What is TextView

The TextView class is a subclass of the View class that displays text on the screen. You can control how the text appears with TextView attributes in the XML layout file. This practical shows how to work with multiple TextView elements, including one in which the user can scroll its contents vertically.

If you have more information than fits on the device's display, you can create a scrolling view so that the user can scroll vertically by swiping up or down, or horizontally by swiping right or left.

You would typically use a scrolling view for news stories, articles, or any lengthy text that doesn't completely fit on the device display. You can also use a scrolling view to enable users to enter multiple lines of text, or to combine UI elements (such as a text field and a button) within a scrolling view.
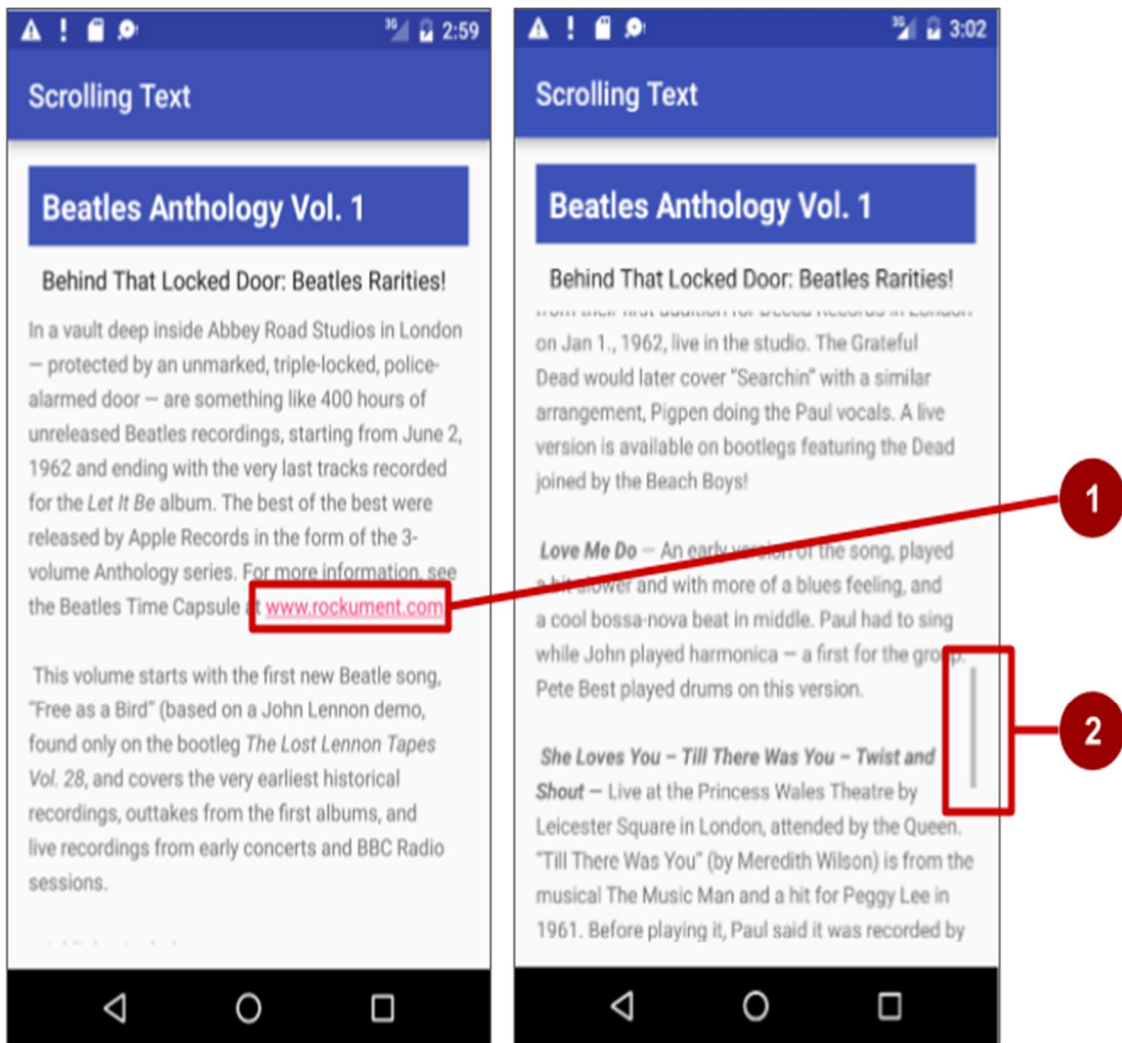
What is ScrollView

The ScrollView class provides the layout for the scrolling view. ScrollView is a subclass of FrameLayout, and developers should place only one view as a child within it, where the child view contains the entire contents to scroll. This child view may itself be a view group (such as a layout manager like LinearLayout) with a complex hierarchy of objects. Note that complex layouts may suffer performance issues with child views such as images. A good choice for a view within a ScrollView is a LinearLayout that is arranged in a vertical orientation, presenting top-level items that the user can scroll through.

With a ScrollView, all of the views are in memory and in the view hierarchy even if they aren't displayed on screen. This makes ScrollView ideal for scrolling pages of free-form text smoothly, because the text is already in memory. However, ScrollView can use up a lot of memory, which can affect the performance of the rest of your app. To display long lists of items that users can add to, delete from, or edit, consider using a RecyclerView, which is described in a separate practical.

App overview

The Scrolling Text app demonstrates the ScrollView UI component. ScrollView is a view group that in this example contains a TextView. It shows a lengthy page of text—in this case, a music album review—that the user can scroll vertically to read by swiping up and down. A scroll bar appears in the right margin. The app shows how you can use text formatted with minimal HTML tags for setting text to bold or italic, and with new-line characters to separate paragraphs. You can also include active web links in the text.
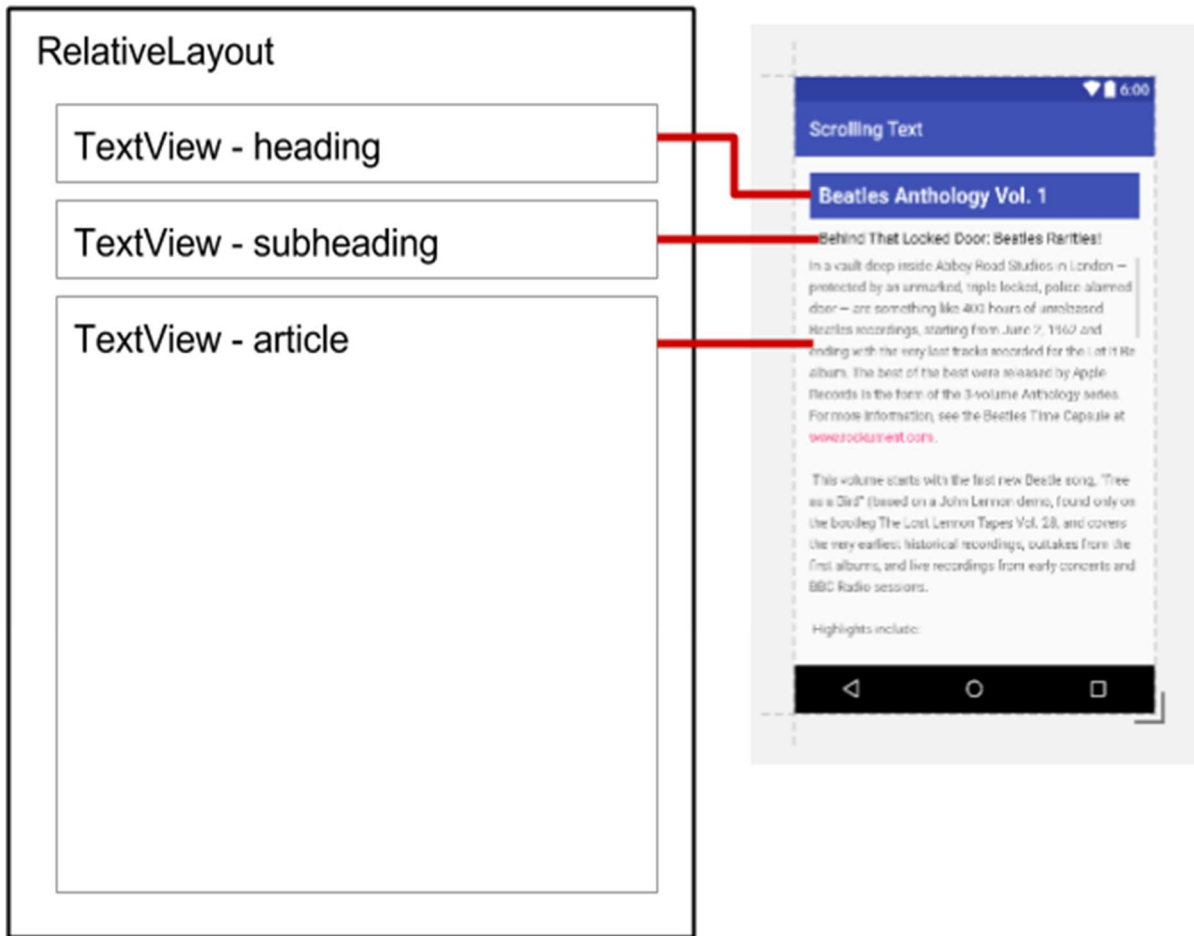
In the above figure, the following appear:
1. An active web link embedded in free-form text
2. The scroll bar that appears when scrolling the text

Add several text views

In this practical, you will create an Android project for the Scrolling Text app, add TextViews to the layout for an article title and subtitle, and change the existing "Hello World" TextView element to show a lengthy article. The figure below is a diagram of the layout.

Project

Add the below code in the xml file

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/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="com.example.android.scrollingtext.MainActivity">

    <TextView
        android:id="@+id/article_heading"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorPrimary"
        android:textColor="@android:color/holo_orange_light"
        android:textColorHighlight="@color/colorAccent"
        android:padding="10dp"
        android:textAppearance="@android:style/TextAppearance.Large"
```

```
        android:textStyle="bold"
        android:text="@string/article_title"/>

    <TextView
        android:id="@+id/article_subheading"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/article_heading"
        android:padding="10dp"
        android:textAppearance="@android:style/TextAppearance"
        android:text="@string/article_subtitle"/>

    <TextView
        android:id="@+id/article"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/article_subheading"
        android:lineSpacingExtra="5sp"
        android:text="@string/article_text"/>

</RelativeLayout>
```

Add the text of an article

For this practical, you will create the article as a single long string in the strings.xml resource.

- In the **app > res > values** folder, open **strings.xml**.
- Enter the values for the strings *article_title and article_subtitle* with a made-up title and a subtitle for the article you are adding. The string values for each should be single-line text without HTML tags or multiple lines.
- Enter or copy and paste text for the *article_text* string.
- You can use the text of your own in the **strings.xml**.
- Here are some tags which will help you in **strings.xml**.

  o As you enter or paste text in the strings.xml file, the text lines don't wrap around to the next line—they extend beyond the right margin. This is the correct behavior—each new line of text starting at the left margin represents an entire paragraph.

  o Enter to represent the end of a line, and another to represent a blank line.

  o If you have an apostrophe (') in your text, you must escape it by preceding it with a backslash ('). If you have a double-quote in your text, you must also escape it ("). You must also escape any other non-ASCII characters.See the "Formatting and Styling" section of String Resources for more details.

  o Enter the HTML and tags around words that should be in bold.

  o Enter the HTML and tags around words that should be in italics. Note, however, that if you use curled apostrophes within an italic phrase, you should replace them with straight apostrophes.

  o You can combine bold and italics by combining the tags, as in ... words.... Other HTML tags are ignored.

- o Enclose The entire text within <string name="article_text"> </string> in the strings.xml file.

- o Include a web link to test, such as www.google.com

- Below is the text provided in the **strings.xml**.

<resources>
   <string name="article_title"**>**Beatles Anthology Vol. 1**</string>**

   <string name="article_subtitle"**>**Behind That Locked Door: Beatles Rarities!**</string>**

   <string name="article_text"**>**In a vault deep inside Abbey Road Studios in London — protected by an unmarked, triple-locked, police-alarmed door — are something like 400 hours of unreleased Beatles recordings, starting from June 2, 1962 and ending with the very last tracks recorded for the **<i>**Let It Be**</i>** album. The best of the best were released by Apple Records in the form of the 3-volume Anthology series.

 For more information, see the Beatles Time Capsule at www.rockument.com.
\n\n

   This volume starts with the first new Beatle song, "Free as a Bird" (based on a John Lennon demo, found only on the bootleg **<i>**The Lost Lennon Tapes Vol. 28**</i>**, and covers the very earliest historical recordings, outtakes from the first albums, and live recordings from early concerts and BBC Radio sessions.
\n\n

****Highlights include:****
\n\n

<i>Cry for a Shadow**</i>** - Many a Beatle fanatic started down the bootleg road, like I did, with a first listen to this song. Originally titled "Beatle Bop" and recorded in a single session that yielded four songs (the other three featured Tony Sheridan with the Beatles as a backing band), "Cry for a Shadow" is an instrumental written by Lennon and Harrison, which makes it unique to this day. John Lennon plays rhythm guitar, George Harrison plays lead guitar, Paul McCartney plays bass, and Pete Best plays drums. The sessions were produced by Bert Kaempfert in Hamburg, Germany, during the Beatles' second visit from April through July of 1961 to play in the Reeperbahn-section clubs.
\n\n

<i>My Bonnie**</i>** and **<i>**Ain't She Sweet**</i>** — At the same session, the Beatles played on "My Bonnie" (the first-ever single with Beatles playing), as the backing band for English singer Tony Sheridan, originally a member of the Jets. The popularity of this single in Liverpool brought the Beatles to the attention of Brian Epstein, who worked in the NEMS record store and tried to meet demand for the disc. John Lennon then sings a fine "Ain't She Sweet" (his first-ever released vocal).
\n\n

<i>Searchin**</i>** — A Jerry Leiber - Mike Stoller comedy song that was a hit for the Co

asters in 1957, and a popular live favorite of the Beatles. The Coasters also had a hit with "Besam e Mucho" and the Beatles covered that song as well. Ringo Starr had by now replaced Pete Best o n drums. The high falsetto is George, who also plays a hesitant lead guitar. This is from their first audition for Decca Records in London on Jan 1., 1962, live in the studio. The Grateful Dead woul d later cover "Searchin" with a similar arrangement, Pigpen doing the Paul vocals. A live version is available on bootlegs featuring the Dead joined by the Beach Boys!
\n\n

<i>Love Me Do</i> — An early version of the song, played a bit slower and with more of a blues feeling, and a cool bossa-nova beat in middle. Paul had to sing while John played harm onica — a first for the group. Pete Best played drums on this version.
\n\n

<i>She Loves You – Till There Was You – Twist and Shout</i> — Live at the Princess Wales Theatre by Leicester Square in London, attended by the Queen. "Till There Was You" (by Meredith Wilson) is from the musical The Music Man and a hit for Peggy Lee in 1961. Before pl aying it, Paul said it was recorded by his favorite American group, "Sophie Tucker" (which got s ome laughs). At the end, John tells the people in the cheaper seats to clap their hands, and the rest to "rattle your jewelry" and then announces "Twist and Shout" (a song by Bert Russell and Phil Medley that was first recorded in 1962 by the Isley Brothers). A film of the performance shows th e Queen smiling at John's remark.
\n\n

<i>Leave My Kitten Alone</i> — One of the lost Beatle songs recorded during the "Be atles For Sale" sessions but never released. This song, written by Little Willie John, Titus Turner, and James McDougal, was a 1959 R&B hit for Little Willie John and covered by Johnny Pre ston before the Beatles tried it and shelved it. A reference to a "big fat bulldog" may have influen ced John's "Hey Bulldog" (Yellow Submarine album), which is a similar rocker.
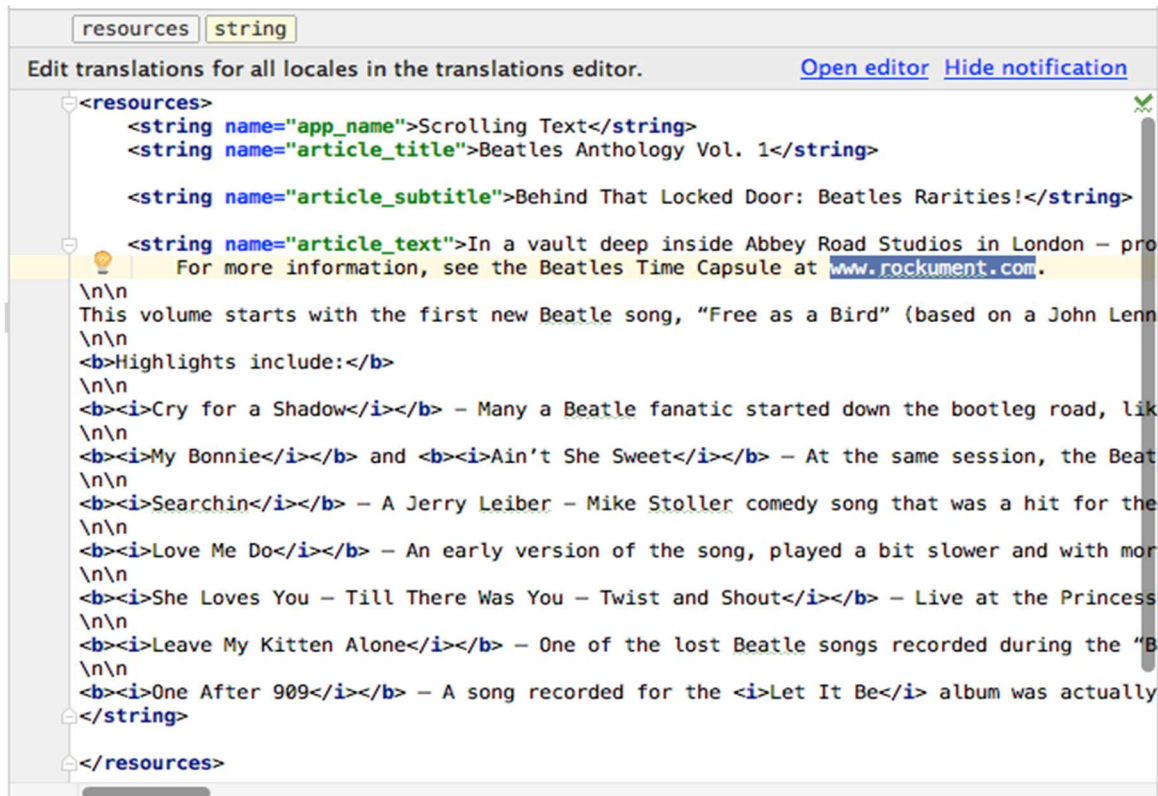\n\n

<i>One After 909</i> — A song recorded for the <i>Let It Be</i> album was actually worked on way back in the beginning, six years earlier. This take shows how they did it much mo re slowly, with an R&B feel to it.
</string>

</resources>

• Below Image shows how it will be displayed in the strings.xml

```
resources   string
```

Edit translations for all locales in the translations editor.   Open editor Hide notification

```
<resources>
    <string name="app_name">Scrolling Text</string>
    <string name="article_title">Beatles Anthology Vol. 1</string>

    <string name="article_subtitle">Behind That Locked Door: Beatles Rarities!</string>

    <string name="article_text">In a vault deep inside Abbey Road Studios in London – pro
        For more information, see the Beatles Time Capsule at www.rockument.com.
\n\n
This volume starts with the first new Beatle song, "Free as a Bird" (based on a John Lenn
\n\n
<b>Highlights include:</b>
\n\n
<b><i>Cry for a Shadow</i></b> – Many a Beatle fanatic started down the bootleg road, lik
\n\n
<b><i>My Bonnie</i></b> and <b><i>Ain't She Sweet</i></b> – At the same session, the Beat
\n\n
<b><i>Searchin</i></b> – A Jerry Leiber – Mike Stoller comedy song that was a hit for the
\n\n
<b><i>Love Me Do</i></b> – An early version of the song, played a bit slower and with mor
\n\n
<b><i>She Loves You – Till There Was You – Twist and Shout</i></b> – Live at the Princess
\n\n
<b><i>Leave My Kitten Alone</i></b> – One of the lost Beatle songs recorded during the "B
\n\n
<b><i>One After 909</i></b> – A song recorded for the <i>Let It Be</i> album was actually
</string>

</resources>
```

Add the autoLink attribute for active web links

• Add the android:autoLink="web" attribute to the article TextView. The XML code for this TextView should now look like this:

```
<TextView
  android:id="@+id/article"
  ...
  android:autoLink="web"
  ... />
```

Add a ScrollView to the layout

To make a view (such as a TextView) scrollable, embed the view inside a ScrollView.

1. Add a ScrollView between the article_subheading TextView and the article TextView. As you enter at the end, and presents the android:layout_width and android:layout_height attributes with suggestions. Choose wrap_content from the suggestions for both attributes. The code should now look like this:

```
<TextView
  android:id="@+id/article_subheading"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:layout_below="@id/article_heading"
  android:padding="10dp"
  android:textAppearance="@android:style/TextAppearance"
  android:text="@string/article_subtitle"/>

<ScrollView
  android:layout_width="wrap_content"
```

```
  android:layout_height="wrap_content"
  android:layout_below="@id/article_subheading"></ScrollView>
<TextView
  android:id="@+id/article"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:layout_below="@id/article_subheading"
  android:lineSpacingExtra="5sp"
  android:autoLink="web"
  android:text="@string/article_text"/>
```
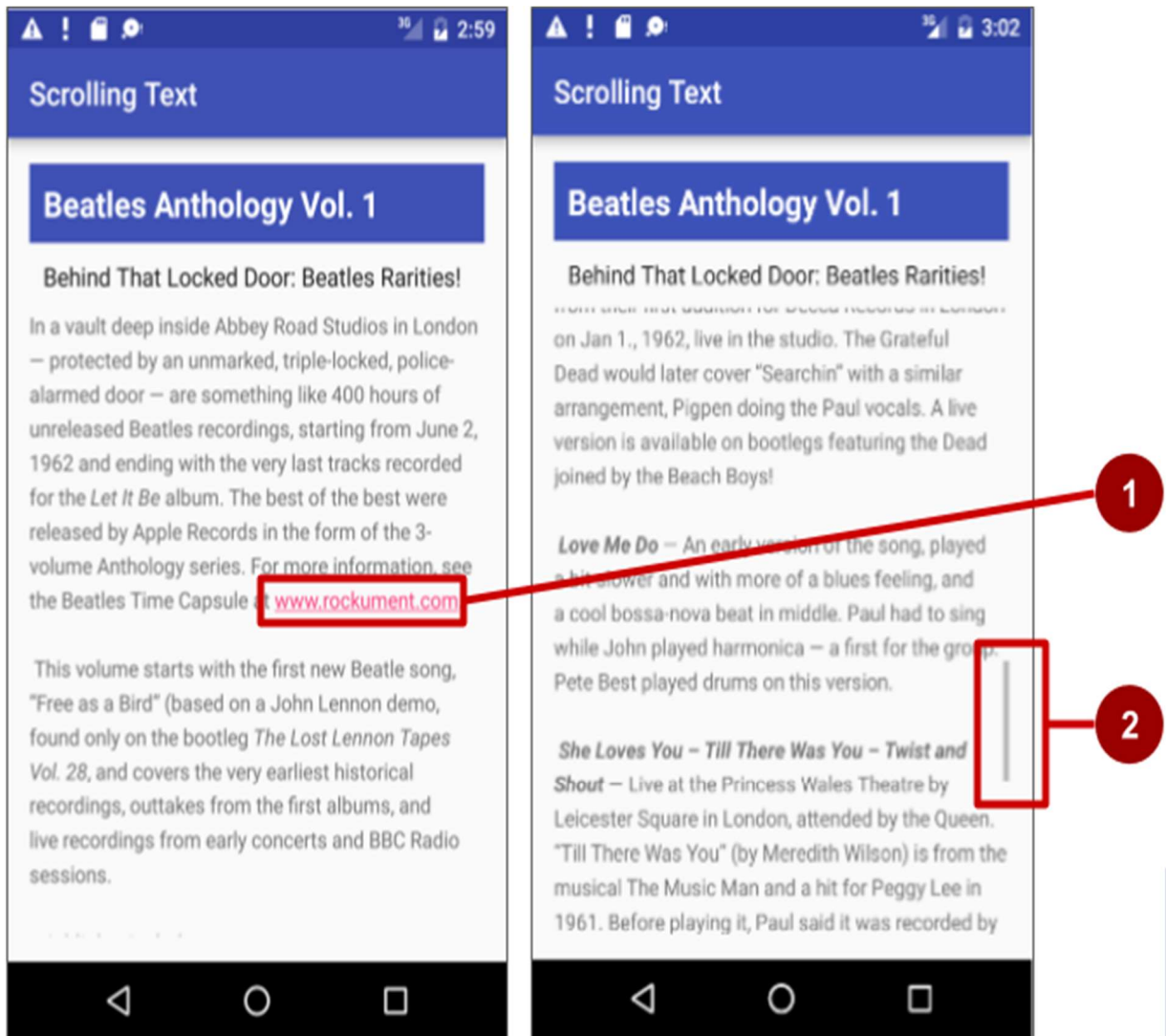
2. Move the ending code after the article TextView so that the article TextView attributes are inside the ScrollView XML element.

3. Remove the following attribute from the article TextView, because the ScrollView itself will be placed below the article_subheading element, and this attribute for TextView would conflict with the ScrollView:

   android:layout_below="@id/article_subheading"

4. The layout should now look like this:

5. Choose Code > Reformat Code to reformat the XML code so that the article TextView now appears indented inside the <Scrollview code. Run the app.

    a.   Swipe up and down to scroll the article. The scroll bar appears in the right margin as you scroll.

    b.   Tap the web link to go to the web page. The android:autoLink attribute turns any recognizable URL in the TextView (such as www.rockument.com) into a web link.

6.   Rotate your device or emulator while running the app. Notice how the scrolling view widens to use the full display and still scrolls properly.

7.   Run the app on a tablet or tablet emulator. Notice how the scrolling view widens to use the full display and still scrolls properly.Scrolling Formatted Text

In the above figure, the following appear:
1.   An active web link embedded in free-form text
2.   The scroll bar that appears when scrolling the text Scrolling Text on a Tablet :

Depending on your version of Android Studio, the activity_main.xml layout file will now look something like the following:

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/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="com.example.android.scrollingtext.MainActivity">

    <TextView
        android:id="@+id/article_heading"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorPrimary"
        android:textColor="@android:color/white"
        android:paddingTop="10dp"
        android:paddingBottom="10dp"
        android:paddingLeft="10dp"
        android:paddingRight="10dp"
        android:textAppearance="@android:style/TextAppearance.Large"
```

10

```xml
        android:textStyle="bold"
        android:text="@string/article_title"/>

    <TextView
        android:id="@+id/article_subheading"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/article_heading"
        android:paddingTop="10dp"
        android:paddingBottom="10dp"
        android:paddingLeft="10dp"
        android:paddingRight="10dp"
        android:textAppearance="@android:style/TextAppearance"
        android:text="@string/article_subtitle"/>

    <ScrollView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/article_subheading">

        <TextView
            android:id="@+id/article"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:lineSpacingExtra="5sp"
            android:autoLink="web"
            android:text="@string/article_text"/>

    </ScrollView>
</RelativeLayout>
```