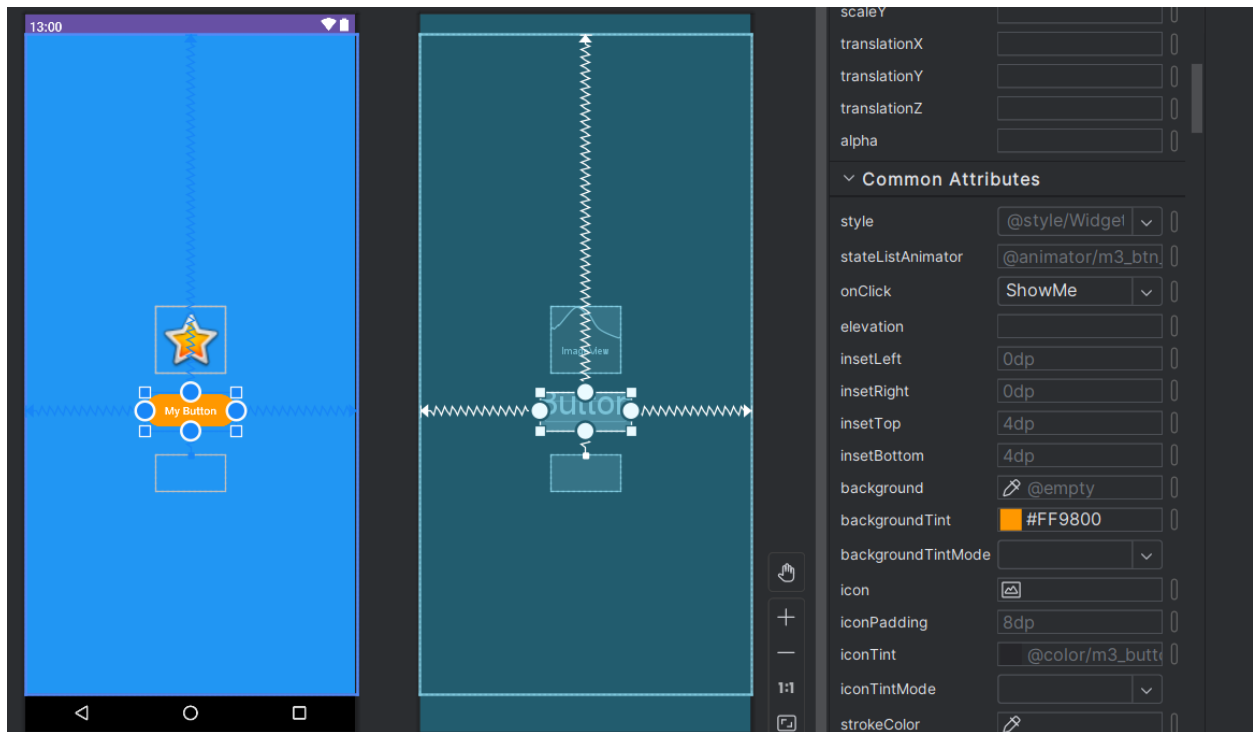
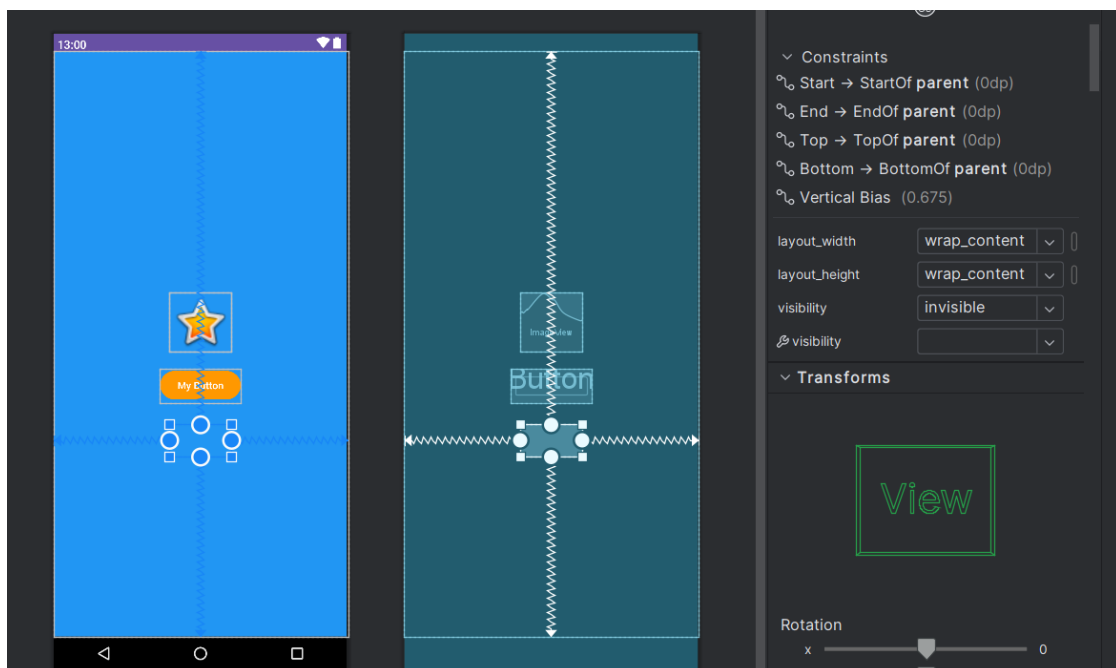


## I. Nút button có thể nhấn và đổi text (Dùng property trong design)

- Đầu tiên hãy cấu hình nút button trong property tìm tới onClick và viết tên hàm là ShowMe



- Tiếp theo là để text bên dưới là chế độ invisible trong visibility



- Trong code MainActivity.java sẽ xử lý như sau:

```

12
13     1 usage
14     private Button mButton;
15     3 usages
16     private TextView mTextView;
17
18     Hungg Duy Bui *
19     @Override
20     protected void onCreate(Bundle savedInstanceState) {
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.activity_main);
23
24         mButton = findViewById(R.id.mButton);
25         mTextView = findViewById(R.id.mTextView);
26
27     }
28
29     1 usage new *
30     public void ShowMe(View v)
31     {
32         mTextView.setVisibility(View.VISIBLE);
33         mTextView.setText(R.string.show_text);
34     }
35 }

```

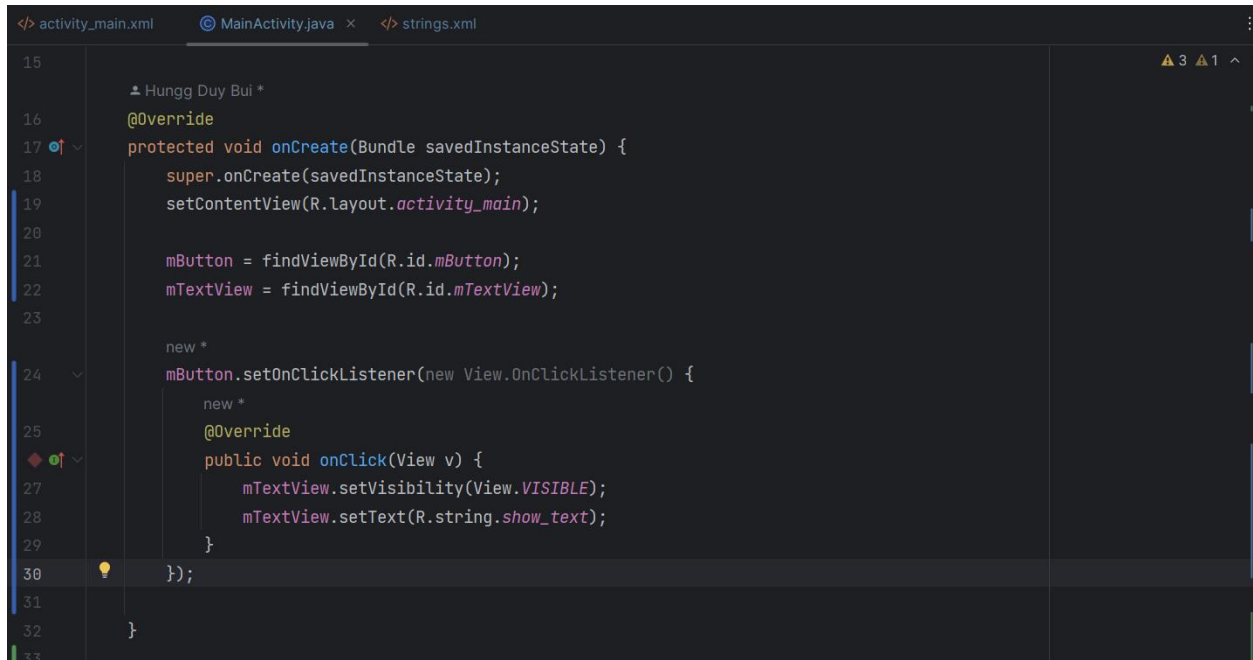
- + Hàm ShowMe được tạo khi chúng ta cấu hình function bên trong property onClick của button. Hàm có chứa tham số là object của View. Class View là lớp cha của tất cả những object như TextView, Button, ImageView,... nên khi nút button được nhấn thì chính nút button đó là object sẽ tham chiếu tới object v

+ `mTextView` chính là `TextView` ta đã ẩn nó đi để khi nhấn nút ta muốn nó hiện ra nội dung bên dưới đã được `setText` thì gọi `setVisibility` cho phép text đó hiển thị cho người xem thấy được

## II. Nút button có thể nhấn và đổi text (Dùng code trong MainActivity.java)

- Cách này chỉ dùng nếu ứng dụng của bạn đơn giản. Ưu tiên dùng property như bên trên

- Chúng ta sẽ sử dụng listener có sẵn của class `View.OnClickListener()` để có thể tạo function `onClick` giống như ở bên design:

A screenshot of an IDE window showing the MainActivity.java file. The code is written in Java and implements the onCreate method. It sets the content view to R.layout.activity\_main, finds the mButton and mTextView views, and sets an OnClickListener for mButton. The listener's onClick method sets the visibility of mTextView to View.VISIBLE and sets its text to R.string.show\_text. The IDE interface includes a toolbar at the top with icons for running, debugging, and other actions. The code is displayed in a dark-themed editor with line numbers on the left and a gutter on the right. The file name MainActivity.java is visible in the tab bar at the top.

```
15
16  Hungg Duy Bui *
17  @Override
18  protected void onCreate(Bundle savedInstanceState) {
19      super.onCreate(savedInstanceState);
20      setContentView(R.layout.activity_main);
21
22      mButton = findViewById(R.id.mButton);
23      mTextView = findViewById(R.id.mTextView);
24
25      new *
26      mButton.setOnClickListener(new View.OnClickListener() {
27          new *
28          @Override
29          public void onClick(View v) {
30              mTextView.setVisibility(View.VISIBLE);
31              mTextView.setText(R.string.show_text);
32          }
33      });
34  }
```