

Aula 08 – Banco de Dados

Programação em Java para a Plataforma Android



Agenda

- Introdução a linguagem SQL
- O banco de dados SQLite
- Como criar e manipular um banco de dados em Android
- Como apresentar os dados na tela do aparelho
- Como criar um log simples de eventos
- Quais as estruturas de dados mais comuns em Java



Armazenamento permanente

- Android disponibiliza um banco de dados chamado SQLite
- SQLite é o banco de dados padrão em várias outras aplicações: Firefox, iPhone, Symbia, Skype, etc.
- SQLite não possui licença, sendo de domínio público

Para usar SQLite é preciso saber um pouco de SQL. Vocês lembram de alguma query?



Comandos de definição

- Para criar uma tabela, podemos usar o comando abaixo:

```
create table tabela (  
    _id integer primary key autoincrement,  
    name text,  
    phone text  
);
```



Comandos de modificação

- Para inserir algumas entradas em nosso banco de dados de exemplo, podemos usar os comandos abaixo

insert into tabela values(null, 'Steven King', '555-1212');

insert into tabela values(null, 'Jonh Smith', '555-2345');

insert into tabela values(null, 'Fred Smithizen', '555-4321');



Comandos de consulta

*select * from tabela where _id = 3;*

select name, phone from tabela where name like “%Smith%”;



Log de eventos

- Crie em banco de dados que logue “eventos”. Um evento possui um título e o tempo em que aconteceu. Eventos são criados via um método `addEventos(titulo)`



O banco de dados

```
import static com.pininapp.bancodedados.database.Constants._ID;
import static com.pininapp.bancodedados.database.Constants.TABLE_NAME;
import static com.pininapp.bancodedados.database.Constants.TIME;
import static com.pininapp.bancodedados.database.Constants.TITLE;

public class Database extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "events.db";
    private static final int DATABASE_VERSION = 1;

    public Database(Context ctx) {
        super(ctx, DATABASE_NAME, null, DATABASE_VERSION);
    }

    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE " + TABLE_NAME + " (" + _ID
            + " INTEGER PRIMARY KEY AUTOINCREMENT, " + TIME
            + " INTEGER, " + TITLE + " TEXT NOT NULL);");
    }

    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }
}
```



Importações estáticas

```
import android.provider.BaseColumns;

public interface Constants extends BaseColumns {
    String TABLE_NAME = "events";
    String TIME = "time";
    String TITLE = "title";
}
```

Precisamos de um layout para mostrar eventos.



Layout

```
<?xml version="1.0" encoding="utf-8" ?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</ScrollView>
```

eventos_activity.xml

E como seria o
programa
principal?



O programa principal

```
public class EventosActivity extends AppCompatActivity {  
  
    private Database database;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.eventos_activity);  
        database = new Database(this);  
        try{  
            addEvent("Hello Android!");  
            Cursor cursor = getEvents();  
            showEvents(cursor);  
        } finally {  
            database.close();  
        }  
    }  
}
```

Como criar **novos**
eventos?

Como ler esses
eventos?

E como mostrar
esses eventos na
tela?



Criando eventos

```
private void addEvent(String title){  
    SQLiteDatabase db = database.getWritableDatabase();  
    ContentValues values = new ContentValues();  
    values.put(Constants.TIME, System.currentTimeMillis());  
    values.put(Constants.TITLE, title);  
    db.insertOrThrow(Constants.TABLE_NAME, null, values);  
}
```



Lendo eventos

```
private static String[] FROM = {Constants._ID,  
                                Constants.TIME,  
                                Constants.TITLE};  
  
private static String ORDER_BY = Constants.TIME + " DESC";  
  
private Cursor getEvents() {  
    SQLiteDatabase db = database.getReadableDatabase();  
    Cursor cursor = db.query(Constants.TABLE_NAME, FROM,  
                             null, null, null, null, ORDER_BY);  
    return cursor;  
}
```



Mostrando eventos na tela

```
private void showEvents(Cursor cursor) {  
    StringBuilder builder = new StringBuilder( "Saved events:\n");  
    while (cursor.moveToNext()) {  
        long id = cursor.getLong(0);  
        long hme = cursor.getLong(1);  
        String htle = cursor.getString(2);  
        builder.append(id).append(": ");  
        builder.append(hme).append(": ");  
        builder.append(htle).append("\n");  
    }  
    TextView text = (TextView) findViewById(R.id.text);  
    text.setText(builder);  
}
```

Vocês saberiam
logar a posição de
eventos de clique
na tela?



Outros eventos

```
public class EventosActivity extends AppCompatActivity
    implements View.OnTouchListener {

    ...

    @Override
    public boolean onTouch(View v, MotionEvent event) {
        addEvent("onTouch");
        showEvents(getEvents());
        return false;
    }
}
```

banco de dados

Saved events:

- 344: 1464982920988: Hello Android!
- 343: 1464982919760: Hello Android!
- 342: 1464982916784: onTouch
- 341: 1464982916693: onTouch
- 340: 1464982916603: onTouch
- 339: 1464982916512: onTouch
- 338: 1464982916407: onTouch
- 337: 1464982916273: onTouch
- 336: 1464982916149: onTouch
- 335: 1464982915303: onTouch
- 334: 1464982915206: onTouch
- 333: 1464982915113: onTouch
- 332: 1464982914982: onTouch
- 331: 1464982914883: onTouch
- 330: 1464981860130: onTouch
- 329: 1464981859994: onTouch
- 328: 1464981859840: onTouch
- 327: 1464981859655: onTouch
- 326: 1464981859442: onTouch
- 325: 1464981859265: onTouch
- 324: 1464981859053: onTouch
- 323: 1464981858891: onTouch
- 322: 1464981858722: onTouch
- 321: 1464981858549: onTouch
- 320: 1464981858367: onTouch
- 319: 1464981858199: onTouch
- 318: 1464981858026: onTouch

Associando dados a visões

- Android disponibiliza aos desenvolvedores alguns recursos gráficos para a exibição de tabelas
- Compare, por exemplo:



Dúvidas

