**package** com.project.assistantofcourse;  
**import** android.content.Intent;  
**import** android.os.AsyncTask;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**import** java.util.ArrayList;  
**import** java.util.Arrays;  
**import** java.util.List;  
**public class** AnswerActivity **extends** AppCompatActivity {  
 **private** Button **btn\_answer**;  
 **private** EditText **et\_answer**;  
 **private** String **question\_id**;  
 **private** ListView **listview**;  
 **private** List<String> **data**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_answer***);  
 *//初始化 UI组件* **btn\_answer**= (Button) findViewById(R.id.***btn\_answer***);  
 **et\_answer**=(EditText)findViewById(R.id.***et\_answer***);  
 **listview** = (ListView)findViewById(R.id.***list\_send***);  
 TextView tv\_question = (TextView) findViewById(R.id.***tv\_question***);  
 Intent intent=getIntent();  
 **question\_id**=intent.getExtras().getString(**"question\_id"**);  
 String title = intent.getExtras().getString(**"title"**);  
 tv\_question.setText(**" "**+ title);  
 **new** User().setCurAnswer(**""**);  
 **new** AnswerListTask().execute(**"http://123.206.106.46:8080/Server/user/answerList.do"**);  
 *//("http://192.168.191.1:8080/Server/user/answerList.do");* **et\_answer**.setText(**""**);  
 **btn\_answer**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 String content=**et\_answer**.getText().toString();  
 **if**(content.equals(**""**)){  
 Toast.*makeText*(AnswerActivity.**this**,**"回答内容不能为空!"**,Toast.***LENGTH\_SHORT***).show();  
 }**else** {  
 **btn\_answer**.setEnabled(**false**);  
 **new** AnswerTask(**question\_id**,content).execute(**"http://123.206.106.46:8080/Server/user/answer.do"**);  
 **new** AnswerListTask().execute(**"http://123.206.106.46:8080/Server/user/answerList.do"**);  
 }  
 }  
 });  
 }  
 **private void** getData(String [] a,**int** b) {  
 **data** = **new** ArrayList<>();  
 **data**.addAll(Arrays.*asList*(a).subList(0, b));*//取前b个元素* }  
 **private class** AnswerListTask **extends** AsyncTask<String, Void, String> {  
 **private** String **id**;  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 **id**=**question\_id**;  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"id="** + **id** );  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 TextView tv= (TextView) findViewById(R.id.***tip3***);  
 String [] s=response.split(**"-"**);  
 **if**(s.**length**>0&&!response.equals(**""**)){  
 tv.setVisibility(View.***GONE***);  
 **if**(!**new** User().getCurAnswer().equals(response)) {  
 getData(s,s.**length**);  
 ArrayAdapter<String> adapter= **new** ArrayAdapter<>(AnswerActivity.**this**, android.R.layout.***simple\_list\_item\_1***, **data**);  
 **listview**.setAdapter(adapter);  
 **new** User().setCurAnswer(response);  
 }  
 }  
 **else**{  
 tv.setVisibility(View.***VISIBLE***);  
 }  
 }  
 }  
 **private class** AnswerTask **extends** AsyncTask<String, Void, String> {  
 **private** String **id**;  
 **private** String **content**;  
 User **user**=**new** User();  
 AnswerTask(String id,String content){  
 **this**.**id**=id;  
 **this**.**content**=content;  
 }  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"id="** + **id**+**"&content="**+**content**+**"&presenter\_id="**+**user**.getId() );  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 **switch**(response){  
 **case "0"**:  
 Toast.*makeText*(AnswerActivity.**this**,**"成功回答!"**,Toast.***LENGTH\_SHORT***).show();  
 **et\_answer**.setText(**""**);  
 **btn\_answer**.setEnabled(**true**);  
 **break**;  
 **default**:  
 Toast.*makeText*(AnswerActivity.**this**,**"回答失败!"**,Toast.***LENGTH\_SHORT***).show();  
 **btn\_answer**.setEnabled(**true**);  
 **break**;  
 }  
 }  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.app.Service;  
**import** android.content.Intent;  
**import** android.os.Environment;  
**import** android.os.Handler;  
**import** android.os.IBinder;  
**import** android.os.Message;  
**import** android.widget.Toast;  
**import** java.io.BufferedWriter;  
**import** java.io.DataInputStream;  
**import** java.io.File;  
**import** java.io.FileOutputStream;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**public class** DownloadService **extends** Service {  
 **private static final int *DOWNLOAD\_SUCCESSFULLY*** = 1;  
 **private** String **filename**;  
 **private** Handler **handler** = **new** Handler() {  
 @Override  
 **public void** handleMessage(Message msg) {  
 **switch** (msg.**what**) {  
 **case *DOWNLOAD\_SUCCESSFULLY***:  
 Toast.*makeText*(getApplicationContext(), msg.**obj** + **"下载完成!"**, Toast.***LENGTH\_SHORT***).show();  
 **break**;  
 **default**:  
 Toast.*makeText*(getApplicationContext(), **"下载出现异常!"**, Toast.***LENGTH\_SHORT***).show();  
 **break**;  
 }  
 **super**.handleMessage(msg);  
 }  
 };  
 @Override  
 **public int** onStartCommand(Intent intent, **int** flags, **int** startId) {  
 **filename** = intent.getExtras().getString(**"filename"**);  
 **new** Thread(**new** Runnable() {  
 @Override  
 **public void** run() {  
 **try** {  
 Message message = **new** Message();  
 message.**obj**=**filename**;  
 message.**what** = ***DOWNLOAD\_SUCCESSFULLY***;  
 **byte**[] buff;  
 DataInputStream dis;  
 BufferedWriter bw;  
 URL url = **new** URL(**"http://123.206.106.46:8080/Server/user/download.do"**);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"path="** + **filename**);  
 bw.flush();  
 bw.close();  
 String dirname = Environment.*getExternalStorageDirectory*().getAbsolutePath() + File.***separator*** + **"courseware"**;  
 FileOutputStream fos = **new** FileOutputStream(**new** File(dirname + File.***separator*** + **filename**));  
 dis = **new** DataInputStream(connection.getInputStream());  
 buff = **new byte**[1024];  
 **int** a;  
 **while** ((a = dis.read(buff, 0, buff.**length**)) != -1) {  
 fos.write(buff, 0, a);  
 }  
 fos.close();  
 dis.close();  
 connection.disconnect();  
 **handler**.sendMessage(message);  
 stopSelf();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }).start();  
 **return super**.onStartCommand(intent, flags, startId);  
 }  
 @Override  
 **public** IBinder onBind(Intent intent) {  
 **throw new** UnsupportedOperationException(**"Not yet implemented"**);  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.AsyncTask;  
**import** android.os.Bundle;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v7.app.AlertDialog;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.view.inputmethod.InputMethodManager;  
**import** android.widget.AdapterView;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**import** java.util.ArrayList;  
**import** java.util.Arrays;  
**import** java.util.List;  
**public class** FragmentDiscuss **extends** Fragment {  
 **private** ListView **listview**;  
 **private** List<String> **data**;  
 **private** View **view**;  
 @Override  
 **public void** onDestroyView() {  
 **super**.onDestroyView();  
 }  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle  
 savedInstanceState) {  
 **view** = inflater.inflate(R.layout.***fragment\_item2\_1***, **null**);  
 Button btn\_question = (Button) **view**.findViewById(R.id.***question***);  
 **listview** = (ListView) **view**.findViewById(R.id.***listview2\_1***);  
 *//设定列表项的选择模式为单选* **new** QueryQuestionTask().execute(**"http://123.206.106.46:8080/Server/user/queryQuestion.do"**);  
 **listview**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 @Override  
 **public void** onItemClick(AdapterView<?> parent, View view, **int** position, **long** id) {  
 *//列表项单击事件* Intent intent=**new** Intent(getContext(),AnswerActivity.**class**);  
 intent.putExtra(**"question\_id"**,**data**.get(2\*position));  
 intent.putExtra(**"title"**,**data**.get(2\*position+1));  
 startActivity(intent);  
 }  
 });  
 *//提问* btn\_question.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **final** EditText editText=**new** EditText(getContext());  
 editText.setText(**""**);  
 editText.setBackgroundResource(R.drawable.***editbox\_background\_normal***);  
 **new** AlertDialog.Builder(getContext())  
 .setTitle(**"请输入您的问题"**)  
 .setView(editText)  
 .setPositiveButton(**"提交"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 String title=editText.getText().toString();  
 **if**(!title.equals(**""**)){  
 **new** QuestionTask(title).execute(**"http://123.206.106.46:8080/Server/user/question.do"**);  
 **new** QueryQuestionTask().execute(**"http://123.206.106.46:8080/Server/user/queryQuestion.do"**);  
 }  
 **else**{*//问题输入不能为空* **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"问题不能为空!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 InputMethodManager inputMethodManager= (InputMethodManager) getContext().getSystemService(Context.***INPUT\_METHOD\_SERVICE***);  
 **if**(inputMethodManager!=**null**) {  
 inputMethodManager.hideSoftInputFromWindow(editText.getWindowToken(),0);  
 dialog.dismiss();  
 }  
 }  
 })  
 .setNegativeButton(**"取消"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 InputMethodManager inputMethodManager= (InputMethodManager) getContext().getSystemService(Context.***INPUT\_METHOD\_SERVICE***);  
 **if**(inputMethodManager!=**null**) {  
 inputMethodManager.hideSoftInputFromWindow(editText.getWindowToken(),0);  
 dialog.dismiss();  
 }  
 }  
 })  
 .show();  
 }  
 });  
 **return view**;  
 }  
 **private void** getData(String [] a, **int** b) {  
 **data** = **new** ArrayList<>();  
 **data**.addAll(Arrays.*asList*(a).subList(0, b));*//取前b个元素* }  
 **private class** QuestionTask **extends** AsyncTask<String, Void, String> {  
 **private** String **title**;  
 **private** User **user**=**new** User();  
 QuestionTask(String title){  
 **this**.**title**=title;  
 }  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"title="**+**title**+**"&presenter\_id="**+**user**.getId());  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 **switch**(response) {  
 **case "0"**:  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"问题成功发起!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **case "-1"**:  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"数据库插入失败"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **default**:  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"问题发起失败!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 }  
 }  
 }  
 **private class** QueryQuestionTask **extends** AsyncTask<String, Void, String> {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 TextView tv= (TextView) **view**.findViewById(R.id.***tip2***);  
 String [] s=response.split(**"-"**);  
 **if**(s.**length**>0&&!response.equals(**""**)){  
 tv.setVisibility(View.***GONE***);  
 getData(s,s.**length**);  
 List<String> data1= **new** ArrayList<>();  
 **for**(**int** i=1;i<**data**.size();i+=2){  
 data1.add(**data**.get(i));  
 }  
 **if**(getContext()!=**null**) {  
 ArrayAdapter<String> adapter = **new** ArrayAdapter<>(getContext(), android.R.layout.***simple\_list\_item\_1***, data1);  
 **listview**.setAdapter(adapter);  
 }  
 }  
 **else**{  
 **if**(response.equals(**""**)){  
 getData(s,0);  
 ArrayAdapter<String> adapter= **new** ArrayAdapter<>(getContext(), android.R.layout.***simple\_list\_item\_1***, **data**);  
 **listview**.setAdapter(adapter);  
 }  
 tv.setVisibility(View.***VISIBLE***);  
 }  
 }  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.AsyncTask;  
**import** android.os.Bundle;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v7.app.AlertDialog;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.view.inputmethod.InputMethodManager;  
**import** android.widget.AdapterView;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ListView;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**import** java.util.ArrayList;  
**import** java.util.List;  
**public class** FragmentMine **extends** Fragment {  
 **private** ListView **listview**;  
 **private** List<String> **data**;  
 **private** String **name**;  
 @Override  
 **public void** onDestroyView() {  
 **super**.onDestroyView();  
 }  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
 *//Toast.makeText(getContext(),"4\_create",Toast.LENGTH\_SHORT).show();* View view = inflater.inflate(R.layout.***fragment\_item4\_1***, **null**);  
 **name**=**""**;  
 Button btn\_logout = (Button) view.findViewById(R.id.***btn\_logout***);  
 **listview**= (ListView) view.findViewById(R.id.***mine***);  
 **data**= **new** ArrayList<>();  
 **new** GetNameTask().execute(**"http://123.206.106.46:8080/Server/user/getName.do"**);  
 *//列表单击事件* **listview**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 @Override  
 **public void** onItemClick(AdapterView<?> parent, **final** View view, **int** position, **long** id) {  
 **if**(position==1&&getContext()!=**null**){*//j检查密码* **final** EditText editText=**new** EditText(getContext());  
 editText.setText(**""**);  
 editText.setBackgroundResource(R.drawable.***editbox\_background\_normal***);  
 **new** AlertDialog.Builder(getContext())  
 .setTitle(**"请输入原密码"**)  
 .setView(editText)  
 .setPositiveButton(**"确定"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 **new** CheckPasswordTask(editText.getText().toString()).execute(**"http://123.206.106.46:8080/Server/user/checkPassword.do"**);  
 InputMethodManager inputMethodManager= (InputMethodManager) getContext().getSystemService(Context.***INPUT\_METHOD\_SERVICE***);  
 **if**(inputMethodManager!=**null**) {  
 inputMethodManager.hideSoftInputFromWindow(editText.getWindowToken(),0);  
 dialog.dismiss();  
 }  
 }  
 })  
 .setNegativeButton(**"取消"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 InputMethodManager inputMethodManager= (InputMethodManager) getContext().getSystemService(Context.***INPUT\_METHOD\_SERVICE***);  
 **if**(inputMethodManager!=**null**) {  
 inputMethodManager.hideSoftInputFromWindow(editText.getWindowToken(),0);  
 dialog.dismiss();  
 }  
 }  
 })  
 .show();  
 }  
 }  
 });  
 *//登出* btn\_logout.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **if**(getContext()!=**null**) {  
 **new** AlertDialog.Builder(getContext())  
 .setTitle(**"退出当前账号?"**)  
 .setPositiveButton(**"确认退出"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 Intent intent = **new** Intent(getContext(), MainActivity.**class**);  
 intent.setFlags(Intent.***FLAG\_ACTIVITY\_CLEAR\_TOP*** | Intent.***FLAG\_ACTIVITY\_SINGLE\_TOP***);  
 onDestroy();  
 startActivity(intent);  
 }  
 })  
 .setNegativeButton(**"取消"**, **null**)  
 .show();  
 }  
 }  
 });  
 **return** view;  
 }  
 **private class** GetNameTask **extends** AsyncTask<String, Void, String> {  
 **private** User **user**=**new** User();  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"id="**+**user**.getId());  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 **if**(!response.equals(**""**)&&getContext()!=**null**) {  
 **name**=response;  
 getData();  
 ArrayAdapter<String> adapter = **new** ArrayAdapter<>(getContext(), android.R.layout.***simple\_list\_item\_1***, **data**);  
 **listview**.setAdapter(adapter);  
 }  
 }  
 }  
 **public void** getData(){  
 **data**.add(**name**);  
 **data**.add(**"修改密码"**);  
 }  
 **private class** CheckPasswordTask **extends** AsyncTask<String, Void, String> {  
 **private** User **user**=**new** User();  
 **private** String **curPassword**;  
 CheckPasswordTask(String curPassword){  
 **this**.**curPassword**=curPassword;  
 }  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"id="**+**user**.getId()+**"&password="**+**curPassword**);  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 **if**(response.equals(**"0"**)&&getContext()!=**null**) {  
 **final** EditText editText0=**new** EditText(getContext());  
 editText0.setText(**""**);  
 editText0.setBackgroundResource(R.drawable.***editbox\_background\_normal***);  
 **if**(getContext()!=**null**) {  
 **new** AlertDialog.Builder(getContext())  
 .setTitle(**"请输入新密码"**)  
 .setView(editText0)  
 .setPositiveButton(**"确定"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 **if** (editText0.getText().toString().length() <= 20 && editText0.getText().toString().length() >= 1) {  
 **new** UpdatePasswordTask(editText0.getText().toString()).execute(**"http://123.206.106.46:8080/Server/user/updatePassword.do"**);  
 } **else** {  
 InputMethodManager inputMethodManager = (InputMethodManager) getContext().getSystemService(Context.***INPUT\_METHOD\_SERVICE***);  
 **if** (inputMethodManager != **null**) {  
 inputMethodManager.hideSoftInputFromWindow(editText0.getWindowToken(), 0);  
 dialog.dismiss();  
 }  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"密码应为1~20位字符!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }  
 })  
 .setNegativeButton(**"取消"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 InputMethodManager inputMethodManager = (InputMethodManager) getContext().getSystemService(Context.***INPUT\_METHOD\_SERVICE***);  
 **if** (inputMethodManager != **null**) {  
 inputMethodManager.hideSoftInputFromWindow(editText0.getWindowToken(), 0);  
 dialog.dismiss();  
 }  
 }  
 })  
 .show();  
 }  
 }  
 **else** {  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"原密码输入错误!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }  
 }  
 **private class** UpdatePasswordTask **extends** AsyncTask<String, Void, String> {  
 **private** User **user**=**new** User();  
 **private** String **newPassword**;  
 UpdatePasswordTask(String curPassword){  
 **this**.**newPassword**=curPassword;  
 }  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"id="**+**user**.getId()+**"&password="**+**newPassword**);  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 **if**(response.equals(**"0"**)) {  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"密码修改成功!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 **else** {  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"数据异常,密码修改失败!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.AsyncTask;  
**import** android.os.Bundle;  
**import** android.os.Environment;  
**import** android.support.v4.app.Fragment;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.AdapterView;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.File;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**import** java.util.ArrayList;  
**import** java.util.List;  
**public class** FragmentResource **extends** Fragment {  
 **private** ListView **listview**;  
 **private** List<String> **data**;  
 **private** View **view**;  
 @Override  
 **public void** onDestroyView() {  
 **super**.onDestroyView();  
 }  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
 **view** = inflater.inflate(R.layout.***fragment\_item3\_1***, **null**);  
 **listview** = (ListView) **view**.findViewById(R.id.***listview3\_1***);  
 *//设定列表项的选择模式为单选* **new** QueryResourceTask().execute(**"http://123.206.106.46:8080/Server/user/queryResource.do"**);  
 **listview**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 @Override  
 **public void** onItemClick(AdapterView<?> parent, View view, **int** position, **long** id) {  
 *//列表项单击事件* **try** {  
 **if** (Environment.*getExternalStorageState*().equals(Environment.***MEDIA\_MOUNTED***)) {  
 String dirname = Environment.*getExternalStorageDirectory*().getAbsolutePath() + File.***separator*** + **"courseware"**;  
 String filename = dirname + File.***separator*** + **data**.get(position);  
 File file = **new** File(filename);  
 File dir = **new** File(dirname);  
 **if** (!dir.exists()) {*//目录不存在时，创建目录* dir.mkdirs();  
 }  
 **if** (!file.exists()) {*//文件不存在，创建文件* file.createNewFile();  
 Toast.*makeText*(getContext(), **data**.get(position) + **"开始下载..."**, Toast.***LENGTH\_SHORT***).show();  
 Intent intentService = **new** Intent(getContext(), DownloadService.**class**);  
 intentService.putExtra(**"filename"**, **data**.get(position));  
 getContext().startService(intentService);  
 } **else** {*//文件已存在，打开文件* Intent intent = **new** Intent();  
 intent.addFlags(Intent.***FLAG\_ACTIVITY\_NEW\_TASK***);  
 intent.setAction(Intent.***ACTION\_VIEW***);  
 String type = Utils.*getMIMEType*(file);  
 intent.setDataAndType(Uri.*fromFile*(file), type);  
 startActivity(intent);  
 }  
 } **else** {  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"SD卡不存在或者不可读写"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 } **catch** (Exception e) {  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"文件读写异常!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }  
 });  
 **return view**;  
 }  
 **private void** getData(String[] a, **int** b) {  
 **data** = **new** ArrayList<String>();  
 **for** (**int** i = 0; i < b; i++) {  
 **data**.add(a[i]);  
 }  
 }  
 **private class** QueryResourceTask **extends** AsyncTask<String, Void, String> {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(30000);  
 connection.setReadTimeout(30000);  
 connection.connect();  
 bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.flush();  
 bw.close();  
 br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String response) {  
 TextView tv = (TextView) **view**.findViewById(R.id.***tip***);  
 String[] s = response.split(**" "**);  
 **if** (s.**length** > 0 && !response.equals(**""**)) {  
 tv.setVisibility(View.***GONE***);  
 getData(s, s.**length**);  
 **if** (getContext() != **null**) {  
 ArrayAdapter<String> adapter = **new** ArrayAdapter<>(getContext(), android.R.layout.***simple\_list\_item\_1***, **data**);  
 **listview**.setAdapter(adapter);  
 }  
 } **else** {  
 **if** (response.equals(**""**)) {  
 getData(s, 0);  
 ArrayAdapter<String> adapter = **new** ArrayAdapter<>(getContext(), android.R.layout.***simple\_list\_item\_1***, **data**);  
 **listview**.setAdapter(adapter);  
 }  
 tv.setVisibility(View.***VISIBLE***);  
 }  
 }  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.graphics.Typeface;  
**import** android.os.AsyncTask;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.os.Message;  
**import** android.support.v4.app.Fragment;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.IOException;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**public class** FragmentVote **extends** Fragment {  
 **private** TextView **tv\_getname** = **null**;  
 **private** TextView **result\_tv** = **null**;  
 **private** Button **vote\_con** = **null**;  
 **private** Button **vote\_pro** = **null**;  
 **private** View.OnClickListener **onClickListener1** = **null**;  
 **private** View.OnClickListener **onClickListener2** = **null**;  
 *//uiHandler在主线程中创建，所以自动绑定主线程* **private** Handler **uiHandler** = **new** Handler() {  
 @Override  
 **public void** handleMessage(Message msg) {  
 **switch** (msg.**what**) {  
 **case** 1:  
 Bundle bundle = msg.getData();  
 String response = bundle.getString(**"response"**);  
 **assert** response != **null**;  
 **if** (response.equals(**""**)) {  
 **tv\_getname**.setVisibility(View.***INVISIBLE***);  
 **result\_tv**.setVisibility(View.***INVISIBLE***);  
 } **else** {  
 **if** (**tv\_getname**.getText() != **" "** + response + **"老师正在向大家对本次讲课的理解程度发起投票..."**) {  
 **tv\_getname**.setText(**" "** + response + **"老师正在向大家对本次讲课的理解程度发起投票..."**);  
 }  
 **tv\_getname**.setVisibility(View.***VISIBLE***);  
 **vote\_con**.setOnClickListener(**onClickListener2**);*//未理解* **vote\_pro**.setOnClickListener(**onClickListener1**);*//已理解* }  
 **break**;  
 }  
 }  
 };  
 @Override  
 **public** View onCreateView(**final** LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_item1\_2***, **null**);  
 TextView vote\_tv = (TextView) view.findViewById(R.id.***vote\_tv2***);  
 Typeface tf = Typeface.*createFromAsset*(getContext().getAssets(), **"fonts/DancingScript-Bold.ttf"**);  
 vote\_tv.setTypeface(tf);  
 **tv\_getname** = (TextView) view.findViewById(R.id.***tv\_getname***);  
 **tv\_getname**.setText(**""**);  
 **tv\_getname**.setVisibility(View.***INVISIBLE***);  
 **result\_tv** = (TextView) view.findViewById(R.id.***result\_tv***);  
 **result\_tv**.setVisibility(View.***INVISIBLE***);  
 **vote\_con** = (Button) view.findViewById(R.id.***vote\_con***);  
 **vote\_pro** = (Button) view.findViewById(R.id.***vote\_pro***);  
 *// 暂无投票发起* View.OnClickListener onClickListener0 = **new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"暂无投票发起!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 };  
 **vote\_con**.setOnClickListener(onClickListener0);  
 **vote\_pro**.setOnClickListener(onClickListener0);  
 *//已理解* **onClickListener1** = **new** View.OnClickListener() {  
 BufferedReader **br** = **null**;  
 BufferedWriter **bw** = **null**;  
 User **user** = **new** User();  
 @Override  
 **public void** onClick(View v) {  
 **new** AsyncTask<String, Void, String>() {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 *//请求参数* **bw** = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 **bw**.write(**"id="** + **user**.getId() + **"&pro=1"**);  
 **bw**.flush();  
 **bw**.close();  
 *//获取响应* **br** = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = **br**.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (**br** != **null**) {  
 **br**.close();  
 }  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String s) {  
 **int** response = Integer.*parseInt*(s);  
 **switch** (response) {  
 **case** 0:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"暂无投票发起,请稍后再试!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **case** 1:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"投票成功,感谢您的参与!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **result\_tv**.setText(**"您的投票结果是:已理解"**);  
 **result\_tv**.setVisibility(View.***VISIBLE***);  
 **break**;  
 **case** 2:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"您已经参加过本次投票!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **default**:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"投票失败!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 }  
 **super**.onPostExecute(s);  
 }  
 }.execute(**"http://123.206.106.46:8080/Server/user/vote.do"**);  
 }  
 };  
 *//未理解* **onClickListener2** = **new** View.OnClickListener() {  
 BufferedReader **br** = **null**;  
 BufferedWriter **bw** = **null**;  
 User **user** = **new** User();  
 @Override  
 **public void** onClick(View v) {  
 **new** AsyncTask<String, Void, String>() {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 *//请求参数* **bw** = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 **bw**.write(**"id="** + **user**.getId() + **"&pro=-1"**);  
 **bw**.flush();  
 **bw**.close();  
 *//获取响应* **br** = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = **br**.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (**br** != **null**) {  
 **br**.close();  
 }  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String s) {  
 **int** response = Integer.*parseInt*(s);  
 **switch** (response) {  
 **case** 0:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"暂无投票发起,请稍后再试!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **case** -1:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"投票成功,感谢您的参与!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **result\_tv**.setText(**"您的投票结果是:未理解"**);  
 **result\_tv**.setVisibility(View.***VISIBLE***);  
 **break**;  
 **case** -2:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"您已经参加过本次投票!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **default**:  
 **if** (getContext() != **null**) {  
 Toast.*makeText*(getContext(), **"投票失败!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 }  
 **super**.onPostExecute(s);  
 }  
 }.execute(**"http://123.206.106.46:8080/Server/user/vote.do"**);  
 }  
 };  
 **new** Thread() {  
 BufferedReader **br** = **null**;  
 BufferedWriter **bw** = **null**;  
 User **user** = **new** User();  
 @Override  
 **public void** run() {  
 **super**.run();  
 **while** (**true**) {  
 String response = **""**;  
 **try** {  
 Thread.*sleep*(1000);  
 } **catch** (InterruptedException e) {  
 e.printStackTrace();  
 }  
 **if** (**user**.getAuthority() == 2) {  
 **try** {  
 URL url = **new** URL(**"http://123.206.106.46:8080/Server/user/start\_vote.do"**);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 *//请求参数* **bw** = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 **bw**.write(**"id="** + **user**.getId() + **"&authority="** + **user**.getAuthority());  
 **bw**.flush();  
 **bw**.close();  
 *//获取响应* **br** = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = **br**.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (**br** != **null**) {  
 **br**.close();  
 }  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 Bundle bundle = **new** Bundle();  
 bundle.putString(**"response"**, response);  
 Message msg = **new** Message();  
 msg.**what** = 1;  
 msg.setData(bundle);  
 **uiHandler**.sendMessage(msg);  
 }  
 }  
 }.start();  
 **return** view;  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.graphics.Typeface;  
**import** android.os.AsyncTask;  
**import** android.os.Bundle;  
**import** android.support.v4.app.Fragment;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.LinearLayout;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.IOException;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**import** java.text.DecimalFormat;  
**public class** FragmentVoted **extends** Fragment {  
 @Override  
 **public void** onDestroyView() {  
 **super**.onDestroyView();  
 }  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_item1\_1***, **null**);  
 TextView vote\_tv= (TextView) view.findViewById(R.id.***vote\_tv1***);  
 *//字体* Typeface tf=Typeface.*createFromAsset*(getContext().getAssets(),**"fonts/DancingScript-Bold.ttf"**);  
 vote\_tv.setTypeface(tf);  
 **final** LinearLayout result\_layout= (LinearLayout) view.findViewById(R.id.***result\_layout***);  
 result\_layout.setVisibility(View.***INVISIBLE***);  
 **final** TextView tv\_1\_1\_1= (TextView) view.findViewById(R.id.***tv\_1\_1\_1***);  
 **final** TextView tv\_1\_1\_2= (TextView) view.findViewById(R.id.***tv\_1\_1\_2***);  
 **final** TextView tv\_1\_1\_3= (TextView) view.findViewById(R.id.***tv\_1\_1\_3***);  
 **final** TextView tv\_1\_1\_4= (TextView) view.findViewById(R.id.***tv\_1\_1\_4***);  
 **final** TextView tv\_1\_1\_5= (TextView) view.findViewById(R.id.***tv\_1\_1\_5***);  
 Button start\_vote\_btn= (Button) view.findViewById(R.id.***start\_vote***);  
 Button stop\_vote\_btn=(Button) view.findViewById(R.id.***stop\_vote***);  
 start\_vote\_btn.setOnClickListener(**new** View.OnClickListener() {*//发起投票* BufferedReader **br** = **null**;  
 BufferedWriter **bw** = **null**;  
 User **user**=**new** User();  
 @Override  
 **public void** onClick(View v) {  
 result\_layout.setVisibility(View.***INVISIBLE***);  
 **new** AsyncTask<String, Void, String>() {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response=**""**;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection=(HttpURLConnection)url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 *//请求参数* **bw** = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 **bw**.write(**"id="** + **user**.getId()+ **"&authority="**+**user**.getAuthority());  
 **bw**.flush();  
 **bw**.close();  
 *//获取响应* **br**=**new** BufferedReader(**new** InputStreamReader(connection.getInputStream(),**"utf-8"**));  
 String line;  
 **while**((line=**br**.readLine())!=**null**){  
 response+=line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }**finally** {  
 **try** {  
 **if** (**br** != **null**) {  
 **br**.close();  
 }  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String s) {  
 **int** response = Integer.*parseInt*(s);  
 **switch**(response){  
 **case** 0:  
 **if**(getContext()!=**null**){  
 Toast.*makeText*(getContext(),**"投票成功发起!!!"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **case** 1:  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"投票正在进行..."**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 **default**:  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"投票发起失败!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 }  
 **super**.onPostExecute(s);  
 }  
 }.execute(**"http://123.206.106.46:8080/Server/user/start\_vote.do"**);  
 }  
 });  
 stop\_vote\_btn.setOnClickListener(**new** View.OnClickListener() {*//停止投票* BufferedReader **br** = **null**;  
 BufferedWriter **bw** = **null**;  
 User **user**=**new** User();  
 @Override  
 **public void** onClick(View v) {  
 **new** AsyncTask<String, Void, String>() {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response=**""**;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection=(HttpURLConnection)url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 *//请求参数* **bw** = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 **bw**.write(**"id="** + **user**.getId());  
 **bw**.flush();  
 **bw**.close();  
 *//获取响应* **br**=**new** BufferedReader(**new** InputStreamReader(connection.getInputStream(),**"utf-8"**));  
 String line;  
 **while**((line=**br**.readLine())!=**null**){  
 response+=line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }**finally** {  
 **try** {  
 **if** (**br** != **null**) {  
 **br**.close();  
 }  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String s) {  
 **if**(!s.equals(**""**)) {  
 String [] a= s.split(**"-"**,2);  
 **float** pros=Integer.*parseInt*(a[0]);  
 **float** cons=Integer.*parseInt*(a[1]);  
 **float** sum=pros+cons;  
 **float** percentage1=0;  
 **float** percentage2=0;  
 **if**(sum!=0) {  
 percentage1 = pros / sum;  
 percentage2 = cons / sum;  
 }  
 DecimalFormat df=**new** DecimalFormat(**"0.00%"**);  
 tv\_1\_1\_1.setText(**" 本次共有"**+(**int**)sum+**"位同学参与投票,结果如下"**);  
 tv\_1\_1\_2.setText((**int**)pros+**"人"**);  
 tv\_1\_1\_3.setText((**int**)cons+**"人"**);  
 tv\_1\_1\_4.setText(df.format(percentage1));  
 tv\_1\_1\_5.setText(df.format(percentage2));  
 result\_layout.setVisibility(View.***VISIBLE***);  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"成功停止投票!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }**else**{  
 **if**(getContext()!=**null**) {  
 Toast.*makeText*(getContext(), **"暂未发起投票!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 **super**.onPostExecute(s);  
 }  
 }.execute(**"http://123.206.106.46:8080/Server/user/stop\_vote.do"**);  
 }  
 });  
 **return** view;  
 }  
 }

**package** com.project.assistantofcourse;  
**import** android.content.Intent;  
**import** android.os.AsyncTask;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.URL;  
**public class** MainActivity **extends** AppCompatActivity {  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Button loginBtn= (Button) findViewById(R.id.***loginBtn***);  
 **final** EditText editText1= (EditText) findViewById(R.id.***account***);  
 **final** EditText editText2= (EditText) findViewById(R.id.***password***);  
 loginBtn.setOnClickListener(**new** View.OnClickListener() {  
 User **user**=**new** User();  
 String **account**;  
 String **password**;  
 @Override  
 **public void** onClick(View v) {  
 **account**=editText1.getText().toString();  
 **password**=editText2.getText().toString();  
 **if**(Utils.*isFastDoubleClick*()){  
 **return**;  
 }  
 **if**(**account**.equals(**""**)||**password**.equals(**""**))*//注意字符串的比较* {  
 Toast.*makeText*(MainActivity.**this**,**"账号和密码均不能为空!"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **else** {  
 **new** AsyncTask<String, Void, String>() {  
 @Override  
 **protected** String doInBackground(String... params) {  
 String response = **""**;  
 BufferedReader br = **null**;  
 BufferedWriter bw;  
 **try** {  
 URL url = **new** URL(params[0]);  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setConnectTimeout(15000);  
 connection.setReadTimeout(10000);  
 connection.connect();  
 *//请求参数* bw = **new** BufferedWriter(**new** OutputStreamWriter(connection.getOutputStream()));  
 bw.write(**"id="** + **account** + **"&password="** + **password**);  
 bw.flush();  
 bw.close();  
 *//获取响应* br = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream(), **"utf-8"**));  
 String line;  
 **while** ((line = br.readLine()) != **null**) {  
 response += line;  
 }  
 connection.disconnect();  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **try** {  
 **if** (br != **null**) {  
 br.close();  
 }  
 } **catch** (Exception e2) {  
 e2.printStackTrace();  
 }  
 }  
 **return** response;  
 }  
 @Override  
 **protected void** onPostExecute(String s) {  
 **super**.onPostExecute(s);  
 **int** response = Integer.*parseInt*(s);  
 **switch** (response) {  
 **case** -1:  
 Toast.*makeText*(MainActivity.**this**, **"账号不存在或密码错误!"**, Toast.***LENGTH\_SHORT***).show();  
 **break**;  
 **case** 0:*//管理员* **user**.setId(Integer.*parseInt*(**account**));  
 **user**.setAuthority(0);  
 Toast.*makeText*(MainActivity.**this**, **"管理员界面正在开发中..."**, Toast.***LENGTH\_SHORT***).show();  
 **break**;  
 **case** 1:*//教师* **user**.setId(Integer.*parseInt*(**account**));  
 **user**.setAuthority(1);  
 Intent intent1 = **new** Intent(MainActivity.**this**, Teacher\_NavigationActivity.**class**);  
 startActivity(intent1);  
 **break**;  
 **case** 2:*//学生* **user**.setId(Integer.*parseInt*(**account**));  
 **user**.setAuthority(2);  
 Intent intent2 = **new** Intent(MainActivity.**this**, Student\_NavigationActivity.**class**);  
 startActivity(intent2);  
 **break**;  
 **default**:  
 Toast.*makeText*(MainActivity.**this**, **"服务器炸了!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }.execute(**"http://123.206.106.46:8080/Server/user/login.do"**);  
 }  
 }  
 });  
 }  
}

**package** com.project.assistantofcourse;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.app.FragmentPagerAdapter;  
**import** java.util.List;  
**class** MyFragmentAdapter **extends** FragmentPagerAdapter {  
 **private** List<Fragment> **list**;  
 MyFragmentAdapter(FragmentManager fm, List<Fragment> list) {  
 **super**(fm);  
 **this**.**list** = list;  
 }*//写构造方法，方便赋值调用* @Override  
 **public** Fragment getItem(**int** arg0) {  
 **return list**.get(arg0);  
 }*//根据Item的位置返回对应位置的Fragment，绑定item和Fragment* @Override  
 **public int** getCount() {  
 **return list**.size();  
 }*//设置Item的数量*}

**package** com.project.assistantofcourse;  
**import** android.os.Bundle;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentTabHost;  
**import** android.support.v4.view.ViewPager;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.view.KeyEvent;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.widget.ImageView;  
**import** android.widget.TabHost;  
**import** android.widget.TextView;  
**import** java.util.ArrayList;  
**import** java.util.List;  
**public class** Student\_NavigationActivity **extends** AppCompatActivity **implements** ViewPager.OnPageChangeListener, TabHost.OnTabChangeListener {  
 **private** FragmentTabHost **mTabHost**;  
 **private** LayoutInflater **layoutInflater**;  
 **private** Class **fragmentArray**[] = { FragmentVote.**class**, FragmentDiscuss.**class** ,FragmentResource.**class**,FragmentMine.**class**};  
 **private int imageViewArray**[] = { R.drawable.***tab\_poll\_btn***, R.drawable.***tab\_comment\_btn***,R.drawable.***tab\_description\_btn*** ,R.drawable.***tab\_person\_outline\_btn***};  
 **private** String **textViewArray**[] = { **"投票"**,**"讨论"**,**"课件"**,**"我的"**};  
 **private** List<Fragment> **list** = **new** ArrayList<>();  
 **private** ViewPager **vp**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_student\_navigation***);  
 initView();*//初始化控件* initPage();*//初始化页面* }  
 @Override  
 **public boolean** onKeyDown(**int** keyCode, KeyEvent event) {  
 **if** (keyCode == KeyEvent.***KEYCODE\_BACK***) {  
 moveTaskToBack(**true**);  
 **return true**;  
 }  
 **return super**.onKeyDown(keyCode, event);  
 }  
 *// 控件初始化控件* **private void** initView() {  
 **vp** = (ViewPager) findViewById(R.id.***student\_pager***);  
 */\*实现OnPageChangeListener接口,目的是监听Tab选项卡的变化，然后通知ViewPager适配器切换界面\*/  
 /\*简单来说,是为了让ViewPager滑动的时候能够带着底部菜单联动\*/* **vp**.addOnPageChangeListener(**this**);*//设置页面切换时的监听器* **layoutInflater** = LayoutInflater.*from*(**this**);*//加载布局管理器  
 /\*实例化FragmentTabHost对象并进行绑定\*/* **mTabHost** = (FragmentTabHost) findViewById(android.R.id.***tabhost***);*//绑定tahost* **mTabHost**.setup(**this**, getSupportFragmentManager(), R.id.***student\_pager***);*//绑定viewpager  
 /\*实现setOnTabChangedListener接口,目的是为监听界面切换），然后实现TabHost里面图片文字的选中状态切换\*/  
 /\*简单来说,是为了当点击下面菜单时,上面的ViewPager能滑动到对应的Fragment\*/* **mTabHost**.setOnTabChangedListener(**this**);  
 */\*新建Tabspec选项卡并设置Tab菜单栏的内容和绑定对应的Fragment\*/* **for** (**int** i = 0; i < **textViewArray**.**length**; i++) {  
 *// 给每个Tab按钮设置标签、图标和文字* TabHost.TabSpec tabSpec = **mTabHost**.newTabSpec(**textViewArray**[i])  
 .setIndicator(getTabItemView(i));  
 *// 将Tab按钮添加进Tab选项卡中，并绑定Fragment* **mTabHost**.addTab(tabSpec, **fragmentArray**[i], **null**);  
 *// mTabHost.setTag(i);* **mTabHost**.getTabWidget()  
 *//.getChildAt(i)* .setBackgroundResource(R.drawable.***selector\_tab\_background***);*//设置Tab被选中的时候颜色改变* }  
 }  
 */\*初始化Fragment\*/* **private void** initPage() {  
 FragmentVote fragmentVote = **new** FragmentVote();  
 FragmentDiscuss fragmentDiscuss = **new** FragmentDiscuss();  
 FragmentResource fragmentResource = **new** FragmentResource();  
 FragmentMine fragmentMine = **new** FragmentMine();  
 **list**.add(fragmentVote);  
 **list**.add(fragmentDiscuss);  
 **list**.add(fragmentResource);  
 **list**.add(fragmentMine);  
 *//绑定Fragment适配器* **vp**.setAdapter(**new** MyFragmentAdapter(getSupportFragmentManager(), **list**));  
 **mTabHost**.getTabWidget().setDividerDrawable(**null**);  
 }  
 **private** View getTabItemView(**int** i) {  
 *//将xml布局转换为view对象* View view = **layoutInflater**.inflate(R.layout.***tab\_content***, **null**);  
 *//利用view对象，找到布局中的组件,并设置内容，然后返回视图* ImageView mImageView = (ImageView) view  
 .findViewById(R.id.***tab\_imageview***);  
 TextView mTextView = (TextView) view.findViewById(R.id.***tab\_textview***);  
 mImageView.setBackgroundResource(**imageViewArray**[i]);  
 mTextView.setText(**textViewArray**[i]);  
 **return** view;  
 }  
 @Override  
 **public void** onPageScrollStateChanged(**int** arg0) {  
 }*//arg0 ==1的时候表示正在滑动，arg0==2的时候表示滑动完毕了，arg0==0的时候表示什么都没做，就是停在那。* @Override  
 **public void** onPageScrolled(**int** arg0, **float** arg1, **int** arg2) {  
 }*//表示在前一个页面滑动到后一个页面的时候，在前一个页面滑动前调用的方法* @Override  
 **public void** onPageSelected(**int** arg0) {*//arg0是表示你当前选中的页面位置Postion，这事件是在你页面跳转完毕的时候调用的。  
 /\* int oldFocusability = widget.getDescendantFocusability();  
 widget.setDescendantFocusability(ViewGroup.FOCUS\_BLOCK\_DESCENDANTS);//设置View覆盖子类控件而直接获得焦点\*/* **mTabHost**.setCurrentTab(arg0);*//根据位置Postion设置当前的Tab  
 // widget.setDescendantFocusability(oldFocusability);//设置取消分割线* }  
 @Override  
 **public void** onTabChanged(String tabId) {*//Tab改变的时候调用* **int** position = **mTabHost**.getCurrentTab();  
 **vp**.setCurrentItem(position,**false**);*//把选中的Tab的位置赋给适配器，让它控制页面切换\*/* }  
}

**package** com.project.assistantofcourse;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentTabHost;  
**import** android.support.v4.view.ViewPager;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.KeyEvent;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewConfiguration;  
**import** android.widget.ImageView;  
**import** android.widget.TabHost;  
**import** android.widget.TextView;  
**import** java.lang.reflect.Field;  
**import** java.util.ArrayList;  
**import** java.util.List;  
**public class** Teacher\_NavigationActivity **extends** AppCompatActivity **implements** ViewPager.OnPageChangeListener, TabHost.OnTabChangeListener {  
 **private** FragmentTabHost **mTabHost**;  
 **private** LayoutInflater **layoutInflater**;  
 **private** Class **fragmentArray**[] = {FragmentVoted.**class**, FragmentDiscuss.**class**, FragmentResource.**class**, FragmentMine.**class**};  
 **private int imageViewArray**[] = {R.drawable.***tab\_poll\_btn***, R.drawable.***tab\_comment\_btn***, R.drawable.***tab\_description\_btn***, R.drawable.***tab\_person\_outline\_btn***};  
 **private** String **textViewArray**[] = {**"投票"**, **"讨论"**, **"课件"**, **"我的"**};  
 **private** List<Fragment> **list** = **new** ArrayList<>();  
 **private** ViewPager **vp**;  
 @Override  
 **protected void** onDestroy() {  
 **super**.onDestroy();  
 }  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 getOverflowMenu();  
 setContentView(R.layout.***activity\_teacher\_\_navigation***);  
 initView();*//初始化控件* initPage();*//初始化页面* }  
 @Override  
 **public boolean** onKeyDown(**int** keyCode, KeyEvent event) {  
 **if** (keyCode == KeyEvent.***KEYCODE\_BACK***) {  
 moveTaskToBack(**true**);  
 **return true**;  
 }  
 **return super**.onKeyDown(keyCode, event);  
 }  
 **private void** getOverflowMenu() {  
 **try** {  
 ViewConfiguration config = ViewConfiguration.*get*(**this**);  
 Field menuKeyField = ViewConfiguration.**class**.getDeclaredField(**"sHasPermanentMenuKey"**);  
 **if** (menuKeyField != **null**) {  
 menuKeyField.setAccessible(**true**);  
 menuKeyField.setBoolean(config, **false**);  
 }  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 *// 控件初始化控件* **private void** initView() {  
 **vp** = (ViewPager) findViewById(R.id.***teacher\_pager***);  
 */\*实现OnPageChangeListener接口,目的是监听Tab选项卡的变化，然后通知ViewPager适配器切换界面\*/  
 /\*简单来说,是为了让ViewPager滑动的时候能够带着底部菜单联动\*/* **vp**.addOnPageChangeListener(**this**);*//设置页面切换时的监听器* **layoutInflater** = LayoutInflater.*from*(**this**);*//加载布局管理器  
 /\*实例化FragmentTabHost对象并进行绑定\*/* **mTabHost** = (FragmentTabHost) findViewById(android.R.id.***tabhost***);*//绑定tabhost* **mTabHost**.setup(**this**, getSupportFragmentManager(), R.id.***teacher\_pager***);*//绑定viewpager  
 /\*实现setOnTabChangedListener接口,目的是为监听界面切换），然后实现TabHost里面图片文字的选中状态切换\*/  
 /\*简单来说,是为了当点击下面菜单时,上面的ViewPager能滑动到对应的Fragment\*/* **mTabHost**.setOnTabChangedListener(**this**);  
 */\*新建Tabspec选项卡并设置Tab菜单栏的内容和绑定对应的Fragment\*/* **for** (**int** i = 0; i < **textViewArray**.**length**; i++) {  
 *// 给每个Tab按钮设置标签、图标和文字* TabHost.TabSpec tabSpec = **mTabHost**.newTabSpec(**textViewArray**[i])  
 .setIndicator(getTabItemView(i));  
 *// 将Tab按钮添加进Tab选项卡中，并绑定Fragment* **mTabHost**.addTab(tabSpec, **fragmentArray**[i], **null**);  
 *// mTabHost.setTag(i);* **mTabHost**.getTabWidget()  
 *//.getChildAt(i)* .setBackgroundResource(R.drawable.***selector\_tab\_background***);*//设置Tab被选中的时候颜色改变* }  
 }  
 */\*初始化Fragment\*/* **private void** initPage() {  
 FragmentVoted fragmentVoted = **new** FragmentVoted();  
 FragmentDiscuss fragmentDiscuss = **new** FragmentDiscuss();  
 FragmentResource fragmentResource = **new** FragmentResource();  
 FragmentMine fragmentMine = **new** FragmentMine();  
 **list**.add(fragmentVoted);  
 **list**.add(fragmentDiscuss);  
 **list**.add(fragmentResource);  
 **list**.add(fragmentMine);  
 *//绑定Fragment适配器* **vp**.setAdapter(**new** MyFragmentAdapter(getSupportFragmentManager(), **list**));  
 **mTabHost**.getTabWidget().setDividerDrawable(**null**);  
 }  
 **private** View getTabItemView(**int** i) {  
 *//将xml布局转换为view对象* View view = **layoutInflater**.inflate(R.layout.***tab\_content***, **null**);  
 *//利用view对象，找到布局中的组件,并设置内容，然后返回视图* ImageView mImageView = (ImageView) view  
 .findViewById(R.id.***tab\_imageview***);  
 TextView mTextView = (TextView) view.findViewById(R.id.***tab\_textview***);  
 mImageView.setBackgroundResource(**imageViewArray**[i]);  
 mTextView.setText(**textViewArray**[i]);  
 **return** view;  
 }  
 @Override  
 **public void** onPageScrollStateChanged(**int** arg0) {  
 }*//arg0 ==1的时候表示正在滑动，arg0==2的时候表示滑动完毕了，arg0==0的时候表示什么都没做，就是停在那。* @Override  
 **public void** onPageScrolled(**int** arg0, **float** arg1, **int** arg2) {  
 }*//表示在前一个页面滑动到后一个页面的时候，在前一个页面滑动前调用的方法* @Override  
 **public void** onPageSelected(**int** arg0) {*//arg0是表示你当前选中的页面位置Postion，这事件是在你页面跳转完毕的时候调用的。  
 /\* int oldFocusability = widget.getDescendantFocusability();  
 widget.setDescendantFocusability(ViewGroup.FOCUS\_BLOCK\_DESCENDANTS);//设置View覆盖子类控件而直接获得焦点\*/* **mTabHost**.setCurrentTab(arg0);*//根据位置Postion设置当前的Tab  
 // widget.setDescendantFocusability(oldFocusability);//设置取消分割线* }  
 @Override  
 **public void** onTabChanged(String tabId) {*//Tab改变的时候调用* **int** position = **mTabHost**.getCurrentTab();  
 **vp**.setCurrentItem(position, **false**);*//把选中的Tab的位置赋给适配器，让它控制页面切换\*/* }  
}

**package** com.project.assistantofcourse;  
**class** User {  
 **private static long** *id*;  
 **private static** String *name*;  
 **private static int** *authority*;  
 **private static** String *curAnswer*;  
 **public long** getId() {  
 **return** *id*;  
 }  
 **public void** setId(**long** id) {  
 User.*id* = id;  
 }  
 **int** getAuthority() {  
 **return** *authority*;  
 }  
 **void** setAuthority(**int** authority) {  
 User.*authority* = authority;  
 }  
 **public** String getName() {  
 **return** *name*;  
 }  
 **public void** setName(String name) {  
 User.*name* = name;  
 }  
 **void** setCurAnswer(String curAnswer){  
 User.*curAnswer* =curAnswer;  
 }  
 String getCurAnswer(){  
 **return** *curAnswer*;  
 }  
}

**package** com.project.assistantofcourse;  
**import** java.io.File;  
**class** Utils {  
 **private static long** *lastClickTime*;  
 *//消除重键* **static boolean** isFastDoubleClick(){  
 **long** time=System.*currentTimeMillis*();*//返回从 1970年1月1号 00:00:00 到目前的毫秒数* **long** timeD=time-*lastClickTime*;  
 **if**(timeD>0&&timeD<3000){  
 **return true**;  
 }  
 *lastClickTime*=time;  
 **return false**;  
 }  
 *//获取MIME类型* **static** String getMIMEType(File file) {  
 String type=**"\*/\*"**;  
 String fName = file.getName();  
 *//获取后缀名* **int** dotIndex = fName.lastIndexOf(**"."**);  
 **if**(dotIndex < 0){  
 **return** type;  
 }  
 String end=fName.substring(dotIndex,fName.length()).toLowerCase();  
 **if**(end.equals(**""**))**return** type;  
 *//找MIME类型* **for** (String[] aMIME\_MapTable : ***MIME\_MapTable***) {  
 **if** (end.equals(aMIME\_MapTable[0]))  
 type = aMIME\_MapTable[1];  
 }  
 **return** type;  
 }  
 **private static final** String[][] ***MIME\_MapTable***={  
 *//{后缀名，MIME类型}* {**".3gp"**, **"video/3gpp"**},  
 {**".apk"**, **"application/vnd.android.package-archive"**},  
 {**".asf"**, **"video/x-ms-asf"**},  
 {**".avi"**, **"video/x-msvideo"**},  
 {**".bin"**, **"application/octet-stream"**},  
 {**".bmp"**, **"image/bmp"**},  
 {**".c"**, **"text/plain"**},  
 {**".class"**, **"application/octet-stream"**},  
 {**".conf"**, **"text/plain"**},  
 {**".cpp"**, **"text/plain"**},  
 {**".doc"**, **"application/msword"**},  
 {**".docx"**, **"application/vnd.openxmlformats-officedocument.wordprocessingml.document"**},  
 {**".xls"**, **"application/vnd.ms-excel"**},  
 {**".xlsx"**, **"application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"**},  
 {**".exe"**, **"application/octet-stream"**},  
 {**".gif"**, **"image/gif"**},  
 {**".gtar"**, **"application/x-gtar"**},  
 {**".gz"**, **"application/x-gzip"**},  
 {**".h"**, **"text/plain"**},  
 {**".htm"**, **"text/html"**},  
 {**".html"**, **"text/html"**},  
 {**".jar"**, **"application/java-archive"**},  
 {**".java"**, **"text/plain"**},  
 {**".jpeg"**, **"image/jpeg"**},  
 {**".jpg"**, **"image/jpeg"**},  
 {**".js"**, **"application/x-javascript"**},  
 {**".log"**, **"text/plain"**},  
 {**".m3u"**, **"audio/x-mpegurl"**},  
 {**".m4a"**, **"audio/mp4a-latm"**},  
 {**".m4b"**, **"audio/mp4a-latm"**},  
 {**".m4p"**, **"audio/mp4a-latm"**},  
 {**".m4u"**, **"video/vnd.mpegurl"**},  
 {**".m4v"**, **"video/x-m4v"**},  
 {**".mov"**, **"video/quicktime"**},  
 {**".mp2"**, **"audio/x-mpeg"**},  
 {**".mp3"**, **"audio/x-mpeg"**},  
 {**".mp4"**, **"video/mp4"**},  
 {**".mpc"**, **"application/vnd.mpohun.certificate"**},  
 {**".mpe"**, **"video/mpeg"**},  
 {**".mpeg"**, **"video/mpeg"**},  
 {**".mpg"**, **"video/mpeg"**},  
 {**".mpg4"**, **"video/mp4"**},  
 {**".mpga"**, **"audio/mpeg"**},  
 {**".msg"**, **"application/vnd.ms-outlook"**},  
 {**".ogg"**, **"audio/ogg"**},  
 {**".pdf"**, **"application/pdf"**},  
 {**".png"**, **"image/png"**},  
 {**".pps"**, **"application/vnd.ms-powerpoint"**},  
 {**".ppt"**, **"application/vnd.ms-powerpoint"**},  
 {**".pptx"**, **"application/vnd.openxmlformats-officedocument.presentationml.presentation"**},  
 {**".prop"**, **"text/plain"**},  
 {**".rc"**, **"text/plain"**},  
 {**".rmvb"**, **"audio/x-pn-realaudio"**},  
 {**".rtf"**, **"application/rtf"**},  
 {**".sh"**, **"text/plain"**},  
 {**".tar"**, **"application/x-tar"**},  
 {**".tgz"**, **"application/x-compressed"**},  
 {**".txt"**, **"text/plain"**},  
 {**".wav"**, **"audio/x-wav"**},  
 {**".wma"**, **"audio/x-ms-wma"**},  
 {**".wmv"**, **"audio/x-ms-wmv"**},  
 {**".wps"**, **"application/vnd.ms-works"**},  
 {**".xml"**, **"text/plain"**},  
 {**".z"**, **"application/x-compress"**},  
 {**".zip"**, **"application/x-zip-compressed"**},  
 {**""**, **"\*/\*"**}  
 };  
}