



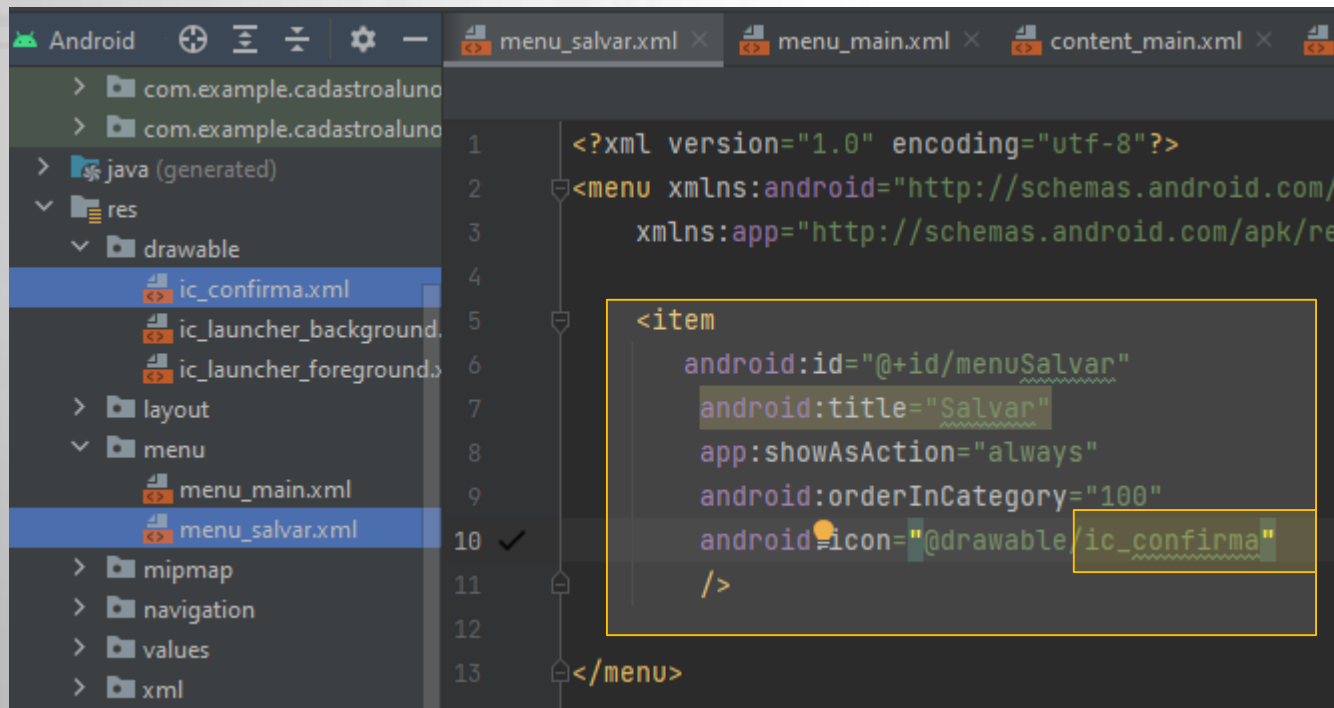
DB lista de Alunos

Prof. Hercules Santos



Layout – Botão Salvar

- 1 – Crie uma nova Activity (Formulário de cadastro)
- 2 – crie um menu para a nova Activity



Acrescente à nova Activity, o
onCreateOptionsMenu

```
public class Cadastro extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_cadastro);
    }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_salvar, menu);
        return super.onCreateOptionsMenu(menu);
    }
}
```



Menu principal – Add +

Altere o menu atual de...

para

```
<item
  android:id="@+id/action_settings"
  android:orderInCategory="100"
  android:title="@string/action_settings"
  app:showAsAction="never" />
```



```
<item
  android:id="@+id/action_settings"
  android:orderInCategory="100"
  android:title="@string/action_settings"
  app:showAsAction="always"
  android:icon="@drawable/ic_novo"/>
```

MainActivity.java

Adicione no evento de click, uma Intent para chamar a nova activity (Cadastro.class)



```
@Override
public boolean onOptionsItemSelected(MenuItem item) {

    int id = item.getItemId();

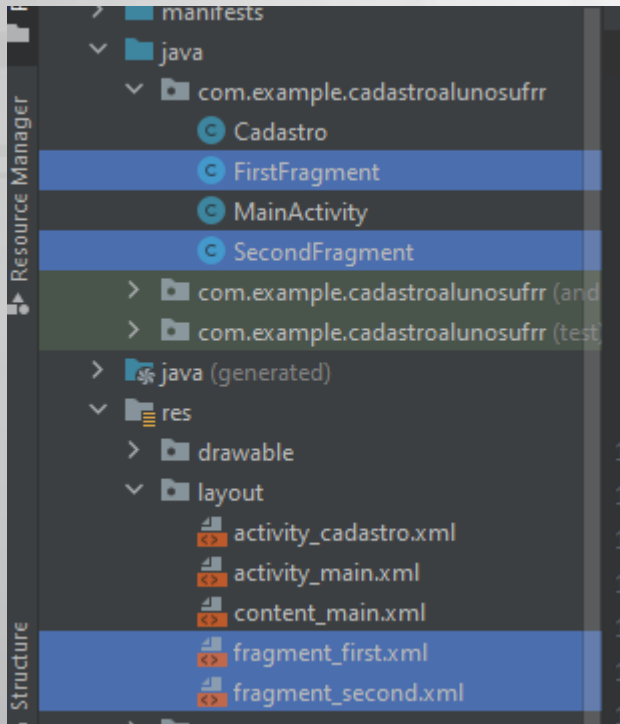
    if (id == R.id.action_settings) {
        Intent intent = new Intent(getApplicationContext(), Cadastro.class);
        startActivity(intent);
        return true;
    }

    return super.onOptionsItemSelected(item);
}
```



Fragmentos

Você pode excluir os Fragmentos (DELETE)



```
public class MainActivity extends AppCompatActivity {
```

```
    private AppBarConfiguration appBarConfiguration;  
    private ActivityMainBinding binding;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);
```

```
        binding = ActivityMainBinding.inflate(getLayoutInflater());  
        setContentView(binding.getRoot());
```

```
        setSupportActionBar(binding.toolbar);
```

```
        NavController navController = Navigation.findNavController(this, R.id.nav_host_fragment_content_main);  
        appBarConfiguration = new AppBarConfiguration.Builder(navController.getGraph()).build();  
        NavigationUI.setupActionBarWithNavController(this, navController, appBarConfiguration);
```

```
    }  
    @Override
```

```
    public boolean onCreateOptionsMenu(Menu menu) {  
        getMenuInflater().inflate(R.menu.menu_main, menu);  
        return true;
```

```
    }  
    @Override
```

```
    public boolean onOptionsItemSelected(MenuItem item) {
```

```
        int id = item.getItemId();
```

```
        if (id == R.id.action_settings) {  
            Intent intent = new Intent(getApplicationContext(), Cadastro.class);  
            startActivity(intent);  
            return true;
```

```
        }  
        return super.onOptionsItemSelected(item);
```

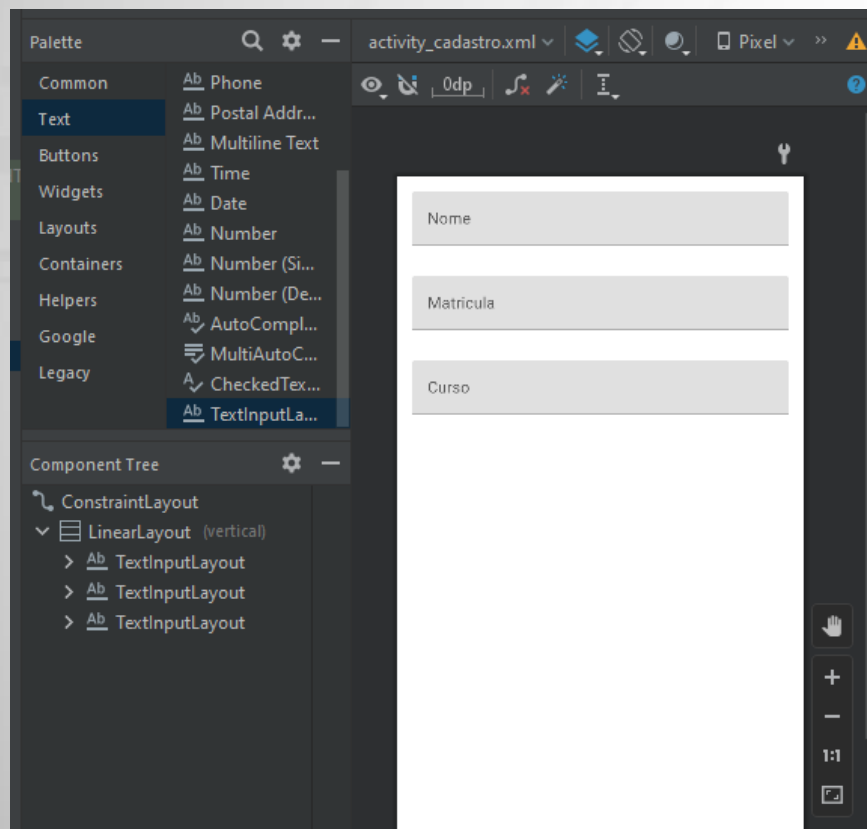
```
    }  
    @Override
```

```
    public boolean onSupportNavigateUp() {  
        NavController navController = Navigation.findNavController(this, R.id.nav_host_fragment_content_main);  
        return NavigationUI.navigateUp(navController, appBarConfiguration)  
            || super.onSupportNavigateUp();  
    }
```

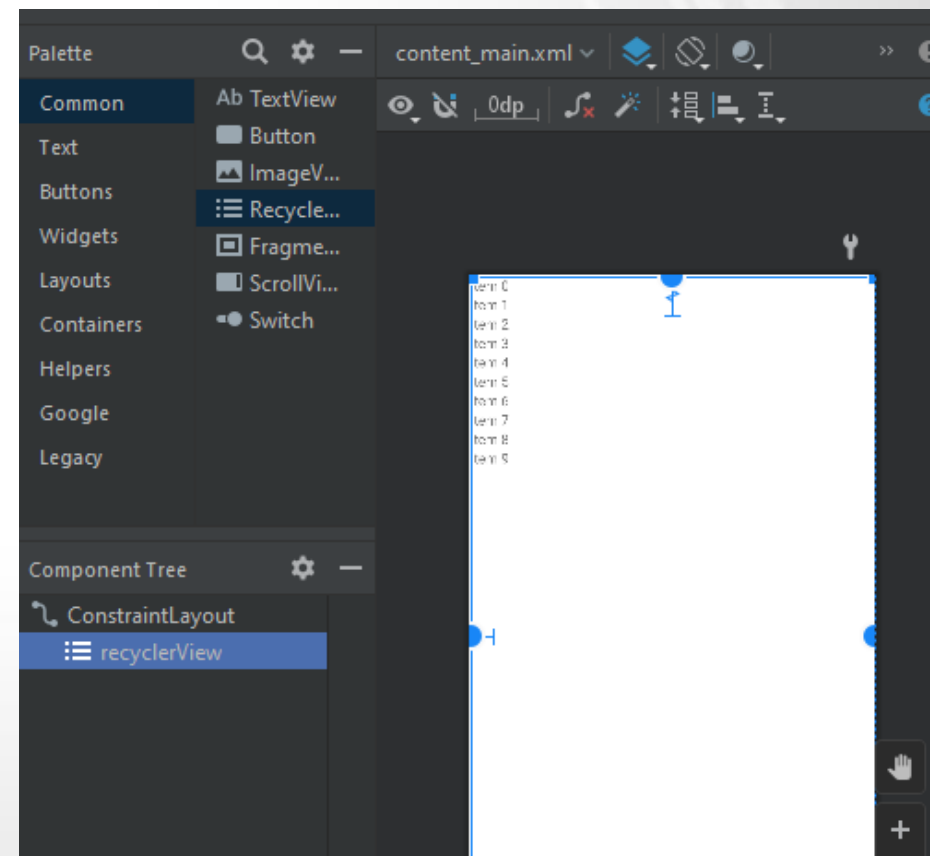
```
}
```



Formulário cadastro



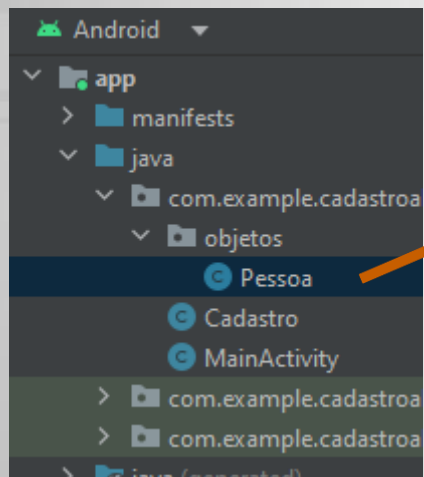
Crie o formulário para a activity_cadastro



Adicione um recyclerView em contente_main



Criar Objeto - Pessoa



```
package com.example.cadastroalunosufr.objetos;

public class Pessoa {
    String nome;
    Integer matricula;
    String curso;

    public String getNome() {return nome;}

    public void setNome(String nome) {this.nome = nome;}

    public Integer getMatricula() {return matricula;}

    public void setMatricula(Integer matricula) {this.matricula = matricula;}

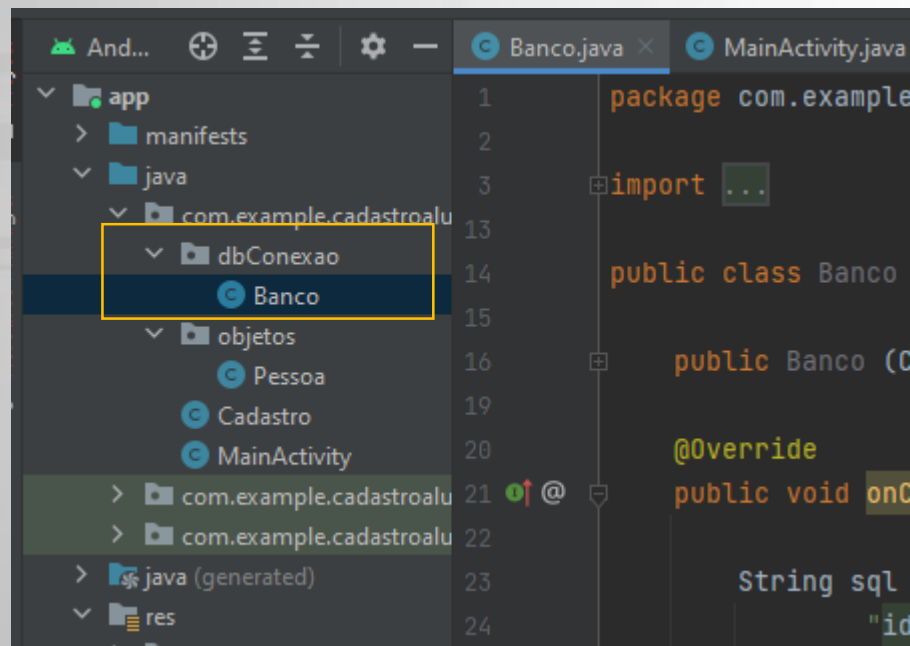
    public String getCurso() {return curso;}

    public void setCurso(String curso) {this.curso = curso;}
}
```

Crie uma Classe Pessoa, que será o objeto



Banco de dados



Crie uma classe Banco.java

Extends SQLiteOpenHelper

```
public class Banco extends SQLiteOpenHelper {  
  
    public Banco (Context context){  
        super(context, "bancoapp", null, 1);  
    }  
}
```

Contrutor

```
@Override  
public void onCreate(SQLiteDatabase sqLiteDatabase) {  
  
}  
  
@Override  
public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
  
}
```

Implemente os métodos



Métodos – Banco de Dados

```
public void onCreate(SQLiteDatabase sqLiteDatabase) {  
  
    String sql = "CREATE TABLE IF NOT EXISTS aluno (" +  
        "id INTEGER PRIMARY KEY AUTOINCREMENT," +  
        "nome VARCHAR," +  
        "matricula INT(6)," +  
        "curso VARCHAR)";  
  
    sqLiteDatabase.execSQL(sql);  
    onCreate(sqLiteDatabase);  
}
```

Método que Cria as tabelas

```
@Override  
public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
  
    String sql = "DROP TABLE IF EXISTS aluno";  
    sqLiteDatabase.execSQL(sql);  
    onCreate(sqLiteDatabase);  
}
```

Método que recria o banco caso seja necessário pra outra versão



Métodos – Banco de dados

```
public void inserirAluno (Pessoa pessoa){
    SQLiteDatabase db = getWritableDatabase();
    ContentValues dados = new ContentValues();

    dados.put("nome", pessoa.getNome());
    dados.put("matricula", pessoa.getMatricula());
    dados.put("curso", pessoa.getCurso());

    db.insert("aluno", null, dados);
}
```

Inserir dados na tabela

```
public List<Pessoa> buscarAluno(){
    SQLiteDatabase db = getReadableDatabase();
    String sqls = "SELECT * FROM aluno;";

    Cursor c = db.rawQuery(sqls, null);
    List<Pessoa> p = new ArrayList<Pessoa>();

    while (c != null && c.moveToNext()){

        int indexNome = c.getColumnIndex("nome");
        int indexMatrc = c.getColumnIndex("matricula");
        int indexCurso = c.getColumnIndex("curso");

        Pessoa alunos = new Pessoa();

        alunos.setNome(c.getString(indexNome));
        alunos.setMatricula(c.getInt(indexMatrc));
        alunos.setCurso(c.getString(indexCurso));
        p.add(alunos);
    }
    return p;
}
```

Listar dados da tabela



```
public void inserirDados(Pessoa p){
    SQLiteDatabase db = getReadableDatabase();
    ContentValues dados = new ContentValues();

    dados.put("nome", p.getNome());
    dados.put("matricula", p.getMatricula());
    dados.put("curso", p.getCurso());

    db.insert("aluno", null, dados);
}
```

Método para Inserir um cadastro

```
public List<Pessoa> buscarAluno(){
    SQLiteDatabase db = getReadableDatabase();
    String sqls = "SELECT * FROM aluno;";

    Cursor c = db.rawQuery(sqls, null);
    List<Pessoa> p = new ArrayList<Pessoa>();

    while (c != null && c.moveToNext()){

        int indexNome = c.getColumnIndex("nome");
        int indexMatrc = c.getColumnIndex("matricula");
        int indexCurso = c.getColumnIndex("curso");

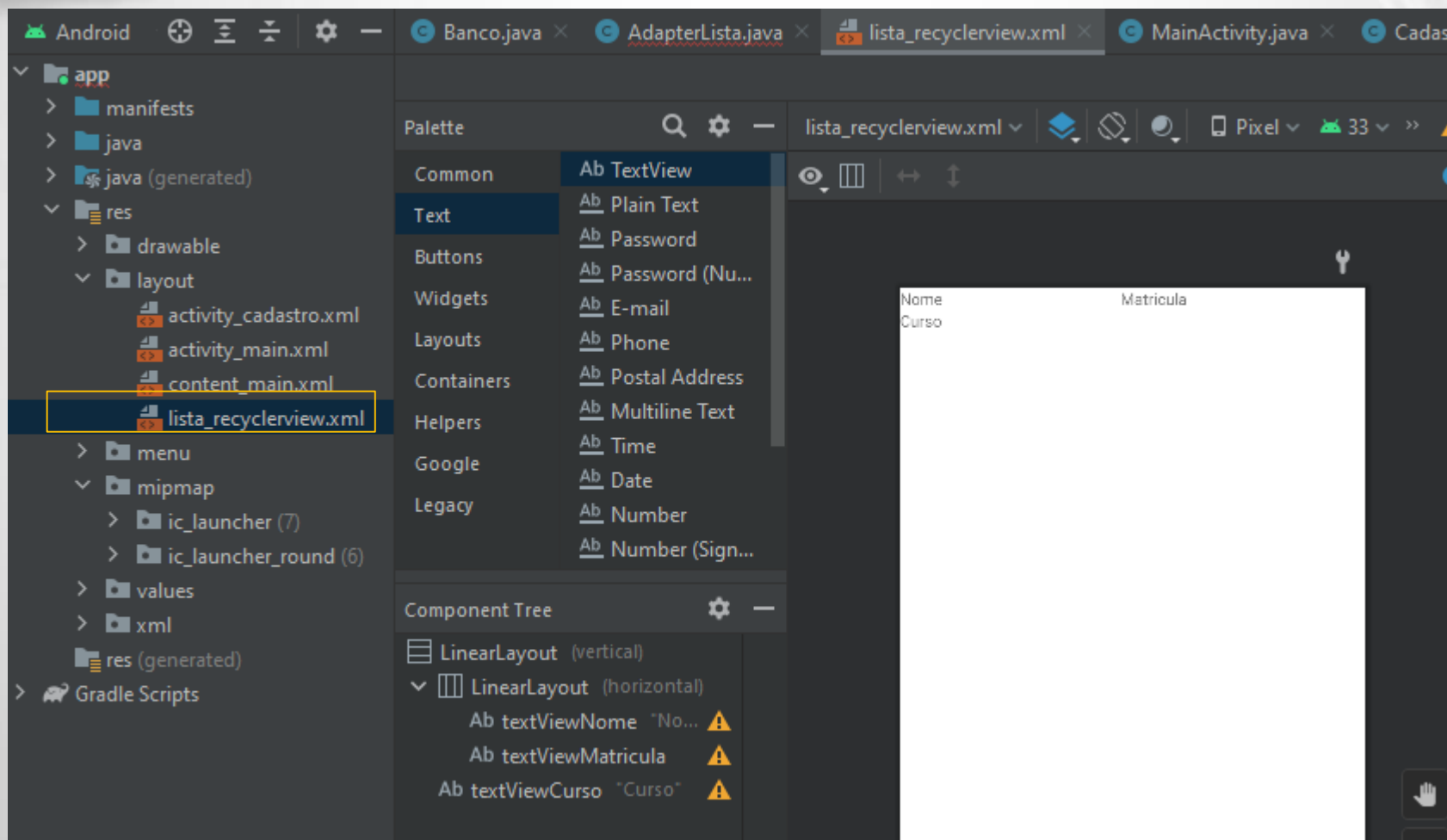
        Pessoa alunos = new Pessoa();

        alunos.setNome(c.getString(indexNome));
        alunos.setMatricula(c.getInt(indexMatrc));
        alunos.setCurso(c.getString(indexCurso));
        p.add(alunos);
    }
    return p;
}
```

Método para listar um cadastro

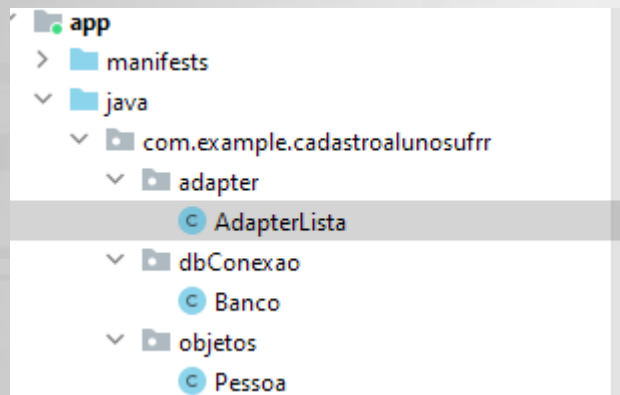


Layout RecyclerView





Adapter



Crie um Adapter para
carregar os dados

```
public class AdapterLista extends RecyclerView.Adapter<AdapterLista.AlunoViewHolder> {  
  
    private List<Pessoa> listalunos;  
  
    @Override  
    public AlunoViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
  
    }  
  
    @Override  
    public void onBindViewHolder(@NonNull AlunoViewHolder holder, int position) {  
  
    }  
  
    @Override  
    public int getItemCount() {  
        return listalunos.size();  
    }  
  
    public class AlunoViewHolder extends RecyclerView.ViewHolder {  
        TextView tvNome;  
        TextView tvMatricula;  
        TextView tvCurso;  
        public AlunoViewHolder(@NonNull View itemView) {  
            super(itemView);  
            tvNome = itemView.findViewById(R.id.textViewNome);  
            tvMatricula = itemView.findViewById(R.id.textViewMatricula);  
            tvCurso = itemView.findViewById(R.id.textViewCurso);  
        }  
    }  
}
```



Implementando os Adapter

Implemente os métodos

- onCreateViewHolder()
- onBindViewHolder()
- getItemCount()

```
public AlunoViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
  
    View itensLista = LayoutInflater.from(parent.getContext()).  
        inflate(R.layout.lista_recyclerview, parent, false);  
    return new AlunoViewHolder(itensLista);  
}
```

```
public void onBindViewHolder(@NonNull AlunoViewHolder holder, int position) {  
    Pessoa aluno = listalunos.get(position);  
    holder.tvNome.setText(aluno.getNome());  
    holder.tvMatricula.setText(aluno.getMatricula());  
    holder.tvCurso.setText(aluno.getCurso());  
}
```

```
public int getItemCount() {  
    return listalunos.size();  
}
```



Atividade

- Liste os dados cadastrados na RecyclerView
- Crie os Eventos de Clique para Abrir o Formulário de cadastro e Salvar no banco
- Os dados devem ser carregados automaticamente quando o usuário abrir o App

- Veja o Slide 08 como usar o RecyclerView
- Veja o Slide 17 como usar o Banco de Dados