|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Android Studio Minggu Ke-3 SMST5 (Video 5-10 Android Studio Untuk Siswa SMK) | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Membuat projek baru dengan nama Counter, menghapus bluescreen yang ada di activity\_main.xml karena tidak diperlukan, dan menampilkan header aplikasinya atau Show System UI.

2. Menambahkan komponen berupa 2 button dan 1 text, kedua button diletakkan diatas dan bawah dengan setting layout yaitu constraintnya. Button pertama constraintnya dicantolkan ke kiri, atas, dan kanan sedangkan button kedua constraintnya dicantolkan ke kiri, bawah, dan kanan. Cantolan kiri disebut constraint start, kanan end, atas top, dan bawah bottom.

3. Di dalam constraint terdapat beberapa pilihan untuk mengatur posisi yaitu wrap content, fixed, dan match constraint. Untuk match constraint berarti posisi component seukuran dengan lebar mobile yang dipakai.

4. Selain mengatur posisi dan jarak dari sebuah komponen dalam tampilan mobile, komponen juga dapat diubah dari segi warnanya yaitu dengan mengatur background dan backgroundTint yang ada di attributes bagian all attributes. Apabila background tidak berubah dan warnanya tetap padahal sudah diganti, maka di dalam code dari komponen itu bisa ditambahkan app:backgroundTint="@null" agar tema warna background dari komponen itu kembali ke background dasar atau masih kosong belum disetting.

5. Di dalam sebuah komponen terdapat text, baik text saja maupun text yang ada di dalam button. Text ini dapat disetting warnanya, ukurannya, font familynya, dsb dengan mengatur settingan text yang ada di all attributes.

6. Mengatur button Counter Up agar ketika diklik maka angka yang ditengah akan bertambah 1 setiap diklik, sebaliknya button Counter Down akan mengurangi 1 angka yang ditengah setiap kali diklik. Caranya dengan memberikan id disetiap komponen yang terlibat yaitu button counter up diberi id “btnUp”, button counter down diberi id “btnDown”, dan text view angka ditengah diberi id “tvHasil”.

Jika sudah diberi id, java akan dengan mudah mengenali komponen mana yang dimaksud. Pada bagian button “btnUp” dan “btnDown” ditambahkan onclick sesuai nama id nya untuk mengatur ketika button diklik maka apa yang harus terjadi selanjutnya. Pada code button “btnUp” akan muncul android:Onclick”btnUp” dan begitu juga pada code button “btnDown”. Di setiap onclick yang ada dalam code komponen pasti akan ada lampu merah di sebelah kiri code yang menandakan onclick ini belum diatur untuk apa, untuk itu pilih create new pada lampu merah tadi agar bisa masuk ke halaman java dan mengatur onclicknya.

Di dalam java, pertama tambahkan sebuah variable dengan type integer bernama count dengan isi 0 seperti ini int count = 0; lalu tambahkan juga TextView tvHasil; Setelah itu buatkan sebuah function dengan mengaktifkannya terlebih dahulu di dalam function onCreate dengan cara menambahkan load(); di code paling bawah. Jika function load sudah aktif, maka buatkan function loadnya dibawah function onCreate. Isi function load dengan code tvHasil = findViewById(R.id.tvHasil); yang berarti variable tvHasil yang dideklarasikan tadi diisi dengan komponen yang memiliki id tvHasil yaitu angka yang textView angka yang ditengah.

Pada function btnUp diisi dengan count++ yang berarti disetiap button “btnUp” diklik maka count akan bertambah 1, sedangkan function btnDown diisi dengan count-- yang berarti disetiap button “btnDown” diklik maka count akan berkurang 1. Dibawah count++ maupun count-- ditambahkan tvHasil.setText(count+""); yang berarti isi tvHasil dengan count saat ini atau isi yang terbaru misalnya sudah menjadi angka 3 karena 3x mengklik button “btnUp”, ada tambahan +”” yang berarti count type integer diubah menjadi count type string agar bisa ditampilkan di TextViewnya.

Dengan begitu, setiap kali diklik button counter up, textview yang ada ditengah akan terisi dengan count terbaru yaitu bertambah 1, dari 0 menjadi 1, begitu juga setiap kali diklik button counter down akan sebaliknya.

7. Menambahkan System.out.println(x) hanya dengan mengetikkan sout lalu enter pada page java.

8. LinearLayout berfungsi untuk mengatur posisi komponen agar sejajar, dalam linearlayout ada 2 pilihan yaitu horizontal yang berarti sejajar menurut garis x atau vertical yang berarti sejajar menurut garis y. Cara menggunakan linearlayout adalah dengan membuka tampilan code dari as lalu mengubah android.support.constraint.ConstraintLayout dengan LinearLayout, lalu menambahkan android:orientation=”” yang di dalam petik 2 itu bisa diisi dengan vertical atau horizontal.

9. Untuk mengatur beberapa komponen agar settingan attributesnya sama misalkan ukuran marginnya sama caranya dengan menekan tombol shift sambil mengklik dan memilih komponen mana saja yang akan disetting pada component tree. Jika beberapa komponen yang dipilih sudah terlihat seperti diblok maka settingan yang terjadi di dalam attributes akan berjalan di beberapa komponen yang dipilih tadi.

10. Biasanya dalam android studio untuk setting margin menggunakan satuan dp sedangkan setting textsize menggunakan sp.

11. Terdapat layout gravity yang dapat digunakan untuk memposisikan suatu komponen. Layout gravity memiliki beberapa macam, salah satunya center yang dapat dicentang agar menjadi true sehingga komponen posisinya berada di tengah.

12. Semua setting attributes yang telah digunakan dalam suatu komponen dapat dilihat di dalam Declared Attributes. Dari setting warna, ukuran, layout, dsb yang ada pada sebuah komponen ada di dalam declared attributes, jadi jika ingin merubah setting sebelumnya bisa melalui declared attributes tanpa harus scroll mencari attributes yang dituju.

13. Untuk membuat projek kalkulator penjumlahan, yang pertama yaitu harus memberikan id disetiap komponennya dulu agar mudah dipanggil ketika digunakan dalam java. Misalnya untuk angka yang ditambilkan diberi id tvHasil, untuk edit text ke satu diberi id etBil\_1 dan edit text ke dua diberi id etBil\_2 dan button penjumlahan diberi id btnJumlah.

Setelah memberikan id pada komponen tersebut, pada bagian button penjumlahan onClick nya juga diberi nama sesuai id nya yaitu btnJumlah. Ketika sudah diberi nama onClicknya, pada tampilan code buat Create in MainActivity agar onclick dapat disetting (ketika button diklik apa yang akan terjadi).

Di dalam page javanya, tambahkan TextView tvHasil; dan EditText etBil\_1, etBil\_2; sebagai deklarasi sebuah variable beserta type nya. Lalu di dalam function onCreate ditambahkan load() untuk mengaktifkan function load.

Function load diisi dengan

tvHasil = findViewById(R.id.*tvHasil*);  
etBil\_1 = findViewById(R.id.*etBil\_1*);  
etBil\_2 = findViewById(R.id.*etBil\_2*);

yang berarti variable tvHasil dengan type TextView sesuai yang dideklarasikan tadi merupakan komponen yang memiliki id tvHasil dengan type TextView, begitu juga dengan etBil\_1 dan etBil\_2

Dengan begitu, TextView tvHasil bisa diisi dengan hasil setelah penjumlahan, EditText etBil\_1 dan etBil\_2 bisa digunakan dalam java untuk dioperasi hitung. Model penjumlahan codenya sebagai berikut.

public void btnJumlah(View view) {  
  
 if (etBil\_1.getText().toString().equals("") || etBil\_2.getText().toString().equals("")){  
 Toast.*makeText*(this, "Ada bilangan yang kosong", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 double bil\_1 = Double.*parseDouble*(etBil\_1.getText().toString());  
 double bil\_2 = Double.*parseDouble*(etBil\_2.getText().toString());  
  
 double hasil = bil\_1 + bil\_2;  
  
 tvHasil.setText(hasil+"");  
 }  
}

yang berarti jika etBil\_1 kosong (ditandai dengan adanya equals(“”)) dan(ditandai dengan adanya ||) etBil\_2 juga begitu maka dia akan menampilkan sebuah pesan singkat yang bertuliskan “Ada bilangan yang kosong”. Namun jika etBil\_1 dan etBil\_2 tidak kosong maka dia akan menjalankan variable bil\_1 dengan type double diisi dengan etBil\_1 yang awalnya merupakan string diubah menjadi double agar dapat dijumlahkan nantinya begitu juga dengan variable bil\_2 yang diisi dengan etBil\_2.

Jika variable bil\_1 dan bil\_2 sudah mendapatkan isi dari etBil\_1 dan etBil\_2 dalam bentuk double, maka dalam variable hasil dapat diisi dengan operasi penjumlahan (+) antara bil\_1 dan bil\_2. Terakhir, hasil dari penjumlahan ini akan disimpan dalam variable hasil lalu terdapat tvHasil.setText(hasil+””); yang berarti tvHasil akan diisi dengan variable hasil yang sudah dirubah ke dalam bentuk string (ditandai dengan adanya +””).

15. Jika sebelumnya adalah operasi penjumlahan, maka untuk operasi pengurangan, perkalian, dan pembagian menggunakan cara yang sama juga. Perbedaannya terletak pada tanda yang digunakan dalam operasi hitungnya, jika penjumlahan menggunakan tanda plus(+) maka pengurangan minus(-) perkalian(\*) dan pembagian(/).

14. Untuk menggunakan pesan dalam tampilan mobile, pada page java bisa diketikkan Toast lalu tekan tab maka akan otomatis keluar Toast.*makeText*(this, "text", Toast.*LENGTH\_SHORT*).show();

15. Menambahkan image png ke dalam folder drawable yang ada di folder res. Memasukkan gambar ke dalam projek dengan cara yang sama seperti menambahkan button yaitu menyeret komponen image ke dalam mobile dan memilih image mana yang akan digunakan.

16. Mengganti warna header dari aplikasi mobile nya dengan warna lain, caranya dengan mengcopy paste code warna (missal #fffff) ke dalam file colors.xml yang ada di folder values dari folder res. Untuk mendapatkan code sebuah warna bisa dilihat dari web material.io

17. Menggunakan blueprint untuk mempermudah melihat posisi komponen dan memasangkan constraint tanpa ada komponen lain yang tertumpuk.

18. Menambahkan spinner yang berfungsi sebagai komponen yang memberikan beberapa pilihan dengan model tanda panah yang mengarah ke bawah sebagai penanda bahwa komponen ini bisa dipilih isinya.

19. Mengisi beberapa data atau array ke dalam komponen spinner, caranya yaitu dengan membuka file strings.xml yang ada di folder values dalam res. Di dalam file strings.xml diisi dengan <String-array name=””></String-array> yang berfungsi untuk menyimpan data array dengan type string, untuk nama bisa diisi dengan misalnya “pilihan” lalu di dalam String-array ini diisi dengan <item></item> dimana item ini adalah data-data yang akan dimasukkan ke dalam spinner, misalnya <item>Celcius To Fahrenheit</item> dan seterusnya sebanyak data yang diperlukan.

Jika String-array sudah terisi dengan semua data, selanjutnya dalam attributes dari komponen spinner pada bagian entries diisi dengan nama dari String-arraynya yaitu pilihan seperti ini @array/pilihan, dengan begitu spinner akan terisi dengan data yang diambil dari array pilihan.

20. Jika sebelumnya mengisi data dan menampilkan data array ke dalam spinner menggunakan entries dan String-array di dalam file strings.xml. Ada juga cara lain untuk memasukkan data array ke dalam spinner untuk ditampilkan, caranya yaitu dengan membuatkan sebuah function baru dan sekaligus mengaktifkannya di dalam function onCreate. Function baru tersebut diberi nama isiSpinner seperti ini

public void isiSpinner() {  
 String[] isi = {"Celcius To Reamur","Celcius To Fahrenheit","Celcius To Kelvin"};  
 ArrayAdapter<String> adapter = new ArrayAdapter<String>(this, android.R.layout.*simple\_spinner\_item*,isi);  
 spinner.setAdapter(adapter);  
}

Jadi, dibuatkan sebuah array bernama isi yang memiliki 3 data yaitu “Celcius To Reamur”, “Celcius To Fahrenheit”, dan “Celcius To Kelvin”. Lalu data array itu dimasukkan ke dalam spinner dengan cara dibuatkan sebuah ArrayAdapter type string dengan nama adapter yang isinya buat ArrayAdapter yang baru di dalam class ini (this) dan isinya diambil dari array bernama isi. Terakhir ada spinner.setAdapter(adapter); yang berarti komponen spinner diisi dengan ArrayAdapter bernama adapter.

21. Untuk membuat button Konversi ketika diklik akan otomatis mengambil data sesuai data spinner yang dipilih caranya dengan mengaktifkan onClick seperti biasanya dengan nama btnKonversi, lalu di dalam page java dibuatkan variable spinner dengan type data Spinner sedangkan di dalam function btnKonversi diisi seperti ini

public void btnKonversi(View view) {  
 String pilihan = spinner.getSelectedItem().toString();  
  
 System.*out*.println(pilihan);  
}

yang berarti ada variable pilihan dengan type data String yang berisi ambil item yang dipilih dari spinnernya atau ambil data yang dipilih dari spinnernya (getSelectedItem) lalu diubah menjadi string. Jika sudah mendapat data, data bisa ditampilkan ke dalam sout (System.out.println (pilihan) karena variable yang menyimpan data tadi bernama pilihan) yang bisa dilihat di logcat android studio.

**Saya Belum Mengerti**

1.