|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Android Studio 36 – 41 Komputer Kit | 95 |

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

1. SQLite Database – 11 SQL Select Where

* Buat function selectUpdate dengan parameter string id
* Buat variabel string berisi perintah sqlwhere (select \* from namaTabel where kondisi)
* Buat variabel cursor isi dengan db.select(namaVarSQL)
* ***((***MainActivity***)*context*)***.selectUpdate***(*barangList**.get***(***position***)***.getIdbarang***())***; panggil function select update dari mainactivity di adapter dan isi parameter id dengan idbarang
* cursor.moveToNext***()***; untuk menampilkannya
* **Inputan**.setText***(*** cursor.getString***(***cursor.getColumnIndex***(*"Kolom"*)))***; untuk mengisi inputan dengan data yang diambil dari kolom

2. SQLite Database – 12 SQL Update

* Di function simpan ambil semua data dari inputan ke dalam beberapa variabel
* Buat pengujian jika salah satu inputan kosong maka tampilkan pesan inputan kosong
* Buat variabel string berisi perintah sql update (update namaTabel set NamaKolom = data baru where kondisi)
* Jalankan menguunakan db.runsql

3. SQLite Database - 13 Menggunakan dialog

* AlertDialog.Builder builder = **new** AlertDialog.Builder***(*this*)***; **dekalrasikan dulu**  
  builder.setTitle***(*"PERINGATAN"*)***; **memberi judul**  
  builder.setMessage***(*"Yakin akan menghapus"*)***; **memberi pesan**  
  builder.setPositiveButton***(*"YA"**, **new** DialogInterface.OnClickListener***() {*** @Override  
   **public void** onClick***(***DialogInterface dialog, **int** which***) {*** String SQL = **"DELETE FROM tblbarang WHERE idbarang ="**+id+**";"**;  
   **if *(*db**.runSQL***(***SQL***)){*** pesan***(*"Data sudah dihapus"*)***;  
   selectData***()***;  
   ***}*else*{*** pesan***(*"Data tidak bisa dihapus"*)***;  
   ***}  
   }  
  })***; **jika menekan tombol ya**  
  builder.setNegativeButton***(*"TIDAK"**, **new** DialogInterface.OnClickListener***() {*** @Override  
   **public void** onClick***(***DialogInterface dialog, **int** which***) {  
    
   }  
  })***; **jika menekan tombol tidak**  
  builder.show***()***; **Menampilkan dialog**

4. SharedPreferences

* Buat layout berisi dua inputan dan 2 button
* Findviewbyme di main activity
* Deklarasikan sharedpreferences
* Di oncreate isi sharedpreferences dengan getSharedPreferences***(*"namaFile"**,***MODE\_PRIVATE)***; mode private itu file hanya bisa diakses oleh aplikasi yang kita buat
* Di function simpan ambil data dari inputan, buat pengujian jika salah satu inputan kosong maka tampilkan pesan dan jika tidak maka deklarasikan SharedPreferences.Editor (SharedPreferences.Editor editor = **sharedPreferences**.edit***()***;) lalu editor.putString***(*"Nama"**,dataInput***)***; untuk menginputkan ke sharedpreferences kemudian clear kan semua inputan

5. Sharedpreferences-2

* Di function tampil tampilkan di inputan **etBarang**.setText***(***sharedPreferences.getString("Nama",nilai jika tidak ditemukan)***)***; menampilkan nilai yang sudah disimpan di Sharedpreferences
* Data tidak hilang meskipun keluar aplikasi

6. Debugging ( Pencarian Error )

* Jika ada salah penulisan maka di nama file ada garis merah dan pada tulisan yang salah bewarna merah
* Jika ada error dari perintah sql berupa string maka aplikasi akan forclose cara mencarinya di run ada tulisan bewarna merah

**Saya Belum Mengerti**