Search By ID.

```
Public ArrayList<Records> Search_Student_by_Id (String Id) {
        db = this.getReadableDatabase();
String query = "Select * from" + TABLE-NAME
  Cursor cursor = db.rawQuery(query, null);
   String student-id = "Empty";
Arraylist<Records> arraylist = new ArrayList<>();
 if (cursor.moveToNext()){
        do {
            student-id = cursor.getString(4);
           if (stdent-id.Equals.(id))
            {
              Records records = new Records();
              records.setusername(cursor.getString(1));
               records.SetEmail(cursor.getString(2));
                 "      "      "      "      "
               arraylist.add(records);
              break
            }
    }while(cursor.moveToNext());
  }
   return arraylist;
}
```

SQLite Connection helper = new SQLite Connection (this);

LoGin. (All code Same as Above).

```
Public Arraylist<Records> User-login (String u-Pwd){
   String Password = "Empty";
 if (u.Pwd.Equals(Password){
        records.Set username = (cursor.getStrng(1));

 return arraylist;
```

Search List USERS :-

```
db = this.getReadableDatabase();
String query = "Select user-name" from" + "TABLE-NAME";
 Cursor cursor = db.rawQuery(query, null);
Arraylist<string> arraylist = new Arraylist();
 String u-name = "Empty";
do {
      u-name = cursor.getstring(0);
      arraylist.add(u-name);
   }while.
return arrayList:
```

Set IN ADAPTER ARRAYLIST

```
listview = (listview) findby ID
(Make array list of String)
arraylist = helper.Search.....
ArrayAdapter arrayadapter.
listview.SetAdapter(array
                    Adapter);
```

SQLite.

(Class Records){ String username, Email...... }. Get & Setters.

Class SQLiteConnection, Extends SQLite OpenHelper

SQLiteConnection Extends SQLite OpenHelper

Implement. ←

↓

Constructor, onCreate, onUpgrade Methods.

Create table & DATABASE.

SQLiteDatabase db;

Public static String DATABASE_NAME = "DATABASE_MANAGEMENT, Db";

Public static String TABLE_NAME = "data";

" " " " COLOUMN1 = "id";

" " " " COLOUMN2 = "user-name";

And Continue.

" " " "

CONSTRUCTOR.

Super (context, DATABASE_NAME, factory null, version 1);

onCreate.

String query = "Create table data" + "(id integer Primary key auto increment not null, user-name text, --- text)";

db. ExecSQL(query);

OnUpgrade.

db. ExecSQL ("DROP TABLE IF EXISTS" + DATABASE-NAME + ";");

db. close ();

INSERT.

db = this. getWritable Database();

ContentValues Content values = new Content Values ();

Content values. Put (coloumn2, U-name);

Content Values. Put (column3, U_Email);

" " " " "

Public boolean Insert_data(u-name, u-Email......)

long result = db. insert (TABLE-NAME
, null
, Content values);

if (result = -1);
return false
Else
return true;

UPDATE.

Public boolean Update(String U-name, String U-Email ....){

db = this. getWritable Database ();

ContentValues Content values ~~Content Values~~ = new Content Values ();

Content values. Put (COLOUMN2, U-name);

Content values. Put (Coloumn 3, U- Email);

db. update (TABLE_NAME, whereclause: "User-ID=?, new String[]{U-id});

return true.

}