**TASK GUIDE (B2X.05)**

1. **Tujuan.**

Student will understand how Countdowntimer works.

1. **Persyaratan.**

Hardware:

* minimal 2 GB RAM, disarankan 8 GB RAM
* minimal 2 GB disk space yang tersedia, disarankan 4 GB (500 MB

untuk IDE + 1.5 GB untuk Android SDK dan emulator system image)

* resolusi layar minimal 1280 x 800
* prosesor Intel yang mendukung fungsionalitas Intel VT-x, Intel EM64T (Intel 64), dan Execute Disable (XD) Bit

Software:

* Microsoft Windows 7/8/10 (32-bit atau 64-bit)
* JDK 8
* Android Studio IDE 3.5 (min)

1. **Referensi.**

Dokumen:

* Panduan

File Pendukung:

* -

Kode tes:

* TestB2AdvancedWidgetsX051.java

1. **Deskripsi.**

Student start to declare Countdowntimer and utilize it to update timer.

1. **Spesifikasi.**
2. Buka task B2X.04 (proyek ColorGameX) yang telah lulus pengujian.
3. Buka file “MyActivity.java” dan tambahkan field baru pada kelas MyActivity, dengan deskripsi ini.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Modifiers access** | **Initial value** |
| countDown | CountDownTimer | - | - |
| FORMAT | String | final | %d:%d |

Cara mendeklarasikan?

public class MyActivity extends AppCompatActivity {

.

.

***<Modifier> <Data type> <Field name> = <initial value>;***

.

.

}

Put here

1. Buat sebuah method private void baru dengan nama “initTime” dan parameter kosong.

|  |
| --- |
| private void initTimer() {  } |

1. Dalam method “initTimer”, definisikan countdown field dengan CountDownTimer, seperti ini.

|  |
| --- |
| countDown = new CountDownTimer(millisInFuture, countDownInterval) {  public void onTick(long millisUntilFinished) {  }  public void onFinish() {  }  }; |

Kode tersebut menunjukkan bahwa definisi CountDownTimer memiliki 2 parameter. millisInFuture adalah nilai (dalam milidetik) waktu berakhir dan countDownInterval adalah nilai langkah waktu (dalam milidetik).

Rubah nilai **millisinFuture** menjadi

|  |
| --- |
| getResources().getInteger(R.integer.maxtimer)\*1000 |

Rubah nilai **countDownInterval** menjadi

|  |
| --- |
| 1 |

Definisi tersebut juga menghasilkan 2 method, onTick dan onFinish. onTick adalah peristiwa yang akan memanggil setiap langkah waktu dan onFinish adalah peristiwa yang akan dipanggil jika waktu mencapai **millisInFuture**.

1. Pada method “onTick”, tentukan teks yang akan ditampilkan di timer (TextView). Letakkan kode ini pada method “onTick”.

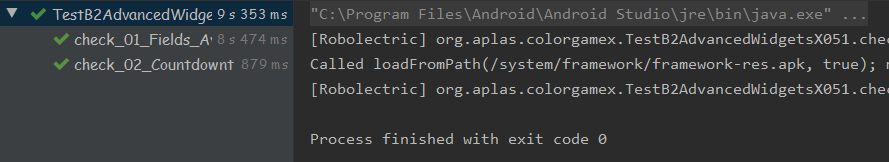
|  |
| --- |
| timer.setText(""+String.format(FORMAT, TimeUnit.MILLISECONDS.toSeconds(millisUntilFinished) - TimeUnit.MINUTES.toSeconds( TimeUnit.MILLISECONDS.toMinutes( millisUntilFinished)), TimeUnit.MILLISECONDS.toMillis(millisUntilFinished) - TimeUnit.SECONDS.toMillis( TimeUnit.MILLISECONDS.toSeconds(millisUntilFinished)))); |

Kode ini akan memperbarui teks pengatur waktu setiap milidetik.

1. Panggil method “initTimer” dalam method “onCreate”, seperti di bawah ini.

|  |
| --- |
| @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_layout);  .  .  .  **initTimer();**  } |

1. **Pengujian.**
2. Salin file “TestB2AdvancedWidgetsX051.java” ke dalam folder “org.aplas.colorgamex (test)”.
3. Klik Kanan pada file “TestB2AdvancedWidgetsX051.java” kemudian pilih Run ‘TestB2AdvancedWidgetsX051’ dan klik. Mungkin butuh waktu lama untuk mengesekusinya.
4. Dapatkan hasil dari tugasmu. Jika berhasil kamu akan mendapatkan tanda centang hijau seperti di bawah ini. Jika tes gagal, kamu akan mendapatkan centang oranye yang berisi pesan dan kamu harus memeriksa pekerjaanmu lagi.



**Anda harus mencoba sampai mendapatkan semua centang hijau dan melanjutkan ke tugas berikutnya.**

**Jalankan Aplikasinya**

Jika sudah lulus tes, tidak ada perubahan UI.