

 07

Exercício 5

Altere a classe `DetalhesProvaFragment` e guarde a referência da `Prova` em um atributo:

```
public class DetalhesProvaFragment extends Fragment{
    private Prova prova;

    //... restante da classe
}
```

Agora precisamos criar a funcionalidade de mostrar os dados da prova na tela. Na classe `DetalhesProvaFragment` altere o método `onCreateView` :

```
public class DetalhesProvaFragment extends Fragment {

    private Prova prova;

    @Override
    public View onCreateView(LayoutInflater inflater,
        ViewGroup container, Bundle savedInstanceState) {

        View layout = inflater.inflate(R.layout.provas_detalhe,
            container, false);

        if (getArguments() != null) {
            this.prova = (Prova) getArguments().getSerializable("prova");
        }

        if (this.prova != null) {
            TextView materia = (TextView) layout
                .findViewById(R.id.detalhe_prova_materia);
            TextView data = (TextView) layout
                .findViewById(R.id.detalhe_prova_data);
            ListView topicos = (ListView) layout
                .findViewById(R.id.detalhe_prova_topicos);

            materia.setText(this.prova.getMateria());
            data.setText(this.prova.getData());

            ArrayAdapter<String> adapter =
                new ArrayAdapter<String>(
                    getActivity(),
                    android.R.layout.simple_list_item_1,
                    prova.getTopicos());

            topicos.setAdapter(adapter);
        }
        return layout;
    }
}
```

Para deixar o código mais fácil de ler, poderíamos criar dois métodos auxiliares: um para buscar os elementos da `view` e outro para populá-los com a `Prova` .
Como ficaria nosso código nesse caso?

Responda

INSERIR CÓDIGO	FORMATAÇÃO

