***CS407 – Mobile Application Development***

**Assignment #04**

Submitted By

**Haseeb Pervaiz Mirza (18321519-047)**

**BSCS (Section A)**

Submitted To

**Dr. Muhammad Usman Ali**

***University of Gujrat***

***Hafiz Hayat Campus***

***Department of Computer Science***

## ApplicationAdapter

package com.example.ems;

import android.content.Context;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseAdapter;

import android.widget.Button;

import android.widget.TextView;

import java.util.List;

public class AplicationAdapter extends BaseAdapter {

List<Application> applicationList;

public AplicationAdapter(List<Application> applicationList) {

Log.i("MY HALL" , "ADAPTER ADDED");

this.applicationList = applicationList;

}

@Override

public int getCount() {

return applicationList.size();

}

@Override

public Object getItem(int position) {

return null;

}

@Override

public long getItemId(int position) {

return 0;

}

@Override

public View getView(int position, View convertView, ViewGroup parent) {

String id = applicationList.get(position).getId();

Bundle bundle = new Bundle();

bundle.putString("id" , id);

LayoutInflater layout = LayoutInflater.from(parent.getContext());

View view = layout.inflate(R.layout.hall\_list , parent , false);

((TextView) view.findViewById(R.id.hall\_list\_view)).setText(applicationList.get(position).getName()+" : "+ applicationList.get(position).getCNIC());

Button Read , Update , Delete;

// Binding View Button

Read= view.findViewById(R.id.hallViewBtn);

Read.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Context context = v.getContext();

Intent intent = new Intent(context, One\_Application\_Activity.class);

intent.putExtras(bundle);

context.startActivity(intent);

}

});

Update = view.findViewById(R.id.hallUpdateBtn);

Update.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Context context = v.getContext();

Intent intent = new Intent(context, Application\_view\_Activity.class);

intent.putExtras(bundle);

context.startActivity(intent);

}

});

if(position >=0) {

Delete = view.findViewById(R.id.hallDeleteBtn);

Delete.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Log.i("EVENT", "DELETING");

Context context = v.getContext();

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(context);

if (applicationDBHelper.deleteHall(Integer.parseInt(id)) != 0) {

removeViewAt(position);

} else {

Log.i("EVENT", "DELETION FAILED");

}

}

});

}

return view;

}

public void removeViewAt(int position){

applicationList.remove(position);

notifyDataSetChanged();

}

}

## Application.java

package com.example.ems;

public class Application {

private String Name;

private String CNIC;

private String Status;

private String CGPA;

private String Degree;

private String id;

public Application(String Name, String CNIC, String status, String CGPA, String degree, String id) {

this.Name = Name;

this.CNIC = CNIC;

Status = status;

this.CGPA = CGPA;

Degree = degree;

this.id = id;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return Name;

}

public void setName(String name) {

this.Name = name;

}

public String getCNIC() {

return CNIC;

}

public void setCNIC(String CNIC) {

this.CNIC = CNIC;

}

public String getStatus() {

return Status;

}

public void setStatus(String status) {

Status = status;

}

public String getCGPA() {

return CGPA;

}

public void setCGPA(String CGPA) {

this.CGPA = CGPA;

}

public String getDegree() {

return Degree;

}

public void setDegree(String degree) {

Degree = degree;

}

}

## Application\_View.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import android.widget.Toast;

public class Application\_view\_Activity extends AppCompatActivity {

TextView Name, CNIC, CGPA, status, degreee;

Button finishbtn;

String id;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.hall\_view\_activity);

Name = findViewById(R.id.HallName);

CNIC = findViewById(R.id.HallCapacity);

CGPA = findViewById(R.id.floors);

status = findViewById(R.id.Rent);

degreee = findViewById(R.id.location);

finishbtn = findViewById(R.id.hallAddAndUpdateBtn);

if(getIntent().getExtras() != null){

Log.i("EVENT" , "UPDATION STEP");

Bundle bundel = getIntent().getExtras();

id = bundel.getString("id");

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(Application\_view\_Activity.this);

Application application = applicationDBHelper.getData(Integer.parseInt(id));

CNIC.setText(application.getStatus());

Name.setText(application.getName());

CGPA.setText(application.getCGPA());

status.setText(application.getDegree());

degreee.setText(application.getCNIC());

}

finishbtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String Name = Application\_view\_Activity.this.Name.getText().toString();

String Capacity = CNIC.getText().toString();

String Floor = CGPA.getText().toString();

String Rent = status.getText().toString();

String Location = degreee.getText().toString();

if(Name.isEmpty()){

ToastShow("Hall Name");

}

else if(Capacity.isEmpty()){

ToastShow("Capacity");

}

else if(Floor.isEmpty()){

ToastShow(" Floors ");

}

else if(Rent.isEmpty()){

ToastShow("Rent");

}

else if(Location.isEmpty()){

ToastShow("Location");

}

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(Application\_view\_Activity.this);

if(getIntent().getExtras() != null ){

if(applicationDBHelper.updateApplication(Integer.parseInt(id) , Name , Capacity , Floor , Rent , Location ))

{

Log.i("APPLICATION" , "UPDATED");

}

else{

Log.i("APPLICATION" , "UPDATION FAILED");

}

}

else {

if (applicationDBHelper.insertApplication(Name , Capacity , Floor , Rent , Location)) {

Log.i("APPLICATION", "Hall Added");

}

else{

Log.i("APPLICATION", "Hall Addition Failed");

}

}

}

});

}

public void ToastShow(String field){

Log.i("MYTAG" , "` `");

Toast.makeText(this , field+" is required",Toast.LENGTH\_LONG).show();

Log.i("MYTAG" , "` `");

}

}

## Application DB\_HELPER

package com.example.ems;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteOpenHelper;

import android.database.sqlite.SQLiteDatabase;

import android.util.Log;

public class ApplicationDBHelper extends SQLiteOpenHelper {

public static final String DATABASE\_NAME = "MyDBName.db";

public static final String APPLICATION\_TABLE\_NAME = "Application";

public static final String APPLICATION\_COLUMN\_ID = "id";

public static final String APPLICATION\_COLUMN\_NAME = "Name";

public static final String APPLICATION\_COLUMNS\_CGPA = "cgpa";

public static final String APPLICATION\_COLUMN\_STATUS = "status";

public static final String APPLICATION\_COLUMN\_DEGREE = "degree";

public static final String APPLICATION\_COLUMN\_CNIC = "cnic";

private HashMap hp;

public ApplicationDBHelper(Context context) {

super(context, DATABASE\_NAME , null, 1);

}

@Override

public void onCreate(SQLiteDatabase db) {

// TODO Auto-generated method stub

db.execSQL(

"create table HALL " +

"(id integer primary key AUTOINCREMENT , Name text,cgpa text,capacity text,floors text, location text)"

);

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

// TODO Auto-generated method stub

db.execSQL("DROP TABLE IF EXISTS HALL");

onCreate(db);

}

public boolean insertHall (String hallName, String capacity,String floors, String rent, String location) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put(APPLICATION\_COLUMN\_STATUS , capacity);

contentValues.put(APPLICATION\_COLUMN\_DEGREE , floors);

contentValues.put(APPLICATION\_COLUMNS\_CGPA , rent);

contentValues.put(APPLICATION\_COLUMN\_NAME, hallName);

contentValues.put(APPLICATION\_COLUMN\_CNIC, location);

db.insert(APPLICATION\_TABLE\_NAME, null, contentValues);

return true;

}

public Application getData(int Id) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select \* from HALL where id="+Id, null );

res.moveToFirst();

String id = res.getString((res.getColumnIndex("id")));

String hallName = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_NAME));

String capacity = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_STATUS));

String floors = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_DEGREE));

String rent = res.getString(res.getColumnIndex(APPLICATION\_COLUMNS\_CGPA));

String location = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_CNIC));

// craeting Event oject to store the info. reterieved from database

Application application = new Application(hallName , location,capacity , floors , rent , id);

return application;

}

/\*

public int numberOfRows(){

SQLiteDatabase db = this.getReadableDatabase();

int numRows = (int) DatabaseUtils.queryNumEntries(db, CONTACTS\_TABLE\_NAME);

return numRows;

}

\*/

public Integer deleteHall (Integer id) {

SQLiteDatabase db = this.getWritableDatabase();

return db.delete("HALL",

"id = ? ",

new String[] { Integer.toString(id) });

}

public boolean updateHall (Integer id,String hallName, String capacity,String floors, String rent, String location) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put(APPLICATION\_COLUMN\_CNIC, location);

contentValues.put(APPLICATION\_COLUMN\_NAME, hallName);

contentValues.put(APPLICATION\_COLUMNS\_CGPA , rent);

contentValues.put(APPLICATION\_COLUMN\_DEGREE , floors);

contentValues.put(APPLICATION\_COLUMN\_STATUS , capacity);

db.update("HALL", contentValues, "id = ? ", new String[] { Integer.toString(id) } );

return true;

}

public List<Application> getAllHalls() {

//ArrayList<String> array\_list = new ArrayList<String>();

List<Application> applicationList = new ArrayList<>();

//hp = new HashMap();

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select \* from HALL", null );

res.moveToFirst();

while(res.isAfterLast() == false){

String id = res.getString((res.getColumnIndex("id")));

String hallName = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_NAME));

String capacity = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_STATUS));

String floors = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_DEGREE));

String rent = res.getString(res.getColumnIndex(APPLICATION\_COLUMNS\_CGPA));

String location = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_CNIC));

Log.i("HALL" , hallName );

Application application = new Application(hallName , location , capacity, floors , rent , id );

applicationList.add(application);

//eventList.add(event);

res.moveToNext();

}

return applicationList;

}

public ArrayList<String> getNames(){

ArrayList<String> hallNames = new ArrayList<String>();

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select hallName from HALL", null );

res.moveToFirst();

while(res.isAfterLast() == false){

String hallName = res.getString(res.getColumnIndex(APPLICATION\_COLUMN\_NAME));

hallNames.add(hallName);

res.moveToNext();

}

return hallNames;

}

}

## Allpication list

package com.example.ems;

import android.content.Intent;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.ListView;

import java.util.ArrayList;

import java.util.List;

public class Applicationl\_List\_Activity extends AppCompatActivity {

List<Application> applicationList = new ArrayList<>();

ListView ApplicationSpinner;

Button addbtn;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.hall\_list\_activity);

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(Applicationl\_List\_Activity.this);

applicationList = applicationDBHelper.getAllHalls();

ApplicationSpinner = findViewById(R.id.Halls);

// Log.i("MYTAG" , eventList.get(0).getTitle());

AplicationAdapter aplicationAdapter = new AplicationAdapter(applicationList);

ApplicationSpinner.setAdapter(aplicationAdapter);

addbtn = findViewById(R.id.addHallbtn);

addbtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent newActivity = new Intent( Applicationl\_List\_Activity.this , Application\_view\_Activity.class);

startActivity(newActivity);

}

});

}

@Override

protected void onResume() {

super.onResume();

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(Applicationl\_List\_Activity.this);

Log.i("MY HALL" , "Resumed");

applicationList = applicationDBHelper.getAllHalls();

ApplicationSpinner = findViewById(R.id.Halls);

// Log.i("MYTAG" , eventList.get(0).getTitle());

AplicationAdapter aplicationAdapter = new AplicationAdapter(applicationList);

Log.i("MY HALL" , String.valueOf(aplicationAdapter.getCount()));

ApplicationSpinner.setAdapter(aplicationAdapter);

}

}

## Main Activity.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

}

## Application.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.TextView;

public class One\_Application\_Activity extends AppCompatActivity {

TextView Name, CNIC, status, cgpa, degree;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.one\_hall\_layout);

Bundle bundel = getIntent().getExtras();

String id = bundel.getString("id");

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(One\_Application\_Activity.this);

Application application = applicationDBHelper.getData(Integer.parseInt(id));

Name = findViewById(R.id.Hall\_Name);

Name.setText(application.getName());

CNIC = findViewById(R.id.hall\_capacity);

CNIC.setText(application.getStatus());

status = findViewById(R.id.hall\_rent);

status.setText(application.getDegree());

cgpa = findViewById(R.id.hall\_floors);

cgpa.setText(application.getCGPA());

degree = findViewById(R.id.hall\_location);

degree.setText(application.getCNIC());

}

}

## Scholarship.java

package com.example.ems;

public class Scholarship {

private String Title;

private String organizer;

private String Description;

private String cgpa;

private String Province;

private String degree;

private String Date;

private String id;

public Scholarship(String title, String organizer, String description, String cgpa, String province, String degree, String Date, String id) {

Title = title;

this.organizer = organizer;

Description = description;

this.cgpa = cgpa;

Province = province;

this.degree = degree;

this.Date = Date;

this.id = id;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getDate() {

return Date;

}

public void setDate(String date) {

this.Date = date;

}

public String getDegree() {

return degree;

}

public void setDegree(String degree) {

this.degree = degree;

}

public String getProvince() {

return Province;

}

public void setProvince(String province) {

Province = province;

}

public String getCgpa() {

return cgpa;

}

public void setCgpa(String cgpa) {

this.cgpa = cgpa;

}

public String getDescription() {

return Description;

}

public void setDescription(String description) {

Description = description;

}

public String getOrganizer() {

return organizer;

}

public void setOrganizer(String organizer) {

this.organizer = organizer;

}

public String getTitle() {

return Title;

}

public void setTitle(String title) {

Title = title;

}

}

## Scholarship\_One.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.ListView;

import java.util.ArrayList;

import java.util.List;

public class Scholarship\_List\_Activity extends AppCompatActivity {

List<Scholarship> scholarshipList = new ArrayList<>();

ListView ScholarshipSpinner;

Button addbtn , refbtn;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.event\_list\_acaticty);

// get\_data();

ScholarshipDBHelper scholarshipDbHelper = new ScholarshipDBHelper(Scholarship\_List\_Activity.this);

scholarshipList = scholarshipDbHelper.getAllEvents();

ScholarshipSpinner = findViewById(R.id.scholerships);

// Log.i("MYTAG" , eventList.get(0).getTitle());

ScholarshipAdapter listadapter = new ScholarshipAdapter(scholarshipList);

ScholarshipSpinner.setAdapter(listadapter);

addbtn = findViewById(R.id.addnewbtn);

addbtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent newActivity = new Intent( Scholarship\_List\_Activity.this, Scholarship\_view\_Activity.class);

startActivity(newActivity);

}

});

}

@Override

protected void onResume() {

super.onResume();

ScholarshipDBHelper scholarshipDbHelper = new ScholarshipDBHelper(Scholarship\_List\_Activity.this);

Log.i("EVENTS" , "Resumed");

scholarshipList = scholarshipDbHelper.getAllEvents();

ScholarshipSpinner = findViewById(R.id.scholerships);

// Log.i("MYTAG" , eventList.get(0).getTitle());

ScholarshipAdapter listadapter = new ScholarshipAdapter(scholarshipList);

ScholarshipSpinner.setAdapter(listadapter);

}

/\*

public void get\_data()

{

String json;

InputStream is = null;

try {

is = getAssets().open("Event.json");

File extdir1 = Environment.getExternalStorageDirectory();

File myDatadir1 = new File(extdir1 , "/testing");

if(!myDatadir1.exists()){

Toast.makeText(this , "File dose not exist",Toast.LENGTH\_LONG).show();

}

File myfile1 = new File(myDatadir1 , "Event.json");

FileInputStream fis = new FileInputStream(myfile1);

int size = fis.available();

byte[] buffer = new byte[size];

fis.read(buffer);

fis.close();

json = new String(buffer , "UTF-8");

JSONArray jsonarray = new JSONArray(json);

for (int i=0 ; i<jsonarray.length();i++)

{

JSONObject obj = jsonarray.getJSONObject(i);

Event s = new Event(obj.getString("Title"),obj.getString("From"),obj.getString("Description") ,obj.getString("Province"), obj.getString("Date") , obj.getString("Degree"));

eventList.add(s);

}

} catch (IOException e) {

e.printStackTrace();

}

catch (JSONException e) {

e.printStackTrace();

}

}

\*/

}

## Scholarship\_List.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.TextView;

public class Scholarship\_One\_Record extends AppCompatActivity {

TextView title , org , date , desc , cgpa, province, deg;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.event\_one\_\_record);

Bundle bundel = getIntent().getExtras();

String id = bundel.getString("id");

ScholarshipDBHelper scholarshipDbHelper = new ScholarshipDBHelper(Scholarship\_One\_Record.this);

Scholarship scholarship = scholarshipDbHelper.getData(Integer.parseInt(id));

title = findViewById(R.id.view\_title);

title.setText(scholarship.getTitle());

org = findViewById(R.id.view\_org);

org.setText(scholarship.getOrganizer());

date = findViewById(R.id.view\_date);

date.setText(scholarship.getProvince());

desc = findViewById(R.id.view\_desc);

desc.setText(scholarship.getDescription());

cgpa = findViewById(R.id.view\_limit);

cgpa.setText(scholarship.getCgpa());

province = findViewById(R.id.view\_budget);

province.setText(scholarship.getDegree());

deg = findViewById(R.id.view\_event);

deg.setText(scholarship.getDate());

}

}

## Scholarship\_View.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.app.AlertDialog;

import java.io.File;

import android.app.DatePickerDialog;

import android.os.Bundle;

import android.os.Environment;

import android.util.Log;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.DatePicker;

import android.widget.Spinner;

import android.widget.TextView;

import android.widget.Toast;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStream;

import java.util.Calendar;

public class Scholarship\_view\_Activity extends AppCompatActivity {

private DatePickerDialog datePickerDialog;

private Button dateButton;

Spinner degspin;

Button addAndUpdateBtn;

TextView title,from,description,province,gpa;

String SelectedDegree, Date;

String Deg;

String id;

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.event\_view\_layout);

initDatePicker();

addAndUpdateBtn = findViewById(R.id.addAndUpdate);

dateButton = findViewById(R.id.datePickerButton);

title = findViewById(R.id.name\_title);

from = findViewById(R.id.from);

description = findViewById(R.id.description);

province = findViewById(R.id.Budget);

gpa = findViewById(R.id.max\_limit);

dateButton.setText(getTodaysDate());

ApplicationDBHelper applicationDBHelper = new ApplicationDBHelper(Scholarship\_view\_Activity.this);

String Degree[] = {"Select Event","BS" , "BSC" , "MS" , "MSC" , "Mphil" , "PHD" , "Inermediate" };

degspin = findViewById(R.id.evspinner);

ArrayAdapter degadapter = new ArrayAdapter<String>(this, android.R.layout.simple\_dropdown\_item\_1line, Degree);

degspin.setAdapter(degadapter);

degspin.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {

if(position==0){

onNothingSelected(parent);

addAndUpdateBtn.setEnabled(false);

}

else{

SelectedDegree = Degree[position];

addAndUpdateBtn.setEnabled(true);

}

}

@Override

public void onNothingSelected(AdapterView<?> parent) {

addAndUpdateBtn.setEnabled(false);

}

});

if(getIntent().getExtras() != null){

Log.i("EVENT" , "UPDATION STEP");

Bundle bundel = getIntent().getExtras();

id = bundel.getString("id");

ScholarshipDBHelper scholarshipDbHelper = new ScholarshipDBHelper(Scholarship\_view\_Activity.this);

Scholarship scholarship = scholarshipDbHelper.getData(Integer.parseInt(id));

title.setText(scholarship.getTitle());

from.setText(scholarship.getOrganizer());

description.setText(scholarship.getDescription());

province.setText(scholarship.getDegree());

gpa.setText(scholarship.getCgpa());

}

addAndUpdateBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String Title = title.getText().toString();

String organizer = from.getText().toString();

String limit = gpa.getText().toString();

String budget = province.getText().toString();

String Description = description.getText().toString();

if(Title.isEmpty()){

ToastShow("Title");

}

else if(organizer.isEmpty()){

ToastShow("Organizer");

}

else if(limit.isEmpty()){

ToastShow("Limit ");

}

else if(budget.isEmpty()){

ToastShow("budget");

}

else if(Description.isEmpty()){

ToastShow("Description");

}

ScholarshipDBHelper scholarshipDbHelper = new ScholarshipDBHelper(Scholarship\_view\_Activity.this);

if(getIntent().getExtras() != null ){

if(scholarshipDbHelper.updateEvent(Integer.parseInt(id) , Title , organizer , SelectedDegree , budget , Date , Description ,limit ))

{

Log.i("EVENT" , "UPDATED");

}

else{

Log.i("EVENT" , "UPDATION FAILED");

}

}

else {

if (scholarshipDbHelper.insertEvent(Title, organizer, SelectedDegree, budget, Date, Description, limit)) {

Log.i("EVENT", "Event Added");

}

else{

Log.i("EVENT", "Event Addition Failed");

}

}

}

});

}

private void writedata() {

String json;

InputStream is = null;

try {

is = getAssets().open("Event.json");

File extdir1 = Environment.getExternalStorageDirectory();

File myDatadir1 = new File(extdir1 , "/testing");

if(!myDatadir1.exists()){

Toast.makeText(this , "File dose not exist",Toast.LENGTH\_LONG).show();

}

File myfile1 = new File(myDatadir1 , "Event.json");

FileInputStream fis = new FileInputStream(myfile1);

int size = fis.available();

byte[] buffer = new byte[size];

fis.read(buffer);

fis.close();

json = new String(buffer , "UTF-8");

JSONArray jsonarray = new JSONArray(json);

JSONObject obj = new JSONObject();

obj.put("Title", title.getText().toString());

obj.put("From", from.getText().toString());

obj.put("Description", description.getText().toString());

obj.put("Province", province.getText().toString());

obj.put("Date", Date );

//Log.i("MYTAG" , Date);

obj.put("Degree", SelectedDegree);

jsonarray.put(obj);

File extdir = Environment.getExternalStorageDirectory();

File myDatadir = new File(extdir , "/testing");

if(!myDatadir.exists()){

myDatadir.mkdir();

}

File myfile = new File(myDatadir , "Event.json");

try {

FileOutputStream fos = new FileOutputStream(myfile);

fos.write(jsonarray.toString().getBytes());

finish();

} catch (FileNotFoundException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

} catch (IOException e) {

e.printStackTrace();

} catch (JSONException e) {

e.printStackTrace();

}

}

private void LessGpa() {

Toast.makeText(this , "GPA must be greater than 2.5",Toast.LENGTH\_LONG).show();

}

// Coding for Date Picker

private String getTodaysDate()

{

Calendar cal = Calendar.getInstance();

int year = cal.get(Calendar.YEAR);

int month = cal.get(Calendar.MONTH);

month = month + 1;

int day = cal.get(Calendar.DAY\_OF\_MONTH);

return makeDateString(day, month, year);

}

private void initDatePicker()

{

DatePickerDialog.OnDateSetListener dateSetListener = new DatePickerDialog.OnDateSetListener()

{

@Override

public void onDateSet(DatePicker datePicker, int year, int month, int day)

{

month = month + 1;

String date = makeDateString(day, month, year);

Date = date;

dateButton.setText(date);

}

};

Calendar cal = Calendar.getInstance();

int year = cal.get(Calendar.YEAR);

int month = cal.get(Calendar.MONTH);

int day = cal.get(Calendar.DAY\_OF\_MONTH);

int style = AlertDialog.THEME\_HOLO\_LIGHT;

datePickerDialog = new DatePickerDialog(this, style, dateSetListener, year, month, day);

//datePickerDialog.getDatePicker().setMaxDate(System.currentTimeMillis());

}

private String makeDateString(int day, int month, int year)

{

return getMonthFormat(month) + " " + day + " " + year;

}

private String getMonthFormat(int month)

{

if(month == 1)

return "JAN";

if(month == 2)

return "FEB";

if(month == 3)

return "MAR";

if(month == 4)

return "APR";

if(month == 5)

return "MAY";

if(month == 6)

return "JUN";

if(month == 7)

return "JUL";

if(month == 8)

return "AUG";

if(month == 9)

return "SEP";

if(month == 10)

return "OCT";

if(month == 11)

return "NOV";

if(month == 12)

return "DEC";

//default should never happen

return "JAN";

}

public void openDatePicker(View view)

{

datePickerDialog.show();

}

public void ToastShow(String field){

Toast.makeText(this , field+" is required",Toast.LENGTH\_LONG).show();

}

}

## Scholarship\_Adapter

package com.example.ems;

import android.content.Context;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseAdapter;

import android.widget.Button;

import android.widget.TextView;

import java.util.List;

public class ScholarshipAdapter extends BaseAdapter {

List<Scholarship> scholarshipList;

public ScholarshipAdapter(List<Scholarship> scholarshipList) {

this.scholarshipList = scholarshipList;

}

@Override

public int getCount() {

return scholarshipList.size();

}

@Override

public Object getItem(int position) {

return scholarshipList.get(position);

}

@Override

public long getItemId(int position) {

return position;

}

@Override

public View getView(int position, View convertView, ViewGroup parent) {

//View v = convertView;

Log.i("EVENT" , scholarshipList.get(position).getId());

String id = scholarshipList.get(position).getId();

Bundle bundle = new Bundle();

bundle.putString("id" , id);

LayoutInflater layout = LayoutInflater.from(parent.getContext());

View view = layout.inflate(R.layout.eventlist , parent , false);

((TextView) view.findViewById(R.id.txt1)).setText(scholarshipList.get(position).getTitle()+" by "+ scholarshipList.get(position).getOrganizer());

Button Read , Update , Delete;

// Binding View Button

Read= view.findViewById(R.id.btnview);

Read.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Context context = v.getContext();

Intent intent = new Intent(context, Scholarship\_One\_Record.class);

intent.putExtras(bundle);

context.startActivity(intent);

}

});

Update = view.findViewById(R.id.uptbtn);

Update.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Context context = v.getContext();

Intent intent = new Intent(context, Scholarship\_view\_Activity.class);

intent.putExtras(bundle);

context.startActivity(intent);

}

});

if(position >=0) {

Delete = view.findViewById(R.id.delbtn);

Delete.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Log.i("EVENT", "DELETING");

Context context = v.getContext();

ScholarshipDBHelper scholarshipDbHelper = new ScholarshipDBHelper(context);

if (scholarshipDbHelper.deleteEvent(Integer.parseInt(id)) != 0) {

removeViewAt(position);

} else {

Log.i("EVENT", "DELETION FAILED");

}

}

});

}

return view;

}

public void removeViewAt(int position){

scholarshipList.remove(position);

notifyDataSetChanged();

}

}

## Db\_adapter.java

package com.example.ems;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteOpenHelper;

import android.database.sqlite.SQLiteDatabase;

import android.util.Log;

public class ScholarshipDBHelper extends SQLiteOpenHelper {

public static final String DATABASE\_NAME = "MyDBName2.db";

public static final String SCHOLARSHIP\_TABLE\_NAME = "Scholarship";

public static final String SCHOLARSHIP\_COLUMN\_ID = "id";

public static final String SCHOLARSHIP\_COLUMN\_TITLE = "title";

public static final String SCHOLARSHIP\_COLUMN\_ORGANIZER = "organizer";

public static final String SCHOLARSHIP\_COLUMN\_DEGREE = "degree";

public static final String SCHOLARSHIP\_COLUMN\_DATE = "Date";

public static final String SCHOLARSHIP\_COLUMN\_PROVINCE = "province";

public static final String SCHOLARSHIP\_COLUMN\_CGPA = "cgpa";

public static final String SCHOLARSHIP\_COLUMN\_DESC = "Description";

private HashMap hp;

public ScholarshipDBHelper(Context context) {

super(context, DATABASE\_NAME , null, 1);

}

@Override

public void onCreate(SQLiteDatabase db) {

// TODO Auto-generated method stub

db.execSQL(

"create table Scholarship " +

"(id integer primary key AUTOINCREMENT , title text,organizer text,event text,budget text, Date text, Description text , max\_limit text)"

);

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

// TODO Auto-generated method stub

db.execSQL("DROP TABLE IF EXISTS EVENTS");

onCreate(db);

}

public boolean insertEvent (String title, String organizer,String event, String budget, String Date,String Description ,String max\_limit) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put(SCHOLARSHIP\_COLUMN\_TITLE, title);

contentValues.put(SCHOLARSHIP\_COLUMN\_ORGANIZER, organizer);

contentValues.put(SCHOLARSHIP\_COLUMN\_PROVINCE, budget);

contentValues.put(SCHOLARSHIP\_COLUMN\_DATE, Date);

contentValues.put(SCHOLARSHIP\_COLUMN\_CGPA, max\_limit);

contentValues.put(SCHOLARSHIP\_COLUMN\_DESC, Description);

contentValues.put(SCHOLARSHIP\_COLUMN\_DEGREE, event);

db.insert(SCHOLARSHIP\_TABLE\_NAME, null, contentValues);

return true;

}

public Scholarship getData(int Id) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select \* from EVENT where id="+Id, null );

res.moveToFirst();

String id = res.getString((res.getColumnIndex("id")));

String title = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_TITLE));

String organizer = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_ORGANIZER));

String description = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_DESC));

String budget = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_PROVINCE));

String event\_type = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_DEGREE));

String date = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_DATE));

String max\_limit = res.getString(res.getColumnIndex(EVENT\_COLUMN\_LIMIT));

// craeting Event oject to store the info. reterieved from database

Scholarship scholarship = new Scholarship(title , organizer , description , max\_limit , date , budget , event\_type, id);

return scholarship;

}

/\*

public int numberOfRows(){

SQLiteDatabase db = this.getReadableDatabase();

int numRows = (int) DatabaseUtils.queryNumEntries(db, CONTACTS\_TABLE\_NAME);

return numRows;

}

\*/

public Integer deleteEvent (Integer id) {

SQLiteDatabase db = this.getWritableDatabase();

return db.delete("EVENT",

"id = ? ",

new String[] { Integer.toString(id) });

}

public boolean updateEvent (Integer id, String title, String organizer,String event, String budget, String Date,String Description ,String max\_limit) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put(SCHOLARSHIP\_COLUMN\_TITLE, title);

contentValues.put(SCHOLARSHIP\_COLUMN\_ORGANIZER, organizer);

contentValues.put(SCHOLARSHIP\_COLUMN\_PROVINCE, budget);

contentValues.put(SCHOLARSHIP\_COLUMN\_DATE, Date);

contentValues.put(EVENT\_COLUMN\_LIMIT, max\_limit);

contentValues.put(SCHOLARSHIP\_COLUMN\_DESC, Description);

contentValues.put(SCHOLARSHIP\_COLUMN\_DEGREE, event);

db.update("EVENT", contentValues, "id = ? ", new String[] { Integer.toString(id) } );

return true;

}

public List<Scholarship> getAllEvents() {

//ArrayList<String> array\_list = new ArrayList<String>();

List<Scholarship> scholarshipList = new ArrayList<>();

//hp = new HashMap();

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select \* from EVENT", null );

res.moveToFirst();

while(res.isAfterLast() == false){

String id = res.getString((res.getColumnIndex("id")));

String title = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_TITLE));

String organizer = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_ORGANIZER));

String description = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_DESC));

String budget = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_PROVINCE));

String event\_type = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_DEGREE));

String date = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_DATE));

String max\_limit = res.getString(res.getColumnIndex(EVENT\_COLUMN\_LIMIT));

Log.i("EVENT" , title );

Scholarship scholarship = new Scholarship(title , organizer , description, max\_limit , date , budget , event\_type , id );

scholarshipList.add(scholarship);

res.moveToNext();

}

return scholarshipList;

}

public ArrayList<String> getNames(){

ArrayList<String> Titles = new ArrayList<String>();

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select title from EVENT", null );

res.moveToFirst();

while(res.isAfterLast() == false){

String title = res.getString(res.getColumnIndex(SCHOLARSHIP\_COLUMN\_TITLE));

Titles.add(title);

res.moveToNext();

}

return Titles;

}

}

## Transaction.java

package com.example.ems;

public class Transection {

private String id;

private String Name;

private String SureName;

private String Event;

private String CNIC;

private String Age;

public Transection(String id, String name, String sureName, String event, String CNIC, String age) {

this.id = id;

Name = name;

SureName = sureName;

Event = event;

this.CNIC = CNIC;

Age = age;

}

public String getEvent() {

return Event;

}

public void setEvent(String event) {

Event = event;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return Name;

}

public void setName(String name) {

Name = name;

}

public String getSureName() {

return SureName;

}

public void setSureName(String sureName) {

SureName = sureName;

}

public String getCNIC() {

return CNIC;

}

public void setCNIC(String CNIC) {

this.CNIC = CNIC;

}

public String getAge() {

return Age;

}

public void setAge(String age) {

Age = age;

}

}

## Transaction\_list.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.ListView;

import java.util.ArrayList;

import java.util.List;

public class Transection\_list\_activity extends AppCompatActivity {

List<Transection> transectionList = new ArrayList<>();

ListView partiSpinner;

Button addbtn;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.participent\_list\_activity);

TransectionDBHelper transectionDBHelper = new TransectionDBHelper(Transection\_list\_activity.this);

transectionList = transectionDBHelper.getAllParticipents();

partiSpinner = findViewById(R.id.Particpents);

// Log.i("MYTAG" , eventList.get(0).getTitle());

TransectionAdapter transectionAdapter = new TransectionAdapter(transectionList);

partiSpinner.setAdapter(transectionAdapter);

addbtn = findViewById(R.id.addPartbtn);

addbtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent newActivity = new Intent( Transection\_list\_activity.this , transection\_view\_activity.class);

startActivity(newActivity);

}

});

}

@Override

protected void onResume() {

super.onResume();

TransectionDBHelper transectionDBHelper = new TransectionDBHelper(Transection\_list\_activity.this);

Log.i("MY Participents" , "Resumed");

transectionList = transectionDBHelper.getAllParticipents();

partiSpinner = findViewById(R.id.Particpents);

// Log.i("MYTAG" , eventList.get(0).getTitle());

partiSpinner = findViewById(R.id.Particpents);

// Log.i("MYTAG" , eventList.get(0).getTitle());

TransectionAdapter transectionAdapter = new TransectionAdapter(transectionList);

partiSpinner.setAdapter(transectionAdapter);

}

## }

## Transaction\_One.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.TextView;

public class Transection\_One\_Record extends AppCompatActivity {

TextView name , sureName , age , cnic , event;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.participent\_one\_record);

Bundle bundel = getIntent().getExtras();

String id = bundel.getString("id");

TransectionDBHelper transectionDBHelper = new TransectionDBHelper(Transection\_One\_Record.this);

Transection transection = transectionDBHelper.getData(Integer.parseInt(id));

name = findViewById(R.id.First\_Name);

name.setText(transection.getName());

sureName = findViewById(R.id.Sure\_Name);

sureName.setText(transection.getSureName());

age = findViewById(R.id.AgE);

age.setText(transection.getAge());

cnic = findViewById(R.id.CniC);

cnic.setText(transection.getCNIC());

event = findViewById(R.id.EveNt);

event.setText(transection.getEvent());

}

}

## Transaction\_View.java

package com.example.ems;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.Spinner;

import android.widget.TextView;

import android.widget.Toast;

public class transection\_view\_activity extends AppCompatActivity {

TextView Name , sureName , Age , CNIC ;

Spinner TransectionSpinner;

Button finishbtn;

String id , event;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.participent\_view\_activity);

Name = findViewById(R.id.parti\_name);

sureName = findViewById(R.id.parti\_sure\_name);

Age = findViewById(R.id.parti\_age);

CNIC = findViewById(R.id.cnic);

EventSpinner = findViewById(R.id.parti\_event\_spinner);

finishbtn = findViewById(R.id.AddUpdatePartiBtn);

String events[] = {"Select..." , "Scholarship1","Scholarship2","Scholarship3","Scholarship4"};

ArrayAdapter evetsadpater = new ArrayAdapter<String>(this, android.R.layout.simple\_dropdown\_item\_1line, events);

EventSpinner.setAdapter(evetsadpater);

EventSpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {

if(position==0){

onNothingSelected(parent);

finishbtn.setEnabled(false);

}

else{

event = events[position];

finishbtn.setEnabled(true);

}

}

@Override

public void onNothingSelected(AdapterView<?> parent) {

finishbtn.setEnabled(false);

}

});

finishbtn = findViewById(R.id.AddUpdatePartiBtn);

if(getIntent().getExtras() != null){

Log.i("EVENT" , "UPDATION STEP");

Bundle bundel = getIntent().getExtras();

id = bundel.getString("id");

TransectionDBHelper transectionDBHelper = new TransectionDBHelper(transection\_view\_activity.this);

Transection transection = transectionDBHelper.getData(Integer.parseInt(id));

Name.setText(transection.getName());

sureName.setText(transection.getSureName());

Age.setText(transection.getAge());

CNIC.setText(transection.getCNIC());

}

finishbtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String name = Name.getText().toString();

String SureName = sureName.getText().toString();

String age = Age.getText().toString();

String cnic = CNIC.getText().toString();

if(name.isEmpty()){

ToastShow("First Name");

}

else if(SureName.isEmpty()){

ToastShow("Sure Name");

}

else if(age.isEmpty()){

ToastShow(" Age ");

}

else if(cnic.isEmpty()){

ToastShow("CNIC");

}

TransectionDBHelper transectionDBHelper = new TransectionDBHelper(transection\_view\_activity.this);

if(getIntent().getExtras() != null ){

if(transectionDBHelper.updateParticipent(Integer.parseInt(id) , name , SureName , age , event , cnic ))

{

Log.i("Participent" , "UPDATED");

}

else{

Log.i("Participent" , "UPDATION FAILED");

}

}

else {

if (transectionDBHelper.insertParticipent( name , SureName , age , event , cnic )) {

Log.i("Participent", "Hall Added");

}

else{

Log.i("Participent", "Hall Addition Failed");

}

}

}

});

}

public void ToastShow(String field){

Toast.makeText(this , field+" is required",Toast.LENGTH\_LONG).show();

}

}

## Transaction\_Adap.java

package com.example.ems;

import android.content.Context;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseAdapter;

import android.widget.Button;

import android.widget.TextView;

import java.util.List;

public class TransectionAdapter extends BaseAdapter {

public TransectionAdapter(List<Transection> transectionList) {

this.transectionList = transectionList;

}

List<Transection> transectionList;

@Override

public int getCount() {

return 0;

}

@Override

public Object getItem(int position) {

return null;

}

@Override

public long getItemId(int position) {

return 0;

}

@Override

public View getView(int position, View convertView, ViewGroup parent) {

String id = transectionList.get(position).getId();

Bundle bundle = new Bundle();

bundle.putString("id" , id);

LayoutInflater layout = LayoutInflater.from(parent.getContext());

View view = layout.inflate(R.layout.participent\_list , parent , false);

((TextView) view.findViewById(R.id.participent\_txt)).setText(transectionList.get(position).getName()+" "+ transectionList.get(position).getSureName());

Button Read , Update , Delete;

// Binding View Button

Read= view.findViewById(R.id.parti\_view\_btn);

Read.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Context context = v.getContext();

Intent intent = new Intent(context, transection\_view\_activity.class);

intent.putExtras(bundle);

context.startActivity(intent);

}

});

Update = view.findViewById(R.id.parti\_update\_btn);

Update.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Context context = v.getContext();

Intent intent = new Intent(context, transection\_view\_activity.class);

intent.putExtras(bundle);

context.startActivity(intent);

}

});

if(position >=0) {

Delete = view.findViewById(R.id.parti\_del\_btn);

Delete.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Log.i("EVENT", "DELETING");

Context context = v.getContext();

TransectionDBHelper transectionDBHelper = new TransectionDBHelper(context);

if (transectionDBHelper.deleteParticipent(Integer.parseInt(id)) != 0) {

removeViewAt(position);

} else {

Log.i("EVENT", "DELETION FAILED");

}

}

});

}

return view;

}

public void removeViewAt(int position){

transectionList.remove(position);

notifyDataSetChanged();

}

}

## Transaction\_DBHelper.java

package com.example.ems;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

public class TransectionDBHelper extends SQLiteOpenHelper {

public static final String DATABASE\_NAME = "MyDBName1.db";

public static final String PARTICIPENT\_TABLE\_NAME = "PARTICIPENTS";

public static final String PARTICIPENT\_COLUMN\_ID = "id";

public static final String PARTICIPENT\_COLUMN\_Name = "name";

public static final String PARTICIPENT\_COLUMN\_SURENAME = "sureName";

public static final String PARTICIPENT\_COLUMN\_Age = "age";

public static final String PARTICIPENT\_COLUMN\_EVENT = "event";

public static final String PARTICIPENT\_COLUMN\_CNIC = "cnic";

private HashMap hp;

public TransectionDBHelper(Context context) {

super(context, DATABASE\_NAME , null, 1);

}

@Override

public void onCreate(SQLiteDatabase db) {

// TODO Auto-generated method stub

db.execSQL(

"create table PARTICIPENTS" +

"(id integer primary key AUTOINCREMENT , name text,sureName text,age text,event text,cnic text)"

);

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

// TODO Auto-generated method stub

db.execSQL("DROP TABLE IF EXISTS PARTICIPENT");

onCreate(db);

}

public boolean insertParticipent (String name, String surename,String age,String event,String cnic) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put(PARTICIPENT\_COLUMN\_Age, age);

contentValues.put(PARTICIPENT\_COLUMN\_CNIC, cnic);

contentValues.put(PARTICIPENT\_COLUMN\_EVENT, event);

contentValues.put(PARTICIPENT\_COLUMN\_SURENAME, surename);

contentValues.put(PARTICIPENT\_COLUMN\_Name, name);

db.insert(PARTICIPENT\_TABLE\_NAME, null, contentValues);

return true;

}

public Transection getData(int Id) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select \* from PARTICIPENTS where id="+Id, null );

res.moveToFirst();

String id = res.getString((res.getColumnIndex("id")));

String name = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_Name)));

String sureName = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_SURENAME)));

String age = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_Age)));

String event = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_EVENT)));

String cnic = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_CNIC)));

// craeting Event oject to store the info. reterieved from database

Transection transection = new Transection(id , name, sureName , event , age, cnic);

return transection;

}

/\*

public int numberOfRows(){

SQLiteDatabase db = this.getReadableDatabase();

int numRows = (int) DatabaseUtils.queryNumEntries(db, CONTACTS\_TABLE\_NAME);

return numRows;

}

\*/

public Integer deleteParticipent (Integer id) {

SQLiteDatabase db = this.getWritableDatabase();

return db.delete("PARTICIPENTS",

"id = ? ",

new String[] { Integer.toString(id) });

}

public boolean updateParticipent (Integer id, String name, String surename,String age,String event,String cnic) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put(PARTICIPENT\_COLUMN\_Age, age);

contentValues.put(PARTICIPENT\_COLUMN\_CNIC, cnic);

contentValues.put(PARTICIPENT\_COLUMN\_EVENT, event);

contentValues.put(PARTICIPENT\_COLUMN\_SURENAME, surename);

contentValues.put(PARTICIPENT\_COLUMN\_Name, name);

db.update("PARTICIPENTS", contentValues, "id = ? ", new String[] { Integer.toString(id) } );

return true;

}

public List<Transection> getAllParticipents() {

//ArrayList<String> array\_list = new ArrayList<String>();

List<Transection> transectionList = new ArrayList<>();

//hp = new HashMap();

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select \* from PARTICIPENTS", null );

res.moveToFirst();

while(res.isAfterLast() == false){

String id = res.getString((res.getColumnIndex("id")));

String name = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_Name)));

String sureName = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_SURENAME)));

String age = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_Age)));

String event = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_EVENT)));

String cnic = res.getString((res.getColumnIndex(PARTICIPENT\_COLUMN\_CNIC)));

// craeting Event oject to store the info. reterieved from database

Transection transection = new Transection(id , name, sureName , event , age, cnic);

transectionList.add(transection);

res.moveToNext();

}

return transectionList;

}

/\*

public ArrayList<String> getNames(){

ArrayList<String> Titles = new ArrayList<String>();

SQLiteDatabase db = this.getReadableDatabase();

Cursor res = db.rawQuery( "select title from EVENT", null );

res.moveToFirst();

while(res.isAfterLast() == false){

String title = res.getString(res.getColumnIndex(EVENT\_COLUMN\_TITLE));

Titles.add(title);

res.moveToNext();

}

return Titles;

}

\*/

}