

blackrgb0,0.6,0 greenrgb0.5,0.5,0.5 cyanrgb0.58,0,0.82 redrgb0.58,0,0.82

CODIGO FUENTE

Kelvin Paul Pucho Zevallos
Angelo Perez Rodriguez

Julio 2019

1 La clase BASE DE DATOS

Creamos la base de datos para sus posteriores ingresos por medio de String-Builder sql.

```
1 //administrar la base de datos
2 public class BDcontenido extends SQLiteOpenHelper {
3
4
5     public BDcontenido(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {
6         super(context, name, factory, version); // esto debe ser indicado desde la clase BDcontenido
7         //super(context, "AEROLINEAS", null, 1);
8     }
9
10    @Override
11    public void onCreate(SQLiteDatabase BaseDeDatos) { //constructor el nombre de mi base de datos=BasedeDatos
12
13        StringBuilder sql = new StringBuilder();
14
15        sql.append("CREATE TABLE IF NOT EXISTS AVIONES(");
16        sql.append("ASIENTOS VARCHAR(20) PRIMARY KEY,");
17        sql.append("CLASES VARCHAR(20),");
18        sql.append("PROCEDS VARCHAR(20),");
19        sql.append("DESTINOS VARCHAR(20),");
20        sql.append("NOMBRES VARCHAR(30),");
21        sql.append("DNIS VARCHAR(8),");
22        sql.append("FECHAS VARCHAR(8),");
23        sql.append("HORASG VARCHAR(8),");
24        sql.append("LUGARS VARCHAR(20),");
25        sql.append("POSASIENTO VARCHAR(20),");
26        sql.append("PRECIOS VARCHAR(20)");
27
28        BaseDeDatos.execSQL(sql.toString());
```

29 }

2 La clase Aviones

En esta clase encontramos el código de acceso a los datos según las validaciones echas dentro del código.

```
1  @Override
2  public boolean onOptionsItemSelected(MenuItem item){
3  int id = item.getItemId();
4      switch (id){
5          case R.id.Registrar:
6              if(verificar()== true){
7                  try{
8                      BDcontenido conexion = new BDcontenido(this,"
9                      AEROLINEAS",null,1);
10                     //abriendo la base de datos en modo lectura y
11                     escritura
12                     SQLiteDatabase BaseDeDatos = conexion.
13                     getWritableDatabase();
14                     StringBuilder sql = new StringBuilder();
15
16                     String respuest = "";
17
18                     if(clasef.isChecked()){
19                         respuest = "F"+castRandmF;
20                     }else if(claseg.isChecked()){
21                         respuest = "G"+castRandmG;
22                     }
23
24                     String dato = respuest;
25
26                     sql.append("INSERT INTO AVIONES (ASIENTOS, CLASES,
27                     PROCEDES, DESTINOS, NOMBRES, DNIS, FECHAS,
28                     HORASG, LUGARS, POSASIENTO, PRECIOS) VALUES ('
29                     ');
30                     sql.append(asiento.getText().toString().trim() + "
31                     ', '");
32                     sql.append(clase.getText().toString().trim() + "
33                     ', '");
34                     sql.append(proced.getText().toString().trim() + "
35                     ', '");
36                     sql.append(destino.getText().toString().trim() + "
37                     ', '");
38                     sql.append(nombre.getText().toString().trim() + "
39                     ', '");
40                     sql.append(dni.getText().toString().trim() + "
41                     ', '")
```

```

        );
        sql.append(spFecha + "','');
        sql.append(spHora+ "','');
        sql.append(dato + "','');
        sql.append(asienPasa.getText().toString().trim() +
            "','');
        sql.append(CostoDolar + "','');

        Toast.makeText(this, "REGISTRO_CON_EXITO", Toast.
            LENGTH_SHORT).show();
        BaseDeDatos.execSQL(sql.toString());
        BaseDeDatos.close();
    }catch (Exception ex){
        AlertDialog.Builder dlg = new AlertDialog.Builder(
            this);
        dlg.setTitle("Aviso");
        dlg.setMessage(ex.getMessage());
        dlg.setNeutralButton("OK", null);
        dlg.show();
    }
    String AsientoN = getIntent().getStringExtra("
        eljasiento");
    if (!AsientoN.isEmpty()) {

        BDcontenido conexion = new BDcontenido(this,"
            AEROLINEAS",null,1);
        SQLiteDatabase BaseDeDatos1 = conexion.
            getWritableDatabase();

        Cursor fila = BaseDeDatos1.rawQuery("select_NOMBRES
            ,LUGARS,PROCEDES,DESTINOS from_AVIONES where_
            ASIENTOS=" + AsientoN, null);

        if (fila.moveToFirst()) {
            MosNombre.setText(fila.getString(0));
            MosClase1.setText(fila.getString(1));
            MosProceds.setText(fila.getString(2));
            MosDestino.setText(fila.getString(3));
            BaseDeDatos1.close();
        }
    }else {
        Toast.makeText(this, "por_favor,coloque_el_numero_
            del_asiento_para_ver_los_contenidos", Toast.
            LENGTH_SHORT).show();
    }
} else{
    Toast.makeText(this, "POR_FAVOR,RELLENO TODOS LOS_

```

```

70             CAMPOS", Toast.LENGTH_SHORT).show();
71         }
72         break;
73     }
74     return super.onOptionsItemSelected(item);
}

```

3 La clase ElijaAsiento

Fue la mas complicada debido a que son 80 asientos con sus respectivos tipo de dato button para elejir el asiento del pasajero

```

1 public class ElijaAsiento extends AppCompatActivity {
2
3     private Button boton1, boton2, boton3, boton4, boton5, ..., boton80
4     @Override
5     protected void onCreate(Bundle savedInstanceState) {
6         super.onCreate(savedInstanceState);
7         setContentView(R.layout.activity_elija_asiento);
8         boton1 = (Button) findViewById(R.id.uno);
9         boton2 = (Button) findViewById(R.id.dos);
10        boton3 = (Button) findViewById(R.id.tres);
11        .
12        .
13        .
14        boton80 = (Button) findViewById(R.id.ochenta);
15    }
16
17    // obtenemos los asientos para luego ser enviados hacia la parte del
18    // activity donde se guardara con la base de datos
19    public void Seleccion(View view){
20        int id = view.getId();
21        if (id == R.id.uno){
22            int a=1;
23            String Nroasiento = String.valueOf(a);
24            boton1.setEnabled(false);
25            Toast.makeText(this, "Asiento Seleccionado", Toast.
26                LENGTH_SHORT).show();
27            Intent esco = new Intent(this, Aviones.class);
28            esco.putExtra("eljasiento", Nroasiento);
29            startActivity(esco);
30        }
31        else if (id == R.id.dos){
32            int a=2;
33            String Nroasiento = String.valueOf(a);
34            boton2.setEnabled(false);

```

```

35         Toast.makeText(this, "Asiento_Seleccionado", Toast.
           LENGTH_SHORT).show();
36         Intent esco = new Intent(this,Aviones.class);
37         esco.putExtra("eljasiento",Nroasiento);
38         startActivity(esco);
39         .
40         .
41         .

```

4 Las clases de Modificar,Mostrar y Eliminar

```

1
2
3 //MODIFICAR
4 case R.id.Modificar:
5
6         BDcontenido conexion = new BDcontenido(this, "AEROLINEAS",
           null, 1);
7         SQLiteDatabase BaseDatabase = conexion.getWritableDatabase
           ();
8
9
10        String Proceds = result3.getText().toString();
11        String Destinos= result4.getText().toString();
12        String Nombres= result5.getText().toString();
13        String Dnis= result6.getText().toString();
14
15        if(Verificar() == true){
16
17                ContentValues registro = new ContentValues();
18                registro.put("PROCEEDS", Proceds);
19                registro.put("DESTINOS", Destinos);
20                registro.put("NOMBRES", Nombres);
21                registro.put("DNIS", Dnis);
22                registro.put("FECHAS", spFecha);
23                registro.put("HORASG", spHora);
24
25                int actualizar = BaseDatabase.update("AVIONES",
           registro, "ASIENTOS=" + AsientoN, null);
26                BaseDatabase.close();
27
28                if(actualizar == 1){
29                        Toast.makeText(this, "Pasaje_modificado", Toast.
           LENGTH_SHORT).show();
30                } else {
31                        Toast.makeText(this, "El_asiento_esta_desocupado,
           puede_registrarse", Toast.LENGTH_SHORT).show();

```

```

32         }
33
34     } else {
35         Toast.makeText(this, "Debes llenar todos los campos",
36             Toast.LENGTH_SHORT).show();
37     }
38     break;
39 //ELIMINAR
40 public void Eliminar(View view){
41     String NumberAsien = NroAsiento.getText().toString();
42
43     try{
44         BDcontenido conexion = new BDcontenido(this,"AEROLINEAS",null
45             ,1);
46         SQLiteDatabase BaseDeDatos = conexion.getWritableDatabase();
47
48         if(!NumberAsien.isEmpty()){
49             int borrar = BaseDeDatos.delete("AVIONES", "ASIENTOS=" +
50                 NumberAsien,null);
51             BaseDeDatos.close();
52
53             if(borrar == 1){
54                 Toast.makeText(this, "El asiento esta ocupado y esta
55                     eliminado", Toast.LENGTH_SHORT).show();
56             }else{
57                 Toast.makeText(this, "El asiento esta desocupado",
58                     Toast.LENGTH_SHORT).show();
59             }
60         }else{
61             Toast.makeText(this, "Introduce el Nro del asiento", Toast
62                 .LENGTH_SHORT).show();
63         }
64     }
65     }catch (Exception ex){
66         AlertDialog.Builder dlg = new AlertDialog.Builder(this);
67         dlg.setTitle("Aviso");
68         dlg.setMessage(ex.getMessage());
69         dlg.setNeutralButton("OK",null);
70         dlg.show();
71     }
72     finish();
73 //MOSTRAR
74 private void ActualizarPasajeros() {
75     Datoslistados = (ListView) findViewById(R.id.datoslist);
76     pasajero = new ArrayList<String>();
77
78     try{

```

```

75         BDcontenido conexion = new BDcontenido(this,"AEROLINEAS",null
76             ,1);
77         SQLiteDatabase BaseDeDatos = conexion.getWritableDatabase();
78
79         StringBuilder sql = new StringBuilder();//generador de cadenas
80         sql.append("SELECT_*_FROM_AVIONES");
81         String sAsiento;
82         String sNombre;
83         String sLugar;
84         String sProcedencia;
85         String sDestinos;
86         String sFechas;
87         String sHoras;
88         String sPrecio;
89
90         Cursor resultado = BaseDeDatos.rawQuery(sql.toString(), null);
91         System.out.println(resultado.getCount());
92
93         if (resultado.getCount() > 0 ){
94             resultado.moveToFirst();
95             do{
96                 sAsiento = resultado.getString(resultado.getColumnIndex
97                     ("ASIENTOS"));
98                 sNombre = resultado.getString(resultado.
99                     getColumnIndexOrThrow("NOMBRES"));
100                 sLugar = resultado.getString(resultado.
101                     getColumnIndexOrThrow("LUGARS"));
102                 sProcedencia = resultado.getString(resultado.
103                     getColumnIndexOrThrow("PROCEEDS"));
104                 sDestinos = resultado.getString(resultado.
105                     getColumnIndexOrThrow("DESTINOS"));
106                 sFechas = resultado.getString(resultado.
107                     getColumnIndexOrThrow("FECHAS"));
108                 sHoras = resultado.getString(resultado.
109                     getColumnIndexOrThrow("HORASG"));
110                 sPrecio = resultado.getString(resultado.
111                     getColumnIndexOrThrow("PRECIOS"));
112
113                 pasajero.add("Asiento:N:"+sAsiento +"\n"+"Nombre:"+
114                     sNombre + "\n" + "Vuelo:"+ sLugar + "\n" + "
115                     Origen:"+ sProcedencia +"\n" + "Destino:"+
116                     sDestinos +"\n" + "Fechas de Abordaje:"+ sFechas +
117                     "\n" + "Hora deabordaje:"+ sHoras +"\n" + "
118                     Precio:"+ sPrecio);
119                 System.out.println(pasajero);
120             }
121             while(resultado.moveToNext());
122         }
123         adaptador = new ArrayAdapter<String>(this, R.layout.
124             disenodel_listview, pasajero);//R.layout.

```



```
        diseno_del_listview = escojemos el diseo de la cle R
        creado por nosotros
110     Datoslistados.setAdapter(adaptador);
111     System.out.println(adaptador);
112
113 }catch (Exception ex){
114     AlertDialog.Builder dlg = new AlertDialog.Builder(this);
115     dlg.setTitle("Aviso");
116     dlg.setMessage(ex.getMessage());
117     dlg.setNeutralButton("OK",null);
118     dlg.show();
119 }
```
