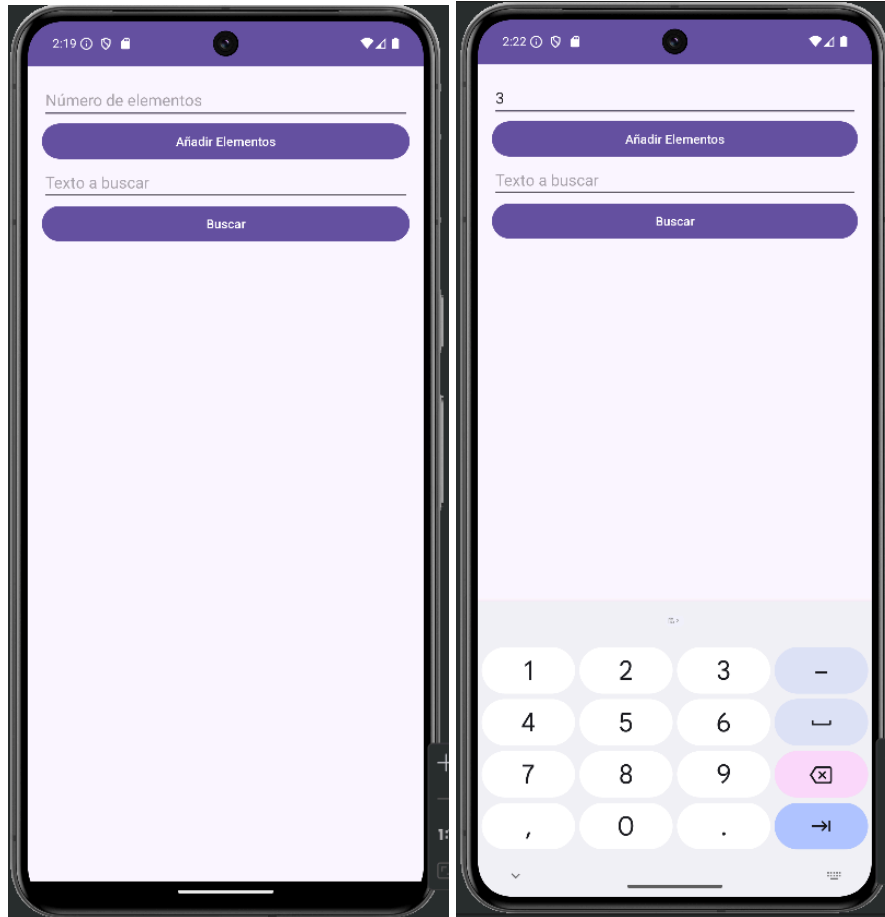


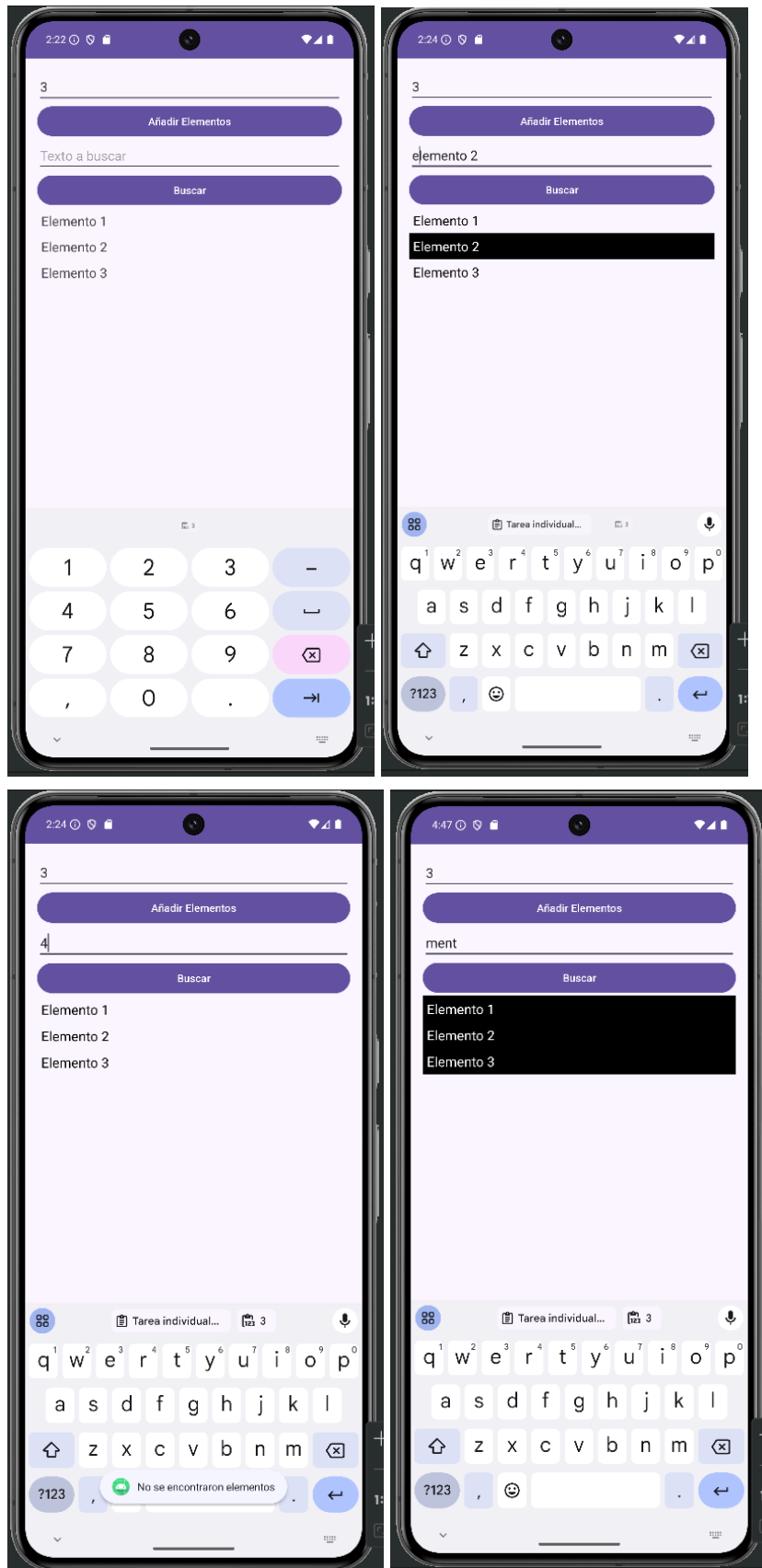
Tarea individual 3 - Actividad que inserta elementos XML desde Java

Inicio con una actividad que contiene un edit text para introducir el numero de elementos, un botón para introducirlos, otro edit text para buscarlos, y su botón para accionar la búsqueda:



Introduzco '3' y le doy a 'Añadir elementos':

Me los añade, y compruebo que hace la búsqueda correctamente, cambiando el color y su fondo al text view generado cuando lo encuentra. Si no lo encuentra, sale un Toast informando de que no se ha encontrado.



```

public class MainActivity extends AppCompatActivity {
    private Button btnSearch; 2 usages
    private LinearLayout containerLayout; 3 usages
    private List<TextView> textViews = new ArrayList<>(); 3 usages

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        etNumberOfElements = findViewById(R.id.etNumberOfElements);
        btnAddElements = findViewById(R.id.btnAddElements);
        etSearchText = findViewById(R.id.etSearchText);
        btnSearch = findViewById(R.id.btnSearch);
        containerLayout = findViewById(R.id.containerLayout);

        btnAddElements.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                addElements();
            }
        });

        btnSearch.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                searchElements();
            }
        });
    }
}

```

```

private void addElements() { 1 usage
    String numStr = etNumberOfElements.getText().toString();
    if (!numStr.isEmpty()) {
        int numElements = Integer.parseInt(numStr);
        containerLayout.removeAllViews();
        textViews.clear();

        for (int i = 0; i < numElements; i++) {
            TextView textView = new TextView(context: this);
            textView.setText("Elemento " + (i + 1));
            textView.setTextSize(18);
            textView.setPadding(left: 16, top: 16, right: 16, bottom: 16);
            containerLayout.addView(textView);
            textViews.add(textView);
        }
    } else {
        Toast.makeText(context: this, text: "Por favor, ingrese un número", Toast.LENGTH_SHORT).show
    }
}

```

```
private void searchElements() { 1 usage
    String searchText = etSearchText.getText().toString().toLowerCase();
    boolean found = false;

    for (TextView tv : textViews) {
        if (tv.getText().toString().toLowerCase().contains(searchText)) {
            tv.setTextColor(Color.WHITE);
            tv.setBackgroundColor(Color.BLACK);
            found = true;
        } else {
            tv.setTextColor(Color.BLACK);
            tv.setBackgroundColor(Color.TRANSPARENT);
        }
    }

    if (!found) {
        Toast.makeText(context: this, text: "No se encontraron elementos", Toast.LENGTH_SHORT).show()
    }
}
```