

Ex. No: 11

Date:

RSS FEED APPLICATION

Aim

To develop an application that makes use of RSS feed

Algorithm

Step 1: Start the program

Step 2: Open a new android project File → new → Android Application Project

Step 3: Design the required tool in the palette window

3(a): In the activity_main.xml file use the list view to display the courses available

3(b): in rss_list_layout.xml file use the textview components to view the title and the links.

3(c): in main_list_layout.xml file use the textviews for the index and the titles.

3(d): in activity_rss_loader.xml file use the Listview and the progress bars

Step 4: create the following java files and write the code in it

4(a): in MainActivity.java create an ArrayList for the courses and the rss-path .add the courses to the ListView and use the MainListAdapter object to handle the events of the listView with the arguments as the rss_list_layout and the course class.

4(b): Rssloader.java write the code to load the feed from the links to display it in the screen using the progressbar of the Activity_rss_loader

4(c): MainListAdapter.java get the index and the titles and display it on to the screen using the main_list_layout.

4(d): MainItemClickListener.java implements the itemClickListener where it gets the various links from the corresponding rss_path

4(e): in RssReader.java it creates the available RSS item list from the RSS url and parses it

4(f): RssListAdapter.java , RssItemClickListener.java, RssItem.java and RssParseHandler.java files are used to manage the rss link and when clicked on a particular item the corresponding feeds are retrieved and displayed to the user.

Step 5: in the AndroidManifest.xml file use the uses-permission xml tag to give the internet permission.

Step 6: Launch the application using the emulator then go to tools->android->android device manager->emulator control. Send the location information to the emulator.

Step 7: Stop the program.

PROGRAM CODING:

//RSS FEED

Main Activity.java

```
package com.example.admin.rss;
```

```
import android.app.Activity;
import android.os.Bundle;
import android.widget.ListView;
import java.util.ArrayList;
```

```
public class MainActivity extends Activity {
    ArrayList<Course> courses = new ArrayList<Course>();
    public static String rss_path="http://ocw.mit.edu/rss/new/";
    @Override
    protected void onCreate (Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        courses.add(new Course("Computer science engineering","mit-newcourses-6.xml"));
        courses.add(new Course("Civil engineering", "mit-newcourses-1.xml"));
        courses.add(new Course("Mechanical engineering", "mit - newcourses-2.xml"));
        courses.add(new Course("Architecture", "mit - newcourses - 4.xml"));
        ListView mainList = (ListView) findViewById(R.id.listView);
        MainListAdapter adapter = new MainListAdapter(this, R.layout.main_list_layout,
courses);
        mainList.setAdapter(adapter);
        MainItemClickListener listener = new MainItemClickListener(courses, this);
        mainList.setOnItemClickListener(listener);
    }
}
```

RssLoder.java:

```
package com.example.admin.rss;
```

```
import android.app.Activity;
import android.os.AsyncTask;
import android.os.Bundle;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
```

```

import android.view.View;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.Toast;
import java.util.ArrayList;
public class RssLoader extends AppCompatActivity {
    Activity local;
    ProgressBar progressBar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_rss_loader);
        local = this;
        progressBar = (ProgressBar) findViewById(R.id.progress);
        Intent i = getIntent();
        //gets the root intent
        String title = i.getStringExtra("title");
        //gets the String supplied with the tag title
        String url = i.getStringExtra("link");
        //gets the String supplied with the tag link
        getRssData getRssData = new getRssData();
        getRssData.execute(url);
        //actionbar
        getSupportActionBar().setTitle(title);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    }

    private class getRssData extends AsyncTask<String, Void, ArrayList<RssItem>> {
        @Override
        protected ArrayList<RssItem> doInBackground(String... params) {
            try {
                RssReader rssReader = new RssReader(params[0]);
                //create rss reader
                return rssReader.getItems();
                //returns items parsed from rss file
            } catch (Exception e) {
                e.printStackTrace();
            }
            return null;
        }
    }

```

```

    }

    @Override
    protected void onPostExecute(ArrayList<RssItem> rssItems) {
        if (rssItems != null) {
            ListView rssList = (ListView) findViewById(R.id.lstRssFeed);
            RssListAdapter adapter = new RssListAdapter(local, R.layout.rss_list_layout,
rssItems);
            rssList.setAdapter(adapter);
            rssList.setOnItemClickListener(new RssItemClickListener(local, rssItems));
            Toast.makeText(local, "Feed Loaded", Toast.LENGTH_SHORT).show();
        } else
            Toast.makeText(local, "Could not load feed. Check your network connections.",
Toast.LENGTH_SHORT).show();
        progressBar.setVisibility(View.INVISIBLE);
    }

    @Override
    protected void onPreExecute() {
        Toast.makeText(local, "Loading Feed", Toast.LENGTH_LONG).show();
    }
}

```

MainListAdapter.java:

```

package com.example.admin.rss;

import android.app.Activity;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.TextView;
import java.util.ArrayList;

public class MainListAdapter extends ArrayAdapter<Course> {
    int layoutResourceId;

    Context context;

```

```

ArrayList<Course> list;
public MainListAdapter(Context context, int resource, ArrayList<Course> objects) {
    super(context, resource, objects);
    this.layoutResourceId = resource;
    this.context = context;
    this.list = objects;
}
@Override
public View getView(int position, View convertView, ViewGroup parent) {
    View row = convertView;
    MainListHolder holder = null;
    if (row == null) {
        LayoutInflater inflater = ((Activity) context).getLayoutInflater();
        row = inflater.inflate(layoutResourceId, parent, false);
        holder = new MainListHolder();
        holder.indexTxt = (TextView) row.findViewById(R.id.txtIndex);
        holder.titleTxt = (TextView) row.findViewById(R.id.txtTitle);
        row.setTag(holder);
    }
    else
        holder = (MainListHolder) row.getTag();
        Course course = list.get(position);
        holder.indexTxt.setText(String.valueOf(position + 1));
        holder.titleTxt.setText(course.getTitle());
        return row;

    }
    static class MainListHolder
    {
        TextView indexTxt, titleTxt;
    }
}

```

MainItemClickListener.java:

```

package com.example.admin.rss;

import android.content.Intent;
import android.widget.AdapterView;
import android.view.View;
import android.app.Activity;
import java.util.ArrayList;

```

```

public class MainListItemClickListener implements AdapterView.OnItemClickListener{
    ArrayList<Course> list;
    Activity activity;
    public MainListItemClickListener(ArrayList<Course> list,Activity activity)
    {
        this.list=list;
        this.activity=activity;
    }
    @Override
    public void onItemClick(AdapterView<?> parent,View view,int position,long id)
    {
        Course course=list.get(position);
        Intent intent=new Intent(activity,RssLoader.class);
        intent.putExtra("title",course.getTitle());
        intent.putExtra("link",MainActivity.rss_path+course.getUrl());

        activity.startActivity(intent);

    }
}

```

RssReader.java:

```

package com.example.admin.rss;

import java.util.ArrayList;
import javax.xml.parsers.SAXParser;
import javax.xml.parsers.SAXParserFactory;

public class RssReader {
    private String rssUrl;
    public RssReader(String rssUrl) {
        this.rssUrl = rssUrl;
    }
    public ArrayList<RssItem> getItems() throws Exception {
        SAXParserFactory factory = SAXParserFactory.newInstance();
        SAXParser parser = factory.newSAXParser();
        RssParseHandler handler = new RssParseHandler();
        parser.parse(rssUrl, handler);
        return handler.getRssItems();
    }
}

```

RssListAdapter.java:

```
package com.example.admin.rss;

import android.app.Activity;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.TextView;
import org.w3c.dom.Text;
import java.util.ArrayList;

public class RssListAdapter extends ArrayAdapter<RssItem> {
    int layoutResourceId;
    Context context;
    ArrayList<RssItem> list;
    public RssListAdapter(Context context, int resource, ArrayList<RssItem> objects) {
        super(context, resource, objects);
        this.layoutResourceId = resource;
        this.context = context;
        this.list = objects;
    }
    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        View row = convertView;
        RssItemHolder holder = null;
        if (row == null) {
            LayoutInflater inflater = ((Activity) context).getLayoutInflater();
            row = inflater.inflate(layoutResourceId, parent, false);
            holder = new RssItemHolder();
            holder.titleTxt = (TextView)row.findViewById(R.id.txtTitle);
            holder.linkTxt = (TextView)row.findViewById(R.id.txtLink);
            row.setTag(holder);
        }
        else
            holder = (RssItemHolder)row.getTag();
        RssItem item = list.get(position);
        holder.titleTxt.setText(item.getTitle());
        holder.linkTxt.setText(item.getLink());
        return row;
    }
    static class RssItemHolder {
        TextView titleTxt, linkTxt;
    }
}
```

RssItemClickListener.java:

```
package com.example.admin.rss;

import android.app.Activity;
import android.content.Intent;
import android.widget.AdapterView;
import java.util.ArrayList;
import android.net.Uri;
import android.view.View;

public class RssItemClickListener implements AdapterView.OnItemClickListener {
    Activity activity;
    ArrayList<RssItem> list;
    public RssItemClickListener(Activity activity, ArrayList<RssItem> list) {
        this.activity = activity;
        this.list = list;
    }
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
        Intent intent = new Intent(Intent.ACTION_VIEW);
        intent.setData(Uri.parse(list.get(position).getLink()));
        activity.startActivity(intent);
    }
}
```

RssItem.java:

```
package com.example.admin.rss;

public class RssItem {
    private String title, link;
    public String getTitle(){
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
    public String getLink(){
        return link;
    }
    public void setLink(String link)
    {
        this.link=link;}
}
```


RssParseHandler.java:

```
package com.example.admin.rss;

import org.xml.sax.Attributes;
import org.xml.sax.SAXException;
import org.xml.sax.helpers.DefaultHandler;
import java.util.ArrayList;

public class RssParseHandler extends DefaultHandler {
    private ArrayList<RssItem> rssItems;
    //list to store rss items parsed
    private RssItem currentItem;
    //stores the currently parsing RssItem
    private boolean parsingTitle, parsingLink;
    //parsing title/link indicators
    public RssParseHandler() { rssItems = new ArrayList<>();
    }
    public ArrayList<RssItem> getRssItems() {
        return rssItems;
    }
    @Override
    public void startElement(String uri, String localName, String qName, Attributes attributes)
throws SAXException {
        if ("item".equals(qName)){
            currentItem = new RssItem();
        }
        //create a new RssItem to store data

        else if ("title".equals(qName)){
            parsingTitle = true;
        }
        else if ("link".equals(qName)){
            parsingLink = true;
        }
        //sets the current item to rss item when parsing item
        //sets appropriate indicators to true when opening tags are processed
    }
    @Override
    public void endElement(String uri, String localName, String qName) throws SAXException
    {
        if ("item".equals(qName)){
            rssItems.add(currentItem);
            currentItem = null;
        }
        else if ("title".equals(qName)){
            parsingTitle = false;
        }
    }
}
```

```

    }
    else if ("link".equals(qName)){
        parsingLink = false;
    }
    //adds the current item to list
    //sets appropriate indicators to false when closing tags are processed
    }
    @Override
    public void characters(char[] ch, int start, int length) throws SAXException {
        if (parsingTitle){
            if (currentItem != null) {
                currentItem.setTitle(new String(ch, start, length));
            }
            //sets title of the current item
        }
        else if (parsingLink){
            if (currentItem != null) {
                currentItem.setLink(new String(ch, start, length));
            }
        }
    }
}

```

Course.java:

```

package com.example.admin.rss;

public class Course {
    private String title,url;
    public Course(String title,String url)
    {
        this.title=title;
        this.url=url;
    }
    public String getTitle()
    {
        return title;
    }
    public String getUrl()
    {
        return url;
    }
}

```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.admin.rss">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".RssLoader" android:parentActivityName=".MainActivity"/>
    </application>
</manifest>
```

activity rss loader.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ListView
        android:layout_width="339dp"
        android:layout_height="260dp"
        android:id="@+id/1stRssFeed"
        android:layout_gravity="center_horizontal" />

    <ProgressBar
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="246dp"
        android:layout_height="match_parent"
        android:id="@+id/progress"
        android:indeterminate="false" />
</LinearLayout>
```

main list layout.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="1">

    <TextView
        android:layout_width="176dp"
        android:layout_height="wrap_content"
        android:text="INDEX"
        android:id="@+id/txtIndex"
        android:layout_gravity="center_horizontal" />

    <TextView
        android:layout_width="138dp"
        android:layout_height="0dp"
        android:text="TITLE"
        android:id="@+id/txtTitle"
        android:layout_gravity="center_horizontal"
        android:layout_weight="0.09" />
</LinearLayout>
```

rss list layout.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TITLE"
        android:id="@+id/txtTitle" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="&quot;http://www.example.com/post1&quot;"
        android:id="@+id/txtLink" />
</LinearLayout>
```

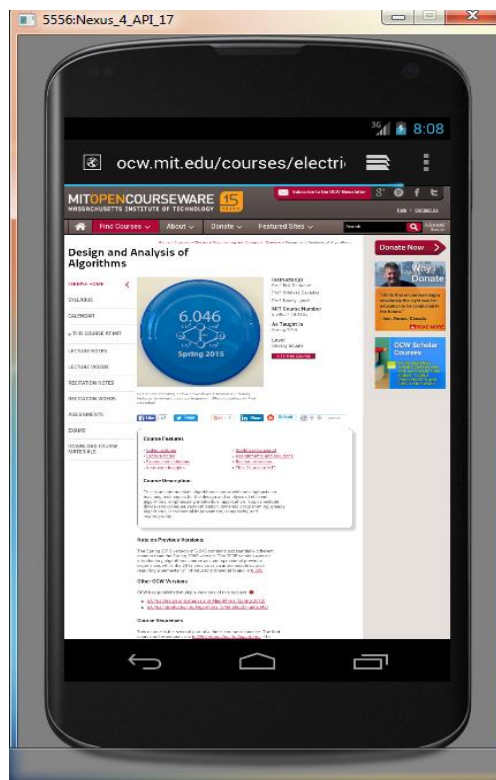
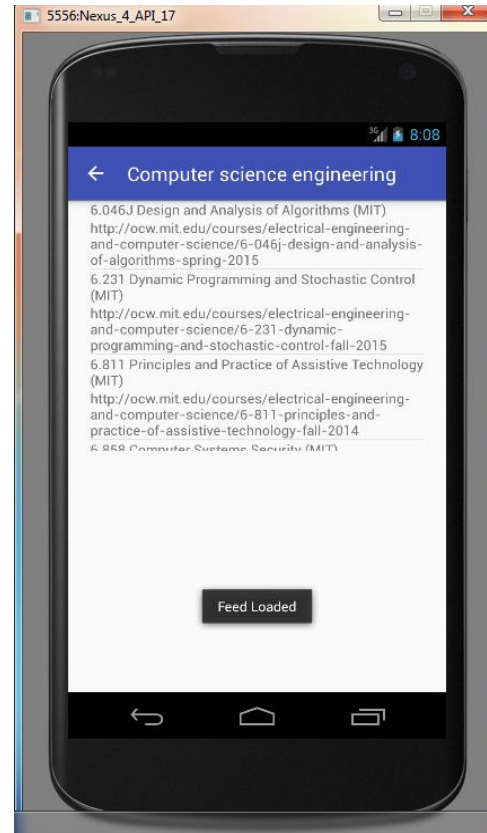
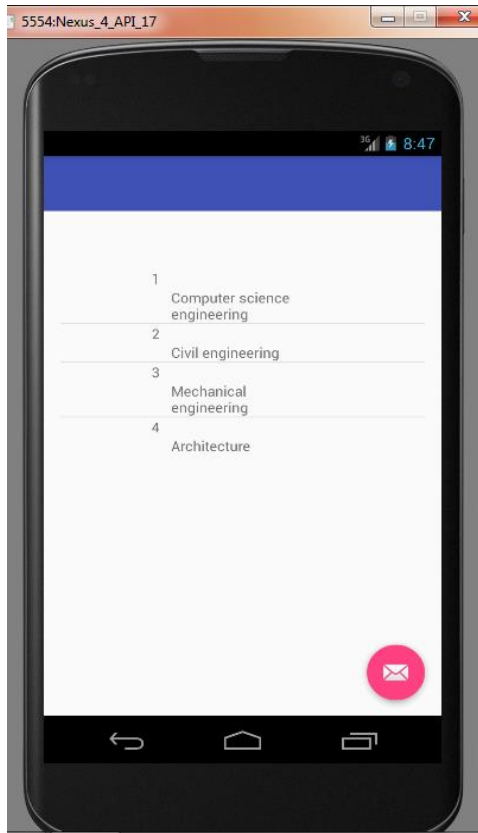
Activity main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context="com.example.admin.rss.MainActivity"
    tools:showIn="@layout/activity_main">

    <ListView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/listView"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="43dp" />

</RelativeLayout>
```

OUTPUT:



RESULT

Thus, the mobile application that makes use of RSS feed has been developed successfully.