert berhasil dijalankan kemudian **function else untuk mengpdate/memperbarui data**, lalu **set semua id yang digunakan untuk mengisikan data**. **Membuat function pesan** untuk menjalankan perintah diatas tadi dengan menggunakan Toast dengan mengisikannya dengan isi kemudian textnya diubah ke isi. Maka ketika data sudah lengkap dan terpenuhi maka akan menjalankan pesan Insert Berhasil namun ketika insert datanya tidak lengkap akan muncul pesan data belum lengkap, lalu data yang dimasukkan tadi akan tampil ke dalam SQLite Reader.
3. **Agar tau ketika terjadi kesalahan/gagal ketika proses menginput data.** Dengan menambahkan di bawah if untuk memasukkan perintah INSERT INTO ke dalam sql dan menambahkannya dengan if else, dengan if(db.runSQL(sql)) lalu pesan untuk keberhasilan insert data kemudian else untuk kegagalan menginsert data.
4. **Memilih data di dalam database.** Dengan menggunakan perintah **SELECT \* FROM tblbarang(nama tabel yang ingin ditampilkan).** Maka, otomatis data yang ada di dalam data tblbarang akan muncul semua. Penggunaan **ORDER BY** dalam select bisa digunakan untuk mengurutkan data dengan **ASC** untuk mengurutkan data dari kecil ke besar dan **DESC** sebaliknya.
5. **Menampilkan data yang ada di dalam database ke dalam RecyclerView.** **Pada class Database** **membuat function cursor** dengan nama select yang valuenya adalah String sql kemudian diisi dengan try dan catch agar tetap bisa berjalan jika ada masalah. **Pada MainACtivity membuat function untuk selectData** dan deklarasikan juga di dala function onCreate, dan isi function selectData dengan **perintah untuk menselect data “SELECT \* FROM tblbarang(nama tabel yang dipilih) ORDER BY barang ASC(agar hasil tampilannya urut dari kecil ke besar)”**, kemudian isikan dengan **Cursor cursor(nama variable class cursor) = db.select(nama dari class cursor)(sql)**, kemudian **cek ada berapa data** yang ada di dalam tabel databasenya dengan menggunakan **getCount() jadi pesan(cursor.getCount()+””)**, maka ketika dijalankan akan muncul pesan yang bertuliskan jumlah data yang ada di dalam tabel database. Kemudian, **membuat class baru** **untuk modelnya** dengan cara klik kanan pada folder java kemudian new lalu Java Class kemudian beri nama **Barang**. **Pada model Barang,** **membuat deklarasi untuk kolom tabel** yang ada di dalam tabel database yang dinamai dengan **private String kemudian idbarang, barang, stok, harga** lalu buat constructornya dengan mengeklik kanan lalu generate kemudian pilih constructor untuk String idbarang, barang, stok, harga, kemudian juga buat **Getter and Setter untuk String idbarang, barang, stok, harga.** Kemudian **membuat adapternya,** dengan mengklik kanan pada folder java kemudian pilih new lalu pilih Java Class lalu beri nama **BarangAdapter**. **Pada BarangAdapter membuat class untuk view holdernya dengan public class ViewHolder extends RecyclerView.ViewHolder**, kemudian pilih create constructor maka otomatis akan dibuatkan construktornya oleh android studio, **kemudian pada class ViewHoldernya** **mendeklarasikan TextView** dari id TextView yang digunakan untuk menampilkan item - item barang seperti **tvBarang, tvStok,TvHarga,tvMenu**, kemudian pada **function ViewHoldernya membuat variable untuk mencari id** yang digunakan untuk TextViewnya seperti **tvBarang = itemView.findViewById(R.id.tvBarang)** begitu juga dengan textview lainnya yang digunakan. **Pada class BarangAdapter tambahkan extends RecyclerView.Adapter<BarangAdapter.ViewHolder>** kemudian, implement method, lalu, didalam class BarangAdapter tadi **mengambil modelnya dengan mendeklarasikan Context context, lalu List<Barang> barangList**, kemudian generate constructor untuk context dan listnya tadi. Pada **function getItemCount** menambahkan **return barangList.size()**. **Pada function onCreateViewHolder dengan value @NonNull ViewHolder viewHolder, int i** kemudian menambahkannya dengan **View c = LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.item\_barang,viewGroup,false),** kemudian **return new ViewHolder(v)**, dimaksudkan agar semua item yang ada di item barang akan masuk ke dalam view holdernya agar bias ditampilkan. **Pada function onBindViewHolder dengan value @NonNull ViewHolder viewHolder, int i** kemudian **menambahkannya** dengan **viewholder.tvBarang.setText(barangList.get(i).getBarang())**, begitu juga untuk text view lainnya yang digunakan untuk menampilkan item barangnya. **Pada MainActivitynya mendeklarasikan List<Barang> databarang = new ArrayList<Barang>()** agar data yang ditampilkan berbentuk array, kemudian mengambil dari adapternya dengan BarangAdapter adapter, kemudian **mendeklarasikan juga RecyclerViewnya** dengan rcvBarang agar dikenali oleh java lalu tambahkan juga dalam function load untuk mencari lokasi idnya, **kemudian pada function load menambahkan rcvBarang.setLayoutManager(new LinearlayoutManager(this))** dan juga menambahkan **rcvBarang.setHasFixedSize(true)** agar data yang ada di dalam recyclernya akan selalu fixed atau diperbarui. Pada function selectData menambahkan **perintah clear()** untuk databarangnya agar data yang dimasukkan atau dipilih akan selalu clear, lalu menambahkan perintah **if (cursor.getCount() > 0** -> jika data yang dimasukkan lebih dari 0 maka) memasukkan **String idbarang = cursor.getString(cursor.getColumnIndex(“idbarang”))** begitu juga semua yang digunakan untuk mengambil item barang dari cursor, karena data yang diambil banyak maka menggunakan looping **while(cursor.moveToNext())**->agar data di masukkan satu persatu, kemudian setelah mengambil item dari cursor masukkan ke modelnya dengan **databarang.add(new Barang(idbarang,barang,stok,harga)),** lalu atur adapternya agar bisa masuk ke dalam adapterya dengan **adapter = new BarangAdapter(this, databarang)**, kemudian memasukkan RecyclerViewnya ke dalam adapternya dengan **rcvBarang.setAdapter(adapter)**, agar data yang dimasukkan selalu terbarui maka memasukkan perintah **adapter.notifyDataSetChanged()**. **Pada MainActivity untuk insert data** ditambahkan function **selectData()** di dalamnya agar data yang diisi akan otomatis terselect dan akan tampil. Maka data yang di dalam databasenya akan tampil di dalam RecyclerViewnya, dan ketika memasukkan data barupun akan terbarui dan akan secara otomatis akan tampil di dalam RecyclerViewnya.
6. **Menampilkan Pop up menu.** **Menambahkan folder resource baru** dengan cara mengklik kanan pada folder res kemudian new pilih Android Resource Directory, kemudian untuk **namanya menu** dan resource typenya pilih menu, **kemudian tambahkan file baru untuk Android Resource Directory** tadi dengan mengklik kanan folder menu tadi yang ditambahkan kemudian pilih new lalu klik Menu Resource File kemudian memberi **nama menu\_item**. Kemudian, **pada menu\_item tambahkan** **komponen menu item untuk ubah dan hapus**. Kemudian agar tampil, pada **BarangAdapter** di dalam **function onBindViewHolder** menambahkan dengan **viewHolder.tvMenu.setOnClickListener(new View.OnClickListener()**->agar text view untuk tampilan pop up menunya dapat di klik. Kemudian, didalam viewHolder untuk text view menu tadi tambahkan function onClicknya dengan menggunakan value **View v** lalu, isikan dengan perintah **PopupMenu popupMenu(variable untuk pop up menu) = new PopupMenu(context**(karena adapternya akan dijalankan di MainActivity), **viewHolder.tvMenu)** kemudian memasukkan menu\_item ke dalam pop up menu dengan **popupMenu.inflate(R.menu.menu\_item**(lokasi menu\_item yang akan dimasukkan ke dalam pop up menu)). Kemudian untuk menampulkan pop up menunya dengan menggunakan perintah show() dengan **popupMenu.show()**.
7. **Menjalankan tompol pop up menu sesuai dengan fungsinya.** Pertama, mengecek apakah perintah yang kita tambahkan berhasil jalan atau tidak. **Pada BarangAdapter** menambahkan **popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener()** kemudian didalamnya tambahkan function **onMenuItemClick(MenuItem item)** kemudian menambahkan switch case untuk tombol menu ubah dan hapus dengan **switch(item.getItemId())** kemudian untuk casenya pertama untuk tombol ubah jadi, **case R.id.ubah** lalu, gunakan Toast untuk menampilkan pesan ketika tombol ubah di klik maka akan muncul pesan ubah. Begitu juga untuk case tombol hapus, kemudian **case R.id.hapus** lalu, Toast untuk menampilkan text hapus. Maka, ketika berhasil tombol ubah pada pop up menu di klik akan muncul pesan ubah dan begitu juga dengan tombol pop up menu hapus.
8. **Menjalankan tombol menu hapus untuk mengahapus data. Pada MainActivity,** **membuat** **function baru** dengan nama **deleteData** dengan value String id, lalu di dalamnya tambahkan **String idbarang = id**, lalu memasukkan perintah sql yang digunakan untuk mengapus data yaitu **String sql = “DELETE FROM tblbarang WHERE idbarang = “+idbarang+”;”**, lalu menambahkan **if(db.runSQL(sql))** maka menampilkan **pesan()(“Data Sudah Dihapus”)**->menggunakan function untuk pesan, kemudian menggunakan **function selectData()** untuk memilih data yang dihapus dan akan terefresh kemudian otomatis akan terhapus, lalu menambahkan else jika data gagal untuk dihapus dengan mengisinya dengan **function pesan(“Data Gagal Dihapus”)**. Kemudian, Pada Switch Case tombol hapus yang berada di dalam BarangAdapter di ubah menjadi **((MainActivity)context).deleteData(barangList.get(i)getIdBarang())**.Maka, jika tombol menu di klik lalu memilih tombol hapus maka data otomatis akan terhapus dan muncul pesan yang sudah di tuliskan tadi.

**Saya Belum Mengerti**

1.

2.

3.