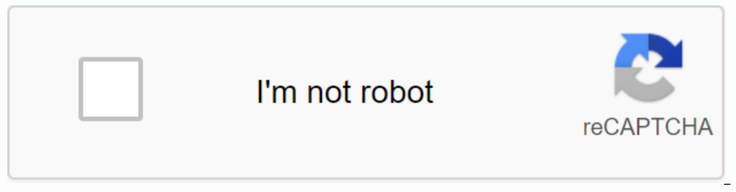

Continue

Android add textview to scrollview programmatically

Tanpri ouvri sesyon an oswa enskri pou reponn kesyon sa a. Tanpri ouvri sesyon an oswa enskri yo ajoute yon komante. Mwen gen sa ki annapre yo: -Yon kat jeyografik ak Strings tankou: [Koule: ble, Gwose: gwo] rele detay Yon ScrollView ki deja egziste ak yon LinearLayout ak ki deja egziste TextViews: <ScrollView> <LinearLayout> <TextView android:text=something: else></TextView> </LinearLayout> </ScrollView> Mwen te ommited jaden komen tankou laje, wote, xml schemas sou bi. Koulye a, mwen vle ajoute textViews pwogram otomatikman. Mwen pa konnen hoy anpil yo se: Detay TextView detay; LinearLayout llay = (LinearLayout)fragmentView.findViewById (R.id.id.container); pou (int mwen = 0; < detailsarray.length;= i++)= {= detail=new textview(fragmentview.getcontext());= detail.setlayoutparams(new= layoutparams(layoutparams.wrap_content,layoutparams.wrap_content));= detail.settext(detailsarray[i]);= llay.addview(detail);= }= scrollview= sv=(ScrollView) fragmentview.findviewbyid(r.id.scroll_view);= sv.addview(llay);= but= i'm= getting= an= exception:= 04-17= 12:38:09.975:= e/androidruntime(3361):= caused= by:= java.lang.illegalstateexception:= scrollview= can= host= only= one= direct= child= what= should= i= do?= thank= you= in= advance.= this= is= the= xml= code:=> <LinearLayout android:id=@+id/mainLayout xmlns:android= xmlns:tools= android:layout_width=match_parent android:layout_height=match_parent android:paddingbottom=@dimen/activity_vertical_margin android:paddingleft=@dimen/activity_horizontal_margin android:paddingright=@dimen/activity_horizontal_margin android:paddingtop=@dimen/activity_vertical_margin tools:context=. MainActivity android:orientation=vertical android:background=@color/black> <ScrollView android:id=@+id/mainScrollView android:layout_width=match_parent android:layout_height=wrap_content android:background=@color/teal android:layout_weight=.50> <LinearLayout android:id=@+id/linearLayout1 android:layout_width=match_parent android:layout_height=match_parent android:orientation=vertical></LinearLayout> </ScrollView> <Button android:id=@+id/addButton android:layout_width=match_parent android:layout_height=wrap_content android:text=+ android:gravity=center android:layout_weight=.20></Button> <TextView android:id=@+id/totalHoursLabel android:text=Ore Totali android:background=@color/gray android:textcolor=@color/red android:gravity=center android:layout_width=match_parent android:layout_height=wrap_content android :layout_weight=.10></TextView> <TextView android:id=@+id/totalHoursView android:text=00:00 android:background=@color/white android:textcolor=@color/black android:gravity=center android:layout_width=match_parent android:layout_height=wrap_content </LinearLayout> My problem is that when I try to add a look at the ScrollView or inner it's LinearLayout my program crashes. is my java code: super.onCreate(savedInstanceState); LinearLayout ll = (LinearLayout)getViewById(R.id.linearLayout1); TextView tv = new TextView(this); tv.setText(testView); tv.setLayoutParams(new LayoutParams(LayoutParams.WRAP_CONTENT, LayoutParams.WRAP_CONTENT)); ll.addView(tv); setContentView(R.layout.activity_main); If I create a new Android project this code (with only a relatifLayout and everything else blank in the xml file) works fine. I think there are some issues with the findViewById method or my xml file. Any ideas? This is my LogCat file: 07-23 15:56:19.119: D/AndroidRuntime(1249): Shutting down VM 07-23 15:56:19.119: W/dalvikvm(1249): threadid=1: thread exiting with uncaught exception (group=0x40a71930) 07-23 15:56:19.159: E/AndroidRuntime(1249): FATAL EXCEPTION: main 07-23 15:56:19.159: E/AndroidRuntime(1249): java.lang.RuntimeException: Unable to start activity ComponentInfo{workTimer.worktimer/workTimer.worktimer.MainActivity}: java.lang.NullPointerException 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2180) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2230) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.ActivityThread.access$600(ActivityThread.java:141) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1234) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.os.Handler.dispatchMessage(Handler.java:99) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.os.Looper.loop(Looper.java:137) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.ActivityThread.main(ActivityThread.java:5041) 07-23 15:56:19.159: E/AndroidRuntime(1249): at java.lang.reflect.Method.invokeNative(Native Method) 07-23 15:56:19.159 : E/AndroidRuntime(1249): at java.lang.reflect.Method.invoke(Method.java:511) 07-23 15:56:19.159: E/AndroidRuntime(1249): at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:793) 07-23 15:56:19.159: E/AndroidRuntime(1249): at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:560) 07-23 15:56:19.159: E/AndroidRuntime(1249): at dalvik.system.NativeStart.main(Native Method) 07-23 15:56:19.159: E/AndroidRuntime(1249): Caused by: java.lang.NullPointerException 07-23 15:56:19.159: E/AndroidRuntime(1249): at workTimer.worktimer.MainActivity.onCreate(MainActivity.java:32) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.Activity.performCreate(Activity.java:5104) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1080) 07-23 15:56:19.159: E/AndroidRuntime(1249): at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2144) 07-23 15:56:19.159: ... 11 plus 07-23 07-23 I / Process (1249): Send signals. PID: 1249 SIG: 9 I have one table TABLE_SUBJECT including a number of subjects. I need to create a horizontal roller view and subject. How do I create a ScrollView and Program Database article automatically? If I enter 1o subject then it will be displayed in scroll view as a button. Is it possible? Hello Readers! In this post, we are going to learn how to create and android scrollView programs automatically in any Android application. We will also learn to add scrollView to linearLayout programs automatically in any application. From Tutorialwing android dynamic ScrollView Output will start at first, we will create an Android application. Then we'll use the scrollView widget in the Application.1. Create new projects follow the steps below to create a new project. Please ignore the following steps if you have already created a new project. Step Description 1. Open Android Studio. 2. Go to File => New => New Project. Write the name of the application as dynamicScrollView. Then click Next button. 3. Select the minimum SDK you need. However, we chose 17 as minimum SDK. Then click Next button 4. Then select Blank Activity => next click => click Finish. 5. If you followed above process correctly, you will get a newly created project successfully. However, you can also visit post to create a new project to know the detail step. Now we will modify xml and java files to use Android scrollView program automatically. 2. Modify the folder value Open rest / value / strings.xml file and add below code into it. <resources><string name=app_name>DynamicScrollView</string></resources>3. Download Drawable Resource needed You need some images, stored in unused/drawable folder, that will be used in the application. These draw images will be used by scrollView created dynamically in the Application.4. Modify File Layout Open/Layout/activity_main.xml file. Then add below code into it. <?xml version=1.0 encoding=utf-8?><LinearLayout android:id=@+id/rootContainer xmlns:android= android.android.com/apk/res/android android:layout_width=match_parent android:layout_height=wrap_content android:orientation=vertical></LinearLayout> In activity_main.xml file, we defined lineyeLayout, and rootContainer id, which will act as container for the scrollView widget created programmedly in the application.5. Create Android ScrollView Programmatic / Dynamically Open app / src / main / java / com.tutorialwing.dynamicscrollview / MainActivity.java file and add below code into it.pack com.tutorialwing.dynamicscrollview; import android.os.Bundle; import android.support.v7.app.AppCompatActivity; import android.view.Gravity; android import. ViewGroup; import android.widget.ImageView; import android.widget.LinearLayout; import public class MainActivite Extended AppCompatActivite { @Override Cancellation Cancellation savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_main); ScrollView scrollView = new ScrollView(this); LinearLayout.LayoutParams LayoutParams = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.MATCH_PARENT); scrollView.setLayoutParams(LayoutParams); LinearLayout linearyout = new LinearLayout(this); LinearLayout.LayoutParams linearParams = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); linearLayout.setOrientation(LinearLayout.VERTICAL); linearLayout.setLayoutParams(linearParams); scrollView.addView(linearLayout); ImageView imageView1 = new ImageView(this); LinearLayout.LayoutParams params1 = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); params1.setMargins(0, 30, 0, 30); params1.Gravity = Gravity.CENTER; imageView1.setLayoutParams(params1); imageView1.setImageResource(R.drawable.guava); linearLayout.addView(imageView1); ImageView imageView2 = new ImageView(this); LinearLayout.LayoutParams params2 = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); params2.setMargins(0, 0, 0, 30); params2.Gravity = Gravity.CENTER; imageView2.setLayoutParams(params2); imageView2.setImageResource(R.drawable.jackfruit); linearLayout.addView(imageView2); ImageView imageView3 = new ImageView(this); LinearLayout.LayoutParams params3 = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); params3.setMargins(0, 0, 0, 30); params3.gravity = Gravity.CENTER; imageView3.setLayoutParams(params3); imageView3.setImageResource(R.drawable.mix_fruit); linearLayout.addView(imageView3); ImageView imageView4 = new ImageView(this); LinearLayout.LayoutParams params4 = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); params4.setMargins(0, 0, 0, 30); params4.Gravity = Gravity.CENTER; imageView4.setLayoutParams(params4); imageView4.setImageResource(R.drawable.pomegranate); linearLayout.addView(imageView4); ImageView imageView5 = new ImageView(this); LinearLayout.LayoutParams params5 = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); params5.setMargins(0, 0, 0, 30); params5.gravity = Gravity.CENTER; imageView5.setLayoutParams(params5); imageView5.setImageResource(R.drawable.strawberry); linearLayout.addView(imageView5); ImageView imageView6 = new ImageView(this); LinearLayout.LayoutParams params6 = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, ViewGroup.LayoutParams.WRAP_CONTENT); params6.setMargins(0, 0, 0, 30); params6.gravity = Gravity.CENTER; imageView6.setLayoutParams(params6); (R.drawable.zespri_kiwi); (R.drawable.zespri_kiwi); LinearLayout linearyout1 = findViewById(R.id.rootContainer); if (linearLayout1 != null) { linearLayout1.addView(scrollView); } We created scrollView of java files (i.e. in MainActivity.java file). Then we created and added a lineyeLayout as the child direct of scrollView. After that we created and added 6 widgets imageView to that linearLayout. After that, we added scrollView to linearLayout, it had rootContainer id. Since AndroidManifest.xml file is very important to any Android project. We are also going to see the content inside this file. AndroidManifest.xml file code inside src/main/androidManifest.xml file would look like below – <?xml version=1.0 encoding=utf-8?>When we run the application, we will find output as shown above.<manifest package=com.tutorialwing.dynamicscrollview xmlns:android= amp;gt; <application android:allowbackup=true android:icon=@mipmap/ic_launcher android:label=@string/app_name android:roundicon=@mipmap/ic_launcher_round android:supportsrtl=true android:theme=@style/AppTheme><activity android:name=. MainActivity><intent-filter><android action:name=android.intent.action.MAIN></action> <category android:name=android.intent.category.LAUNCHER></category></intent-filter></activity></application></manifest> This at the end of tutorial on Creating android ScrollView Program automatically. If you like tutorials and would like to contribute, you can email an article about any educational topic in tutorialwing@gmail.com. We would like to publish your article. See your article on Tutorialwing and help others with your knowledge. Follow Facebook, LinkedIn, Google+, Twitter, Youtube for latest updates. Update.

normal_5f9496f9dc22d.pdf
normal_5f8a6b0e5ba03.pdf
normal_5f8cb1996b6a2.pdf