|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Android Studio video 36-41   * Memahami fungsi SQL Select Where * Menggunakan SQL Select Where * Memahami fungsi SQL Update * Menggunakan SQL Update * Memahami fungsi dialog * Menggunakan dialog sebelum delete * Memahami fungsi SharedPreference * Menggunakan SharedPreference * Memahami Error * Menemukan Error | 90 |

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

**1.** Memahami bahwa syntax sql untuk select where adalah SELECT \* FROM table\_name WHERE condition;

**2.** Memahami bahwa sql select where berfungsi untuk select data

**3.** Memahami cara meng-select data mana yang akan diubah (menggunakan sql select where) sebelum data tersebut diupdate, yaitu :

* Membuat function untuk select data yang akan di update data lalu pada parameternya diisi string id karena pengambilan data berfokus pada id nya
* Deklarasikan variable String untuk idbarang;
* Pada function untuk selectUpdate buat variable untuk menampung idbarang yaitu idbarang = id;
* Lalu tambahkan variable string untuk menampung script sql daripada select where "DELETE FROM tblbarang WHERE idbarang = "+idbarang+";";
* Gunakan cursor untuk menjalankan sql select dalam database Cursor cursor = db.select(sql);
* Lalu tambahkan perintah cursor.moveToNext(); untuk menggerakan cursor nya
* Ambil data dan letakkan data pada form dengan format script idform.setText(cursor.getString(cursor.getColumnIndex("variable data nya misal barang")));
* Untuk mengatur agaar tulisan di keterangan atas form menunjukkan indikator update ketika data tersebut di update maka di set terlebih dahulu tvPilhan.setText(“update”);

**4.** Memahami dan dapat menerapkan penggunaan sql update untuk proses update data setelah data di select :

* Pada function else bagian else untuk update data isi dengan script sql untuk update data String sql = "UPDATE tblbarang" + "SET barang = '"+barang+"', stok="+stok+", harga="+harga+" WHERE idbarang= "+idbarang+";"; yang dibungkus dalam variable dengan format data String
* Lalu buat if untuk menjalankan sql nya yaitu (db.runSQL(variable)){ isi dengan pesan apabila data berhasil di update dan tidak lupa function selectData() untuk refresh jadi tampilan akan langsung menunjukkan perubahan daripada data dan pada indikator data update-insert
* Sedangkan else nya akan diisi message apabila update data gagal dilakukan

**5.** Memahami proses jalannya program yaitu :

* Ketika aplikasi berjalan maka program akan menjalankan function load terlebih dahulu untuk mengakses database/mengambil data
* Lalu menjalankan function simpan untuk mengecek apakah ada data yang dapat ditampilkan/diproses diantaranya barang, stok, harga dan indikator
* Kemudian data tersebut di cek didalam if apakah kosong atau terdapat value
* Apabila kosong maka akan menampilkan pesan bahwa data kosong
* Apabila terdapat data maka akan menampilkan data dimana bisa terdapat opsi insert, hapus maupun ubah yang dapat dipilih untuk digunakan nantinya

**6.** Memahami & mengetahui bahwa syntax sql untuk update data adalah UPDATE table\_name SET column1 = value1, column2 = value2, .. WHERE condition;

**7.** Memahami bahwa dialog dapat berfungsi sebagai alert sebelum melakukan sesuatu perubahan/penghapusan/proses pada sebuah data

**8.** Memahami dan dapat menerapkan pengguanaan alert dialog untuk warning sebelum penghapusan data (seperti pada materi) yaitu :

* Pada function deleteData Karena akan menggunakan fokus id dari idbarang yang di deklarasikan maka format data String nya dihapus sehingga hanya menyisakan idbarang = id;
* Lalu menmbuat AlertDialog.Builder nama\_alrt = new AlertDialog.Builder(context:this);
* Set judul untuk peringatan nama\_alert.setTitle("PERINGATAN!");
* Set keterangan daripada alert nya nama\_alert.setMessage("Yakin ingin menghapus?");
* Untuk memberikan opsi YA dan TIDAK maka tambahkan setPositiveButton dan setNegativeButton
* Untuk opsi YA maka letakkan dala setPositiveButton dan pindahkan if else script penghapusan data kedalam method nya
* Sedangkan untuk opsi TIDAK maka gunakan syntax dialog.cancel(); dalam method di setNegativeButton sehingga pop up alert dialog akan menutup apabila opsi TIDAK di klik yang artinya proses penghapusan dibatalkan

**9.** Memahami bahwa sharedPreference adalah file yang digunakan untuk menyimpan berbagai setting-setting dari sebuah aplikasi

**10.** Memahami bahwa sharedPreference dapat berbentuk dengan berbagai tipe data

**11.** Memahami bahwa pada sharedPreference dapat mengatur MODE\_PRIVATE agar yang bisa menggunakan file hasil sharedPreference yg ditentukan hanyalah aplikasi satu tersebut/aplikasi yang diizinkan saja

**12**. Mengetahui & memahami bahwa di dalam getString() terdapat 2 macam yaitu String key untuk nama yang sudah disimpan sebelumnya dan defValue yaitu default value

**13.** Memahami bahwa meskipun aplikasi sudah di close lalu di open ulang maka data masih akan tetap tersimpan di dalam sharedPreference, oleh karena itu hal tersebut banyak dijadikan pilihan untuk menyimpan setting sebuah aplikasi

**14.** Dapat menerapkan penggunaan sharedPreference pada program, misa untuk menyimpan data inputan lalu di tampilkan lagi :

* Membuat lembar kerja baru
* Pada xml nya tambahkan atribut yaitu 2 text view untuk input barang dan jumlah stok nya
* Lalu 2 button untuk simpan data dan menampilkan nya
* Beri id pada text view yaitu etBarang dan etStok lalu id pada button yaitu simpan dan tampil
* Tambahkan onClick pada kedua button lalu cretae method nya
* Deklarasikan variable TextView untuk kedua form tadi karena itu merupakan media yang akan terhubung dengan input sehingga perlu diisi.
* Lalu deklarasi juga untuk SharedPreferences sharedPreferences;
* Membuat function load untuk menghubungkan dengan form TextView menggunakan finViewById
* Lalu untuk menyambungkan dengan sharedPreferences gunakan getSharedPreferences(“barang”, MODE\_PRIVATE)
* Membuat function untuk mengisi preferences dengan params (String barang, Float stok) karena data barang yg akan di gunakan berbentuk String sementara data stok berbentuk Float
* Gunakan editor agar isi di dalam sharedPrefereces dapat diubah-ubah SharedPreferences.Editor editor = sharedPreferences.edit();

Format untuk mengubah datanya editor.putString(“nama yang disimpan”, variablenya); terakhir tambahkan editor.apply(); untuk mengaktifkan editor

* Pada method simpan hasil onCLick tadi ambil data barang terlebih dahulu menggunaan getText().toString(); lalu untuk data Float perlu di transform dulu menjadi bentuk string dengan menggunakan Float.parseFloat(..)
* Lalu tambahkan if (variable barang.isEmpty() || stok == 0.0) maka akan menampilkan pesan data kosong
* Sementara else nya akan mengaktifkan function isiSharedPreferences(variable1, variable2 dari TextView); yang artinya datanya akan berhasil diisikan kedalam sharedPreferences dan menampilkan pesan data sudah di simpan
* Kemudian set lagi Form TextView nya menjadi kosong dengan id.setText(“”);
* Sedangkan pada method tampil ambil data barang dan stok dari sharedPreferences dengan script TipeData variable = sharedPreferences.getString("variable/nama", ""); untuk stok maka defValue nya diisi 0 karena default jumlah adalah 0
* Kemudia setText nya lagi agar data yang dipanggil akan ditampilkan pada form dengan syntax id.setText(variable); dan tambahkan +”” untuk tipe data selain string

**15.** Debbuging/pencarian error yaitu proses mencari error atau kesalahan pada code, proses tersebut disebut debbuging. Berasal dari kata bug yang artinya kutu karena error bentuknya kecil sehingga butuh ketelitian untuk mencari nya.

**16.** Memahami Guide debbuging error/bug pada program :

* Salah dalam penulisan/typo misal String jadi Strinf atau misal Strin [bug semacam ini biasanya ditunjukkan dalam indikator merah]
* Ada pula bug yang tidak ditampilkan pada indikator namun akan menyebabkan code tidak dapat di compile dan akan menunjukkan keterangan kegagalan compile yang artinya ada code yang salah, bisa juga aplikasi menjadi force close karena adanya kesalahan tadi [lalu untuk mencari kesalahan/bug nya ada di line mana bisa dengan cara mengakses run pada pojok aplikasi lalu ke debug untuk membaca petunjuk kalimat berwarna merah dan akan menunjukkan line kesalahan lalu di klik maka akan mengarahkan ke baris nya]
* Untuk dapat menyelesaikan error sangat penting untuk membaca keterangan kesalahan yang ada agar dapat mengetahui bagian mana dari code yang perlu di benarkan.
* Bisa juga dengan mengakses logCat setelah di run karena biasanya akan menunjukkan keterangan error lebih spesifik bagian mana dan apa yang terjadi [kalimat berwarna merah lah menjadi fokus bagian dari error code].
* Memahami juga bahwa apabila bug tersebut berbentuk string maka java tidak akan menujukkan indikasi errornya dan aplikasi akan berjalan terus hingga ke compile, namun dalam penggunaan nya program tidak akan berjalan seperti seharusnya.
* Memahami bahwa untuk proses debbuging dipastikan program pada kondisi di run/compile/berjalan
* Memhaami pula bahwa daalm bug terdapat 2 jenis kesalahan yaitu kesalahan program/java dan kesalag string /dari tulisan yang kita buat

**Saya Belum Mengerti**

1.

2.

3.