|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **SUKKUR IBA UNIVERSITY**  MERIT – QUALITY – EXCELLENCE  Lab handout 08 | | | |  |
| Mobile Application Development (CSE-426), Fall 2022, BE-EE(CS)-VIII | | | | | |
| Name: Shakir Ali | | CMS ID#: 031-19-0009 | | Instructor: Dr. Abdul Aziz | |
| Section: A Lab group: NA | | Department: Electrical Engineering | | Marks obtained : out of 100% | |
| **NOTE: Must follow submission instructions** | | |  | | |

**Understanding and using activity lifecycle**

**Lab Exercise and Submission**

|  |
| --- |
| Exercise 1 |
| Design and develop stopwatch application. Where activity is destroyed with change of orientation. |
| Screenshot of AVD output here: |
| Insert XML code for layout here:  *<?*xml version="1.0" encoding="utf-8"*?>* <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:padding="16dp"  tools:context=".MainActivity">   <TextView  android:id="@+id/time\_view"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0:00:00"  android:layout\_gravity="center\_horizontal"  android:textAppearance="@android:style/TextAppearance.Large"  android:textSize="56sp" />   <Button  android:id="@+id/start\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="20dp"  android:onClick="onClickStart"  android:text="Start"  tools:ignore="OnClick" />   <Button  android:id="@+id/stop\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickStop"  android:text="Stop" />  <Button  android:id="@+id/reset\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickReset"  android:text="Reset" />  </LinearLayout> |
| Insert Activity code here:  package com.example.shakir\_lab08\_01;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle; import android.os.Handler; import android.view.View; import android.widget.TextView;  import java.util.Locale;  public class MainActivity extends AppCompatActivity {   private int seconds = 0;  private boolean running;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);   runTimer();  }  *//Start the stopwatch running when the Start button is clicked.* public void onClickStart(View view) {  running = true;  }  *//Stop the stopwatch running when the Stop button is clicked.* public void onClickStop(View view) {  running = false;  }  *//Reset the stopwatch when the Reset button is clicked.* public void onClickReset(View view) {  running = false;  seconds = 0;  }  private void runTimer() {  final TextView timeView = (TextView)findViewById(R.id.*time\_view*);  final Handler handler = new Handler();  handler.post(new Runnable() {  @Override  public void run() {  int hours = seconds/3600;  int minutes = (seconds%3600)/60;  int secs = seconds%60;  String time = String.*format*(Locale.*getDefault*(),  "%d:%02d:%02d", hours, minutes, secs);  timeView.setText(time);  if (running) {  seconds++;  }  handler.postDelayed(this, 1000);  }  });  } } |

|  |
| --- |
| Exercise 2 |
| Update the stopwatch application such that time continues while switching in between landscape and portrait mode. |
| Screenshot of AVD output here: |
| Insert XML code for layout here:  *<?*xml version="1.0" encoding="utf-8"*?>* <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:padding="16dp"  tools:context=".MainActivity">   <TextView  android:id="@+id/time\_view"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0:00:00"  android:layout\_gravity="center\_horizontal"  android:textAppearance="@android:style/TextAppearance.Large"  android:textSize="56sp" />   <Button  android:id="@+id/start\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="20dp"  android:onClick="onClickStart"  android:text="Start"  tools:ignore="OnClick" />   <Button  android:id="@+id/stop\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickStop"  android:text="Stop" />  <Button  android:id="@+id/reset\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickReset"  android:text="Reset" />  </LinearLayout> |
| Insert Activity code here:  package com.example.shakir\_lab08\_02;  import androidx.appcompat.app.AppCompatActivity;   import android.os.Bundle; import android.os.Handler; import android.view.View; import android.widget.TextView; import java.util.Locale;  public class MainActivity extends AppCompatActivity {   private int seconds = 0;  private boolean running;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  if (savedInstanceState != null) {  seconds = savedInstanceState.getInt("seconds");  running = savedInstanceState.getBoolean("running");  }  runTimer();  }   @Override  public void onSaveInstanceState(Bundle savedInstanceState) {  super.onSaveInstanceState(savedInstanceState);  savedInstanceState.putInt("seconds", seconds);  savedInstanceState.putBoolean("running", running);  }   *//Start the stopwatch running when the Start button is clicked.* public void onClickStart(View view) {  running = true;  }  *//Stop the stopwatch running when the Stop button is clicked.* public void onClickStop(View view) {  running = false;  }  *//Reset the stopwatch when the Reset button is clicked.* public void onClickReset(View view) {  running = false;  seconds = 0;  }   private void runTimer() {  final TextView timeView = (TextView)findViewById(R.id.*time\_view*);  final Handler handler = new Handler();  handler.post(new Runnable() {  @Override  public void run() {  int hours = seconds/3600;  int minutes = (seconds%3600)/60;  int secs = seconds%60;  String time = String.*format*(Locale.*getDefault*(),  "%d:%02d:%02d", hours, minutes, secs);  timeView.setText(time);  if (running) {  seconds++;  }  handler.postDelayed(this, 1000);  }  });  } } |

|  |
| --- |
| Exercise 3 |
| Further extend the application such that when you click on the Start button the timer starts: it stops when the app is no longer visible, and it starts again when the app becomes visible again. |
| Screenshot of AVD output here: |
| Insert XML code for layout here:  *<?*xml version="1.0" encoding="utf-8"*?>* <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:padding="16dp"  tools:context=".MainActivity">   <TextView  android:id="@+id/time\_view"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0:00:00"  android:layout\_gravity="center\_horizontal"  android:textAppearance="@android:style/TextAppearance.Large"  android:textSize="56sp" />   <Button  android:id="@+id/start\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="20dp"  android:onClick="onClickStart"  android:text="Start"  tools:ignore="OnClick" />   <Button  android:id="@+id/stop\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickStop"  android:text="Stop" />  <Button  android:id="@+id/reset\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickReset"  android:text="Reset" />  </LinearLayout> |
| Insert Activity code here:  package com.example.shakir\_lab08\_03;  import androidx.appcompat.app.AppCompatActivity;   import android.os.Bundle; import android.os.Handler; import android.view.View; import android.widget.TextView; import java.util.Locale;  public class MainActivity extends AppCompatActivity {   private int seconds = 0;  private boolean running;  private boolean wasRunning;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  if (savedInstanceState != null) {  seconds = savedInstanceState.getInt("seconds");  running = savedInstanceState.getBoolean("running");  wasRunning = savedInstanceState.getBoolean("wasRunning");  }  runTimer();  }   @Override  public void onSaveInstanceState(Bundle savedInstanceState) {  super.onSaveInstanceState(savedInstanceState);  savedInstanceState.putInt("seconds", seconds);  savedInstanceState.putBoolean("running", running);  savedInstanceState.putBoolean("wasRunning", wasRunning);  }   @Override  protected void onStop() {  super.onStop();  wasRunning = running;  running = false;  }  @Override  protected void onStart() {  super.onStart();  if (wasRunning) {  running = true;  }  }   *//Start the stopwatch running when the Start button is clicked.* public void onClickStart(View view) {  running = true;  }  *//Stop the stopwatch running when the Stop button is clicked.* public void onClickStop(View view) {  running = false;  }  *//Reset the stopwatch when the Reset button is clicked.* public void onClickReset(View view) {  running = false;  seconds = 0;  }   private void runTimer() {  final TextView timeView = (TextView)findViewById(R.id.*time\_view*);  final Handler handler = new Handler();  handler.post(new Runnable() {  @Override  public void run() {  int hours = seconds/3600;  int minutes = (seconds%3600)/60;  int secs = seconds%60;  String time = String.*format*(Locale.*getDefault*(),  "%d:%02d:%02d", hours, minutes, secs);  timeView.setText(time);  if (running) {  seconds++;  }  handler.postDelayed(this, 1000);  }  });  } } |

|  |
| --- |
| Exercise 4 |
| Further extend the application such that when you click on the Start button, the timer starts; it stops when the app is partially obscured by another activity; and it starts again when the app is back in the foreground. |
| Screenshot of AVD output here: |
| Insert XML code for layout here:  *<?*xml version="1.0" encoding="utf-8"*?>* <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:padding="16dp"  tools:context=".MainActivity">   <TextView  android:id="@+id/time\_view"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0:00:00"  android:layout\_gravity="center\_horizontal"  android:textAppearance="@android:style/TextAppearance.Large"  android:textSize="56sp" />   <Button  android:id="@+id/start\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="20dp"  android:onClick="onClickStart"  android:text="Start"  tools:ignore="OnClick" />   <Button  android:id="@+id/stop\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickStop"  android:text="Stop" />  <Button  android:id="@+id/reset\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_horizontal"  android:layout\_marginTop="8dp"  android:onClick="onClickReset"  android:text="Reset" />  </LinearLayout> |
| Insert Activity code here:  package com.example.shakir\_lab08\_04;  import androidx.appcompat.app.AppCompatActivity;   import android.os.Bundle; import android.os.Handler; import android.view.View; import android.widget.TextView; import java.util.Locale;  public class MainActivity extends AppCompatActivity {   private int seconds = 0;  private boolean running;  private boolean wasRunning;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  if (savedInstanceState != null) {  seconds = savedInstanceState.getInt("seconds");  running = savedInstanceState.getBoolean("running");  wasRunning = savedInstanceState.getBoolean("wasRunning");  }  runTimer();  }   @Override  public void onSaveInstanceState(Bundle savedInstanceState) {  super.onSaveInstanceState(savedInstanceState);  savedInstanceState.putInt("seconds", seconds);  savedInstanceState.putBoolean("running", running);  savedInstanceState.putBoolean("wasRunning", wasRunning);  }   @Override  protected void onPause() {  super.onPause();  wasRunning = running;  running = false;  }  @Override  protected void onResume() {  super.onResume();  if (wasRunning) {  running = true;  }  }   *//Start the stopwatch running when the Start button is clicked.* public void onClickStart(View view) {  running = true;  }  *//Stop the stopwatch running when the Stop button is clicked.* public void onClickStop(View view) {  running = false;  }  *//Reset the stopwatch when the Reset button is clicked.* public void onClickReset(View view) {  running = false;  seconds = 0;  }   private void runTimer() {  final TextView timeView = (TextView)findViewById(R.id.*time\_view*);  final Handler handler = new Handler();  handler.post(new Runnable() {  @Override  public void run() {  int hours = seconds/3600;  int minutes = (seconds%3600)/60;  int secs = seconds%60;  String time = String.*format*(Locale.*getDefault*(),  "%d:%02d:%02d", hours, minutes, secs);  timeView.setText(time);  if (running) {  seconds++;  }  handler.postDelayed(this, 1000);  }  });  } } |
| Manifest code here:  *<?*xml version="1.0" encoding="utf-8"*?>* <manifest xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  package="com.example.shakir\_lab08\_04">   <application  android:allowBackup="true"  android:dataExtractionRules="@xml/data\_extraction\_rules"  android:fullBackupContent="@xml/backup\_rules"  android:icon="@mipmap/ic\_launcher"  android:label="@string/app\_name"  android:roundIcon="@mipmap/ic\_launcher\_round"  android:supportsRtl="true"  android:theme="@style/Theme.Shakir\_Lab08\_04"  tools:targetApi="31">  <activity  android:name=".MainActivity"  android:exported="true">  <intent-filter>  <action android:name="android.intent.action.MAIN" />   <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  </activity>  </application>  </manifest> |