|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Android Studio Minggu-4 | 80 |

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

1. Bisa melengkapi code dari minggu yang sebelumnya (konversi suhu) sesuai dengan yang ada di video (celcius to reamur) dan melanjutkan beberapanya sendiri (seperti celcius to fahrenheit, dan yang lainnya). Serta dapat ditampilkan dan digunakan pada saat di run (dijalankan).

2. Command, untuk mematikan suatu code (agar tidak dapat terbaca dan dipergunakan kembali), dengan cara memberikan /\* …. \*/ atau //…. (titik-titik tersebut untuk diisikan dari code yang akan dimatikan nantinya).

3. Bisa membuat function sesuai dengan yang ada di video (public void).

4. Entries, digunakan untuk mengisikan isi spinner yang telah diatur sebelumnya di dalam code strings.xml dan kemudian ditampilkan ke dalam isian spinnernya (pada saat ditampilkan (run), maka isi dapat dipilih sesuai dengan apa yang sudah diisikan (diatur) sebelumnya di dalam code strings.xml).

5. System.out.println, untuk menampilkan pesan atau sesuatu yang telah dibuat dan diatur sebelumnya di code, yang kemudian setelah sesuatu di klik atau diberi apapun yang sesuai dengan arahan sebelumnya, akan memunculkan pesan atau sesuatu tersebut, dan munculnya ada di Logcat. Bisa membuatnya dengan cepat dengan mengetikkan sout lalu mengklik enter (otomatis keluarnya sama seperti System.out.println).

6. Logcat, untuk memunculkan segala data dari sebuah project yang sedang dijalankan (di run), bilamana ada error juga bisa dilihat dan ditemukan solusinya lewat sini. (Pada saat pembuatan Activity Lifecycle lebih banyak dipergunakan). Berada dibagian bawah sendiri berjejeran dengan beragam hal salah satunya seperti TODO, Problems, dan lainnya. Pada Logcat bisa juga di search ingin mencari data yang di load yang mana (seperti System.out.println).

7. Bisa mempergunakan id yang sebeumnya telah diatur pada tiap komponen sesuai dengan yang ada di video.

8. Load(), bila setelah membuat code, maka hasil yang akan ditampilkannya di lewatkan function load() dan agar dapat digunakan, harus dimasukkan ke dalam function onCreate bawaan dari MainActivity.java dari android studionya (di bawahnya, bukan langsung di dalamnya, namun masih satu tutup kurung dengan onCreate).

9. FindViewById, untuk menempatkan suatu komponen yang sudah memiliki id masing-masing (dipanggilnya menggunakan ini). (Penggunaannya seperti findViewById(R.id.etBarang); (R untuk res (resource), id untuk menunjukkan bahwa akan memanggil id, dan etBarang merupakan id dari suatu komponen yang telah diubah sebelumnya (id dari Plain Text))).

10. Double, untuk mengubah suatu variabel dari yang awalnya string, lalu dapat berubah menjadi double (penggunaannya harus diikuti dengan parseDouble dan bila sudah menggunakan double, maka jangan dicampurkan dengan integer (int)).

11. String, untuk code yang berisikan teks atau kalimat menggunakan ini dan bukannya integer (int) maupun double.

12. Run App (berlogo segitiga warna hijau menghadap ke kanan di bagian atas (di sebelah tampilan sambungan device)), untuk menjalankan/memulai code yang sudah dibuat sebelumnya (memutar dan menunjukkan hasilnya di device yang sudah ditujukan (misal di nox, maka akan langsung muncul di layar nox tersebut (bila success/berhasil) namun bila gagal/error, maka tidak akan muncul di tampilan layar nox dan menampilkan bagian errornya yang mana dan apa)).

13. Stop (berlogo warna merah persegi/kotak di bagian atas (dekat dengan run app)), untuk memberhentikan aplikasi/project yang sedang dijalankan.

14. Bisa membuat code dari perubahan (konversi suhu) sesuai dengan rumus yang tersedia (dan juga sesuai dengan yang ada di video).

15. Bisa mengubah dari string ke integer (int) menggunakan count+” ”. Bisa juga kata dari count tersebut diubah selain count.

16. Set Text, untuk mengisikan teks atau sesuatu yang telah diisikan ke dalamnya dan menampilkannya.

17. Bilamana membuat suatu function untuk menjalankan sesuatu, tidak lupa untuk mendeklarasikannya terlebih dahulu, termasuk variabelnya.

18. Menggunakan if untuk memberikan batasan bilamana tempat yang seharusnya diberikan angka untuk menghitung konversi suhu masih kosong, maka akan memunculkan “Nilai Tidak Boleh Kosong”. Dan menggunakan else yang diberikan if lagi, bila tempatnya sudah berisikan angka dan pilihan dari isi spinnernya sudah dpilih (misal memilih celcius to reamur), maka otomatis akan langsung mengonversikan nilai dari yang sudah dipilih dari isi pilihan tersebut (sebelumnya diberikan code di dalam else yang berisikan if lagi yang sudah diatur per konversi pilihan suhunya (sendiri-sendiri, seperti celcius to reamur sudah disendirikan (functionnya sudah masing-masing), jadi bila memilih pilihan ini akan otomatis menjalankan code yang sudah diatur di dalam pilihan celcius to reamur tersebut)). Bisa juga menghasilkan kalimat seperti “Belum Dibuat” pada pilihan celcius to fahrenheit sebelumnya (namun sekarang sudah saya tambahkan dan dapat dipergunakan untuk mengonversi suhu juga). Pilihan yang dipilih juga harus benar-benar sesuai dengan apa yang ingin diubah atau dikonversikan, karena pada tiap pilihan sudah berbeda perhitungan dan hasilnya nanti.

19. Toast, (keluarnya memunculkan tulisan yang sudah diatur sebelumnya dan terletak dibagian bawah dari layarnya). Bisa juga menampilkannya menggunakan showToast pada saat di Alert Dialog Button.

20. Bisa mengganti atau mengubah icon launcher sesuai dengan yang ada di video.

21. Mengganti icon launcher dengan cara menambahkan gambar yang ingin dijadikan dari suatu tampilan icon tersebut ke dalam mipmap dan diletakkan ke dalam folder ic\_launcher (klik paste dari copy an gambar yang dpilih akan jadikan icon), lalu pilih ..\app\src\main\res\mipmap-xhdpi, kemudian pilih ok. Selanjutnya, pada pemberian nama dari gambar yang akan dimasukkan tadi, diharuskan mrnggunakan huruf kecil dan tidak boleh ada spasi untuk memberikan namanya. Setelah itu klik ok kembali bila sudah selesai. Lalu menuju ke AndroidManifest.xml untuk mengganti iconnya. Dengan cara mengubah pada bagian android:icon=”….” (titik-titik tersebut berisikan nama folder dan file dari gambar yang akan dijadikan icon, seperti @mipmap/termometer).

22. Bila ingin icon yang telah diubah menjadi bulat, maka harus mengganti android:roundIcon=”….” (titik-titik tersebut berisikan nama folder dan file dari gambar yang sama seperti yang sudah diubah pada android:icon=”….” , seperti @mipmap/termometer). Di dalam AndroidManifest.xml (dekat dengan android:icon=”….”).

23. Activity Lifecycle, merupakan suatu runtutan dari jalannya suatu program, dari mulai awal startnya, yang lalu diberhentikan (pause) dan dikembalikan (resume), sampai dengan dihilangkan (destroy).

24. Bisa memahami dari runtutan yang terdapat pada Activity Lifecycle.

25. Bisa membuat dan menjalankan (pada saat di System.out.println) Acvtivity Lifecycle sesuai dengan yang ada di video.

26. Menambahkan activity lifecycle tersebut dengan klik kanan, lalu pilih Generate, kemudian pilih Override Methods, setelah itu bisa dipilih mana yang akan digunakan atau diperlukan (bisa di search juga, misalkan mencari onStart dan mengetikkannya ke dalam kotakan search, setelah itu akan keluar onStart yang dicari tadi). Setelahnya bisa langsung di klik ok, maka otomatis akan membuat @Override beserta functionnya.

27. OnCreate, pertama kali pada saat aplikasinya akan dijalankan, maka terbuatnya dari sini terlebih dahulu. (Akan keluar bilamana suatu project atau aplikasi dijalankan (di run) pada saat pertama kali).

28. Onstart, memulai suatu aplikasi setelah terbuat atau terbentuk oleh onCreate. (Akan keluar bilamana suatu project atau aplikasi dijalankan (di run) pada saat pertama kali). Akan juga langsung terrestart dan masuk ulang ke sini pada saat bebarengan dengan resume setelah di pause dan stop.

29. OnPostResume, mengulangi suatu project yang sebelumnya terhenti sejenak. (Akan keluar bilamana suatu project atau aplikasi dijalankan kembali pada saat setelah dikeluarkan dan memulai ulang aplikasinya (keluar juga bebarengan dengan onCreate dan onStart)).

30. OnPause, menghentikan suatu project atau aplikasi yang nantinya dapat langsung dimuat kembali. (Akan keluar, bilamana aplikasi androidnya di tab keluar dari emulatornya (bisa klik home maupun tab berisikan aplikasi dan dapat dibersihkan (clear all))).

31. OnStop, menghentikan dan memberhentikan suatu aplikasi (akan keluar, bilamana aplikasi androidnya di tab keluar dari emulatornya (bisa klik home maupun tab yang berisikan aplikasi dan dapat dibersihkan (clear all))).

32. OnDestroy, menghancurkan dan menutup suatu aplikasi dan bilamana aplikasinya dibuka kembali akan mengarah kepada onCreate yang membuat aplikasinya kembali (serta memuat ulang onStart dan onResume). aplikasi (Akan keluar, bilamana aplikasi androidnya di tab keluar dari emulatornya (bisa klik home maupun tab yang berisikan aplikasi dan dapat dibersihkan (clear all) dan kemudian di clear all (untuk menghilangkan atau menghapuskan aplikasi tersebut))).

33. Linear Layout, untuk menjadikan layer dari tampilan otomatis bertingkat (bersap-sap) dan otomatis menjadi rapi (seperti pada saat mengatur suatu komponen lewat handle yang dikaitkan dengan bagian bawah dari komponen (bila komponen tersebut berada di bawahnya) yang ada diatasnya dan mengatur marginnya sama rata). Mengubahnya dari code activity\_main.xml dibagian codenya pada bagian atas sendiri. Yang bertuliskan android.support.constraint.ConstraintLayout diganti menjadi Linear Layout. Maka penutup bagian bawahnya langsung otomatis ikut berubah juga mengikuti bagian atasnya. Setelah itu otomatis tampilan layernya akan tersusun rapi dan seperti sudah bergaris-garis (sudah tertata).

34. Menambahkan android:orientation, untuk menjadikan garis yang menjadikan tersusun rapinya layer dengan ini (pilihannya terdadpat vertical dan horizontal, pada video menggunakan vertical).

35. Margin, untuk memberikan jarak pada komponen yang ada di dalam layer (baik pada layout\_margin maupun pada jarak handle).

36. Bisa membuat margin sesuai dengan yang ada pada video (tidak lupa untuk akhiran atau satuannya diberikan dp).

37. Bisa membuat id untuk tiap komponen sesuai dengan yang ada pada video (dan sudah bisa mempergunakannya pada saat pemanggilan komponen pada code).

38. Bisa mengganti teks yang ada pada komponen sesuai dengan yang ada di video (misal seperti button yang diubah namanya menjadi TOAST, bisa mengubahnya baik dari dalam design maupun code).

39. On Click, akan menampilkan suatu kejadian dengan di klik terlebih dahulu (komponennya di klik terlebih dahulu, lalu akan menampilkan apa). Pada awal setelah menambahkannya, maka akan terjadi error, membenarkannya dengan mengklik logo seperti bohlam lampu namun berwarna merah di samping kiri dari baris yang error (onClick), lalu kemudian klik Create ‘btnBarang(View)’ in ‘MainActivity’ (pada error onClick di button btnBarang, bila error pada button lain atau attribute dari komponen yang lain, maka namanya akan mengikuti dari id komponen tersebut (bukan lagi btnBarang)). Setelah itu, maka akan terbuat void atau method (public void btnBarang (View view) {}) di main activity.

40. Bisa membuat dan menampilkannya (menjalankannya) message dialog dengan beragam macam (toast, alert dialog, dan alert dialog button) sesuai dengan yang ada pada video.

41. Memahami fungsi atau kegunaan dari message dialog yang telah dibuat sesuai dengan yang ada di video.

42. Alert Dialog, akan memunculkan pesan seperti pada saat ada peringatan (memiliki title dan message di dalamnya). Letaknya ada di tengah-tengah dari layar.

43. Bisa membuat showAlert sesuai dengan yang ada di video. (Membuat AlertDialog.Builder pada saat pembuatannya). Digunakan untuk memunculkan alert yang telah dibuat sebelumnya (termasuk ke dalam suatu variabel).

44. SetTitle, untuk memberikan nama pada judul dari showAlert (pada bagian atasnya seperti “PERHATIAN !”).

45. SetMessage, untuk memberikan nama pada pesan di bawahnya title dari shiwAlert (pada bagian bawahnya setTitle, dibuat dinamis agar pesannya dapat diganti-ganti setiap saat dan sesuai dengan yang diinginkan).

46. Bisa membuat Alert Dialog Button sesuai dengan yang ada di video. Digunakan untuk menampilkan pesan hampir mirip seperti Alert Dialog, namun perbedaannya ada pada tambahan pesan peringatan, terdapat tambahan penghapusan data yang biasanya muncul dan terdapat pilihan ya atau tidak. Bila di klik salah satunya maka akan menghasilkan sesuatu yang telah dibuat sebelumnya (yang dibuat melalui setPositivieButton dan setNegativeButton). Bila salah satunya sudah dipilih, maka akan kembali pada pilihan button (awal yang terdapat tiga (3) button).

47. SetPositiveButton, untuk pilihan positif dari alert dialog button (positif yang dimaksud yakni seperti jawabannya Ya yang merajuk ke arah setuju (positif)). (Dan setelah di klik atau di setujui maka akan menjalankan tulisan “Data Sudah dihapus !” (seperti pada saat di video)).

48. SetNegativeButton, untuk pilihan negatif dari alert dialog button (negatif yang dimaksud yakni seperti jawabannya Tidak yang merajuk ke arah tidak setuju (negatif)). (Dan setelah di klik atau di setujui (untuk memilih tidak) maka akan menjalankan tulisan “Data tidak dihapus” (seperti pada saat di video)).

49. Bisa membuat project baru (beserta tiap isinya) sesuai dengan yang ada pada video.

50. Button, suatu komponen berbentuk kotakan (lebih ke arah persegi panjang).

51. PlainText, suatu komponen kosong yang bisa diisikan kata, kalimat, angka, dan yang lainnya (tidak seperti Number (Decimal) yang hanya dapat digunakan untuk mengisikan angka di dalamnya).

52. Wrap Content, ukuran dari suatu komponen yang mengikuti isi yang ada di dalamnya (misalkan bertambah panjang tulisannya, maka kotakan komponennya akan ikut melebar sesuai dengan isinya). Menggunakannya dengan cara tidak perlu merubah garis apapun yang ada di tengah pengaturan handle yang terdapat di dalam Constraint Widget.

53. Fixed, ukuran dari suatu komponen yang tidak mengikuti isi yang ada di dalamnya (misalkan bertambah panjang tulisannya, maka kotakan komponennya tidak akan ikut melebar (akan menetap) berapapun isi yang ada di dalamnya). Menggunakannya dengan cara mengklik sekali garis yang ada di tengah pengaturan handle yang terdapat di dalam Constraint Widget.

54. Match Constraint, ukuran dari suatu komponen yang tidak mengikuti isi yang ada di dalamnya, namun kotakannya mengikuti lebar dari tampilan layarnya, berapapun isi yang di dalamnya tidak akan memengaruhi lebar kotakan. Menggunakannya dengan cara mengklik dua kali garis yang ada di tengah pengaturan handle yang terdapat di dalam Constraint Widget.

55. Bisa membuat activity baru sesuai dengan yang ada pada video. Dengan cara klik kanan pada folder app (atas sendiri di bagian project), lalu pilih new dan setelah itu memilih activity. Pada activity terdapat beragam pilihan (seperti pada saat memilih activity di awal mula membuat project baru, misal seperti pada video yang memilih empty activity). Kemudian mengaturnya terlebih dahulu d awal seperti apa nama projectnya, dan lain-lain (seperti pada saat awal membuat project baru). Di klik centang juga pada bagian Generate Layout File (tepat di bawah activity name). Setelah terbuat, maka akan memiliki file activity\_main.xlm dan MainActivity.java lebih dari satu (sesuai dengan berapa jumlah activity yang akan dibuat).

56. Bisa mengganti label nama pada action bar (sesuai dengan yang ada pada video).

57. Intent Activity untuk memanggil atau menautkan satu activity ke activity lainnya (seperti MainActivity.java yang ditautkan pada activity Barang.java yang nantinya juga bisa dikembalikan kembali pada MainActivity.java lagi melalui simbol panah ke kiri (untuk kembali)).

58. Bisa membuat Intent Activity sesuai dengan yang ada pada video serta sudah menyelesaikan dua (2) button lainnya (penjualan dan pembelian) yang diberi Intent Activity (sama seperti button barang).

59. Android:parentActivityName, digunakan untuk memindahkan suatu activity ke activity lainnya yang dituliskan nama activitynya di dalamnya (seperti activity Barang.java yang bilamana tanda panah dibagian pojok atas sebelah kiri (bawaan otomatis dari android:parentActivityName) di klik, maka akan kembali pada activity awalnya (MainActivity.java)). Jangan lupa untuk memperhatikan tanda tutupnya juga.

60. Android:label, digunakan untuk mengganti nama dari label bagian atas di activity (label action bar). (Seperti yang awalnya bernamakan IntentActicity dan kemudian diubah nama menjadi Barang).

61. Hint (pada attributes), untuk membuat tulisan seperti dibelakang layar dari kotakannya (jadi pada saat kotakannya diisi, maka tulisan tersebut akan hilang dan tergantikan oleh tulisan yang diisikan di kotakan yang telah diberi hint tadi. Bila tidak ada tulisannya, maka yang muncul adalah tulisan yang sudah dibuat di hint tadi pada kotakannya.

62. Bisa membuat suatu kata atau kalimat bila dituliskan pada plain teks yang jadi satu layer dengan button yang telah diberikan Intent Activity sebelumnya, maka kata atau kalimat tersebut akan ikut ditampilkan pada layer yang buttonnya telah di klik (misal menuliskan kata barang dan mengklik button barang, maka kata barang tersebut akan tampil pada layar barang (Barang.java)). Membuatnya dengan cara membuat suatu function (load) dan membuat beberapa variabel yang diajdikan satu ke dalam function suatu button tersebut (seperti btnBarang). Serta pada activity\_barang.xml nya ditambahkan komponen TextView yang sudah diatur sebelumnya agar kata atau kalimat yang diisikan pada plain text di activity sebelumnya (MainActivity.java) bisa muncul pada TextView di acticity barang (Barang.java), dengan memasukkan dan menambahkan function load() dan ambilData() ke dalam Barang.java. Dimasukkan pula pada function onCreate bawaan dari Andorid Studionya (baik dari MainActivity.java maupun Barang.java).

63. Get Text, untuk mendapatkan hasil atau sesuatu yang telah diisikan ke dalamnya dan menampilkannya.

64. Intent.putExtra, digunakan untuk mengirimkan pesan pada Barang.java dari MainActivity.java yang nantinya diterima disana dan juga ditampilkan.

65. GetStringExtra, digunakan untuk menerima pesan dari intent.putExtra dari MainActivity.java dan menampilkannya pada text view yang telah disediakan.

66. ScrollView, untuk menjadikan layar (layer) bisa di scroll ke atas dan ke bawah. Di dalam dari komponen ScrollView tersebut sudah otomatis berisikan LinearLayout di dalamnya (jadi komponen yang dimasukkan akan teratur dalam LinearLayout dari ScrollView tersebut). Ukuran layar dari ScrollView sendiri dapat diatur ingin menjadi berapa. Jadi bila ada beberapa isi di dalamnya dan ukuran layer dari ScrollViewnya tidak memuat semua isinya, maka bisa diatur sesuai dengan keinginan, mau yang dimunculkan awalnya ada berapa (misal tiga (3) seperti pada video) dan isi selanjutnya (yang lainnya) dapat dilihat dan diisikan (bilamana komponennya suatu teks) bila di scroll ke bawah. Dan bila ingin melihat komponen yang sudah hilang atau tertutup karena sudah di scroll ke bawah, maka bisa di scroll ke atas kembali untuk menampilkan dan melihatnya lagi.

67. Bisa membuat ScrollView sesuai dengan yang ada pada video.

68. Bisa memahami fungsi dan kegunaan dari ScrollView itu sendiri.

69. Layout\_width, digunakan untuk mengubah lebar suatu komponen pada Android Studio (bagian design dari layernya, misal digunakan seperti pada ukuran lebar dari komponen ScrollView). Belakangnya ditambahkan dp untuk satuannya.

70. Layout\_height, digunakan untuk mengubah tinggi suatu komponen pada Android Studio (bagian design dari layernya, misal digunakan seperti pada ukuran tinggi dari komponen ScrollView). Belakangnya ditambahkan dp untuk satuannya.

71. Handle, untuk mengatur pegangan yang terdapat 4 (empat) sisi bagian (atas, kanan, bawah, dan kiri). Mengaturnya bisa dengan menarikannya (menautkan) ke sisi yang akan diberikan pegangan atau bisa juga dengan di klik tombol berlogo tanda panah dengan border bulat biru pada Layout (Constraint Widget) di bagian Attributes. Komponen diharuskan diberi handle, karena bila tidak, maka komponen akan menuju ujung pojok kiri atas bila saat di run (dijalankan) atau bisa juga di cek sudah terakitkan atau belum dengan megeceknya di toggle (logo seperti kunci inggris).

72. Untuk menjadikan margin semua komponennya sama, maka mengklik salah satu komponennya (dari atas ke bawah saja biar lebih mudah), kemudian sambil menekan tombol shift, dan mengarahkannya ke bawah atau bisa juga di klik satu-satu per komponennya. Lalu bila mengubah marginnya, maka semua komponennya akan diberikan margin yang sama juga.

73. Bisa mengubah isi teks di dalam suatu komponen yang sudah dipilih dan yang akan digunakan nantinya (seperti button, plain text, dan yang lainnya). Baik dari design maupun code nya.

74. Blueprint, untuk mengetahui tata letak dari komponennya agar bisa pas atau baik. Untuk memunculkan blueprint, dapat dimunculkan pada tombol di sebelah kiri yang bergambar seperti lapisan kertas yang menumpuk (yang di sebelah kanannya ada gambar seperti rotate hp). Dengan cara di klik lalu bisa dipilih, mau yang layer halaman putih saja, blueprint nya saja, maupun tampil keduanya secara bersamaan.

75. Show Display UI, untuk menampilkan tampilan layer pada saat pembuatan dengan design jadi lebih mirip dengan aslinya (sama dengan pada saat di run (dijalankan) di emulatornya). Logonya seperti mata dan letaknya ada dibawah dari pemilihan layer design dan blueprint (yang logonya seperti lapisan kertas yang bertumpuk).

76. Bisa mengatur ukuran margin keseluruhannya secara otomatis menjadi berapa dp (bisa dilihat pada saat pengaturan handle, maka akan otomatis berukuran sesuai dengan yang diatur ini) dengan mengaturnya di bagian tombol yang bertuliskan angka dengan dp di belakanngnya (logonya), terletak di dekat pengaturan logo mata (saat mengatur Show Display UI). Lalu memilihkan angkanya mau berapa ukuran marginnya (misal 24dp seperti pada video).

77. Bisa menampilkan kalender dan mengambil tanggalan yang diinginkan (dari tampilan kalendernya) dan diletakkan tanggal dari kalender tersebut pada sebuah komponen teks sesuai dengan yang ada pada video.

78. DAY\_OF\_MONTH (digunakan pada saat menampilkan kalender bagian memilih atau mengambil tanggal sesuai dengan yang diinginkan), untuk memilih hari pada saat pengaturan date picker (bisa langsung menunjukkan tanggal berapa hari ini).

79. MONTH (digunakan pada saat menampilkan kalender bagian memilih atau mengambil tanggal sesuai dengan yang diinginkan), untuk memilih bulan pada saat pengaturan date picker (bisa langsung menunjukkan bulan berapa hari ini, namun untuk bulan, harus ditambahkan +1, karena kalua tidak, maka bulannya akan menunjukkan bulan sebelumnya dari bulan sekarang, misal sekarang bulan 8 (agustus), maka yang ditampilkan ialah bulan 7 (juli)).

80. YEAR (digunakan pada saat menampilkan kalender bagian memilih atau mengambil tanggal sesuai dengan yang diinginkan), untuk memilih tahun pada saat pengaturan date picker (bisa langsung menunjukkan tahun berapa hari ini).

81. DatePickerDialog, untuk menampilkan suatu kalender (dan nantinya bisa pilih tanggalannya).

82. Pada saat pembuatan DatePickerDialog, menambahkan DatePickerDialog.OnDateSetListener() (pada saat penambahan, terdapat beberapa pilihan dari DatePickerDialognya, kemudian memilih yang memiliki logo warna hijau, yaitu DatePickerDialog.OnDateSetListener()), yang kemudian bila di klik maka akan langsung membuat @Override sendiri dan beberapa hal yang dibutuhkan (otomatis sudah dibuatkan). Pada awalnya akan error (berwarna merah) dan membenarkannya dengan menambahkan koma (,) diantara dua tutup kurung yang berjejeran sama (}}). Setelah menambahkan koma tersebut, kemudian mengetikkan variabel yang telah dibuat sebelumnya (thn, bln, tgl) dan diharuskan jangan sampai urutannya salah atau terbalik (nanti hasilnya akan berbeda). Serta mengganti beberapa variabel bawaan dari tahun, bulan, dan tanggalnya (diganti menjadi thn pada year, bln pada month, dan tgl pada dayofMonth).

83. Vector Asset, berisi berbagai macam logo yang dapat digunakan untuk menunjang kebutuhan pada saat pembuatan project (seperti gambar kepala dari logo android (bawaan dari android studionya (yang bernamakan ic\_launcher\_background.xml) dan gambar kotakan yang ada gambar orang di dalamnya (yang dinamakan orang, seperti pada video))). Diharuskan memberi nama untuk Vector Assetnya dengan huruf kecil dan tanpa spasi (jarak).

84. Bisa membuat dan mengubah warna dari Vector Asset sesuai dengan yang ada pada video.

85. Bisa mengubah warna dari Vector Asset dengan cara menambahkan warna pada file dari Vector Asset yang ingin diubah (orang.xml pada video) yang ada pada folder drawable (bagian code android:fillColor=”….” (titik-titik tersebut berisikan nama maupun code dari warnanya)). Warna bisa dipilih dari file colors, menambahkan warnanya sendiri (langsung memilih), maupun dengan memasukkan heksadesimal dari warna tersebut. (Namun pada Android Studio saya, saya perlu menghilangkan (menghapus) code android:tint=”?attr/colorControlNormal” agar warna yang telah di setting untuk perubahan warna dapat berubah sesuai dengan warna yang sudah dipilihkan tersebut. Bila tidak maka warnanya akan tetap hitam seperti awal mula bawaan dari saat pertama kali dipilih (Vector Assetnya)).

86. Android:fillColor, untuk mengubah warna dari Vector Asset yang ingin diubah warnanya.

87. Bisa menampilkan Vector Asset yang telah dibuat dan dipilih sebelumnya di layer (akan tampil juga pada saat di run) dengan cara yang sama seperti pada saat menambahkan gambar. Menggunakan ImageView, tetapi kemudian memilih bagian project (karena disini tempat dari Vector Asset tersebut).

88. Menggunakan layout\_height dan layout\_width untuk memperbesar ukuran dari Vector Asset yang dimasukkan ke dalam layer.

89. CardView, untuk membuat sesuatu seperti kartu nama, yang bisa dimasukkan LinearLayout yang horizontal (untuk Vector Assetnya), maupun yang vertical (untuk teksnya) menjadi satu.

90. Bisa membuat dan menggunakan CardView sesuai dengan yang ada pada video (beserta penggunaan layoutnya dan ditambahkan Vector Asset juga).

91. CardCornerRadius (attributes khusus untuk CardView), untuk membuat tiap sisi dari Card Viewnya menjadi lebih bulat (tidak kotak seperti persegi panjang (yang tisp sisinya lancip)). Bisa diatur sesuai dengan keinginan atau selera. Satuan belakangnya menggunakan dp.

92. CardBackgroundColor, untuk mengganti warna background dari Card View yang telah dibuat sebelumnya. Bisa diubah melalui warna yang telah ada pada colors.xml, memilih warnanya langsung, maupun dengan heksadesimal dari warnanya.

93. Pada saat ingin mengganti atau mengubah sesuatu dari suatu komponen (attributesnya), maka komponen yang ingin diubah tersebut harus di klik (dipilih) terlebih dahulu. Karena bila tidak, maka tidak ada yang berubah dari suatu komponen yang ingin kita ubah attributes nya tersebut.

94. RecyclerView, untuk mengisikan banyak data di dalamnya (langsung terdapat tulisan-tulisan dari bawaan Adnroid Studionya (seperti text 0, text 1, dan masih banyak lagi lainnya)).

95. Match\_parent, digunakan untuk menjadikan ukuran lebar dan tinggi suatu komponen menjadi full (penuh) satu layer (layar).

96. Wrap\_content, digunakan untuk menjadikan ukuran lebar dan tinggi suatu komponen mengikuti isi dari yang ada di dalamnya.

97. Text Size, untuk mengubah ukuran dari suatu teks sesuai dengan yang diinginkan (belakang dari angkanya harus diberi sp (satuannya) untuk mengubah ukuran teksnya, misal seperti 24sp).

98. Text Style, untuk menjadikan suatu teks menjadi lebih tebal (bold), miring (italic), maupun normal (pada kotakan disebelah dari tulisan pilihannya ini harus di klik, untuk memakai style tersebut).

99. Bisa membuat layout dari RecyclerView sesuai dengan yang ada pada video.

100. Membuat activity\_main.xml (item\_siswa.xml) lagi (yang nantinya akan dimasukkan ke dalam Recycler View) dengan cara mengklik folder res dan memilih folder layout, kemudian klik kanan dan memilih New, yang setelahnya memilih Layout resource file, kemudian mengaturnya terlebih dahulu (seperti nama file, element rootnya, dan lain-lain), dan setelah itu klik OK. Membuatnya lagi digunakan untuk membuat list siswa nantinya.

**Saya Belum Mengerti**

1. Belum bisa mencoba logo petir (untuk menjalankan projectnya dengan lebih cepat daripada saat menekan tombol run) disamping tombol run, karena mungkin versi dari android studio saya berbeda (saya sudah coba mencari yang sama, namun belum menemukan yang sama).

2. Pada saat Activity Lifecycle, bagian onDestroy, sudah saya buat dan saya jalankan di emulatornya sesuai dengan cara untuk memunculkan System.out.println dari onDestroy tersebut. Namun tidak keluar pada Logcat saya (hanya bagian onDestroy ini, untuk lima (5) lainnya (onCreate, onStart, onResume, onPause, dan onStop) sudah bisa muncul). Namun saya sudah paham dengan fungsi onDestroy tersebut.

3.