package com.example.overwork;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.DatePickerDialog;  
import android.content.Intent;  
import android.os.Bundle;  
import android.text.format.DateUtils;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.DatePicker;  
import android.widget.EditText;  
import android.widget.Spinner;  
import android.widget.TextView;  
import android.widget.Toast;  
  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.InputStreamReader;  
import java.io.OutputStream;  
import java.io.UnsupportedEncodingException;  
import java.net.HttpURLConnection;  
import java.net.MalformedURLException;  
import java.net.ProtocolException;  
import java.net.URL;  
import java.util.Calendar;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class MainActivity extends AppCompatActivity implements View.OnClickListener {  
  
  
 Button input, data\_input;  
 EditText down, sit;  
 String[] sex = {"м","ж"};  
 TextView date;  
 String current\_sex;  
 Calendar Date= Calendar.*getInstance*();  
 Spinner spinner;  
 int day, month, year, intSex, pulse1, pulse2, result;  
  
  
  
 @Override

protected void onCreate(Bundle savedInstanceState) {   
 super.onCreate(savedInstanceState); // вызов метода OnCreate из родительского класса  
 setContentView(R.layout.*activity\_main*); // активити мейн  
 date = findViewById(R.id.*date*); // связка переменных

down = findViewById(R.id.*down*);  
 sit = findViewById(R.id.*sit*);  
 input = findViewById(R.id.*input*);  
 input.setOnClickListener(this);  
 data\_input = findViewById(R.id.*input\_data*);  
 data\_input.setOnClickListener(this);  
 // связь списка со спинером  
 ArrayAdapter<String> adapter = new ArrayAdapter<String>(this, android.R.layout.*simple\_spinner\_item*, sex);  
 adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*);  
 spinner = (Spinner) findViewById(R.id.*male*);  
 spinner.setAdapter(adapter);  
   
 spinner.setSelection(0);  
 //обработчик нажатия  
 spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() { // обработчик нажатия  
 @Override  
 public void onItemSelected(AdapterView<?> parent, View view, // метод вызывается по нажатию на значение  
 int position, long id) {  
 current\_sex = sex[position];  
 }  
 @Override  
 public void onNothingSelected(AdapterView<?> arg0) {  
 }  
 });  
 setInitialDate();  
  
 }  
  
  
 // обработчик выбора даты  
 DatePickerDialog.OnDateSetListener d= new DatePickerDialog.OnDateSetListener() {  
 public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth) {  
 Date.set(Calendar.*YEAR*, year);  
 Date.set(Calendar.*MONTH*, monthOfYear);  
 Date.set(Calendar.*DAY\_OF\_MONTH*, dayOfMonth);  
 setInitialDate(); // обработик даты  
 }  
 };  
  
  
  
 @Override  
 public void onClick(View view) { // проверка нажатия  
 switch (view.getId()){  
 case R.id.*input*:  
 if (down.getText().toString().equals("") || sit.getText().toString().equals("")) {  
 Toast.*makeText*(this, "Заполните все поля!", Toast.*LENGTH\_SHORT*).show();  
 }  
 else {  
 year = Integer.*parseInt*(String.*valueOf*(Date.get(Calendar.*YEAR*)));  
 month = Integer.*parseInt*(String.*valueOf*(Date.get(Calendar.*MONTH*)));  
 day = Integer.*parseInt*(String.*valueOf*(Date.get(Calendar.*DAY\_OF\_MONTH*)));  
 switch(current\_sex){  
 case "м":  
 intSex = 1;  
 break;  
 case "ж":  
 intSex = 2;  
 }  
 pulse1 = Integer.*parseInt*(String.*valueOf*(down.getText()));  
 pulse2 = Integer.*parseInt*(String.*valueOf*(sit.getText()));  
 Thread thread = new Thread(task);  
 thread.start();  
 try {  
 thread.join();  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
 // переключение на второе окно  
 Intent intent = new Intent(this, ResultActivity.class); // создаем интент  
 intent.putExtra("result", result); // добавляем пезультат  
 startActivity(intent); // запускаем интент  
 }  
 break;  
 case R.id.*input\_data*:  
 // отображаем выбор даты  
 new DatePickerDialog(MainActivity.this, d,  
 Date.get(Calendar.*YEAR*),  
 Date.get(Calendar.*MONTH*),  
 Date.get(Calendar.*DAY\_OF\_MONTH*))  
 .show();  
 break;  
 }  
 }  
  
  
 // установка начальной даты  
 private void setInitialDate() { // метод где форматируются значение из Date в текст, который идет в TextView  
 date.setText(DateUtils.*formatDateTime*(this,  
 Date.getTimeInMillis(),  
 DateUtils.*FORMAT\_SHOW\_DATE* | DateUtils.*FORMAT\_SHOW\_YEAR*));  
 }  
  
 Runnable task = new Runnable() {  
  
 @Override  
 public void run() {  
 try {  
 URL url = new URL("http://abashin.ru/cgi-bin/ru/tests/burnout");  
 HttpURLConnection connection = (HttpURLConnection) url.openConnection();  
 connection.setRequestMethod("POST"); // метод отправки запроса  
  
 connection.setRequestProperty("Connection", "close"); // свойства запроса  
  
 connection.setRequestProperty(  
 "Accept-Language",  
 "ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7");  
  
 connection.setDoOutput(true); // разрешает отправку тела через соединение  
 connection.setInstanceFollowRedirects(false); // устанавливает, должен ли этот HttpURLConnection экземпляр автоматически следовать за перенаправлениями HTTP  
 String parameters = "day="+day+"&month="+month+"&year="+year+"&sex="+intSex+"&m1="+pulse1+"&m2="+pulse2;  
 OutputStream out = connection.getOutputStream(); // получение потока для передачи данных  
 out.write(parameters.getBytes("UTF-8"));   
 out.close(); // закрываем поток  
 InputStream stream = connection.getInputStream(); // встречный поток  
 BufferedReader in = new BufferedReader(new InputStreamReader(stream)); // читаем данные от сервера  
 String data = null;  
 StringBuffer response = new StringBuffer(); // переменная  
 while ((data = in.readLine()) != null) { // записываем значения в data

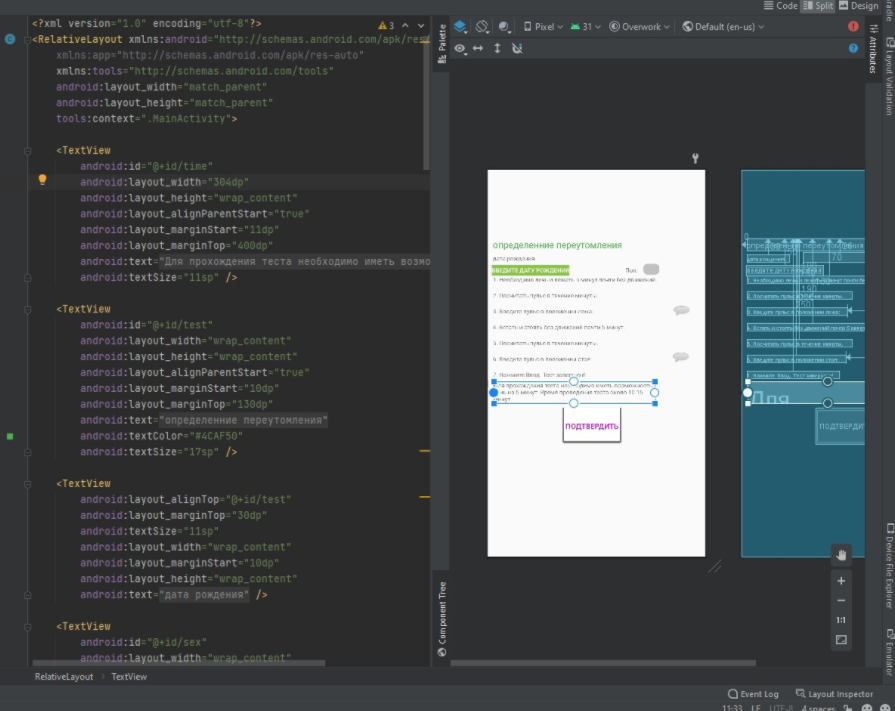
response.append(data);  
 }  
 in.close();  
  
  
 String parsedResponse = new String(response.toString().getBytes(), "UTF-8");

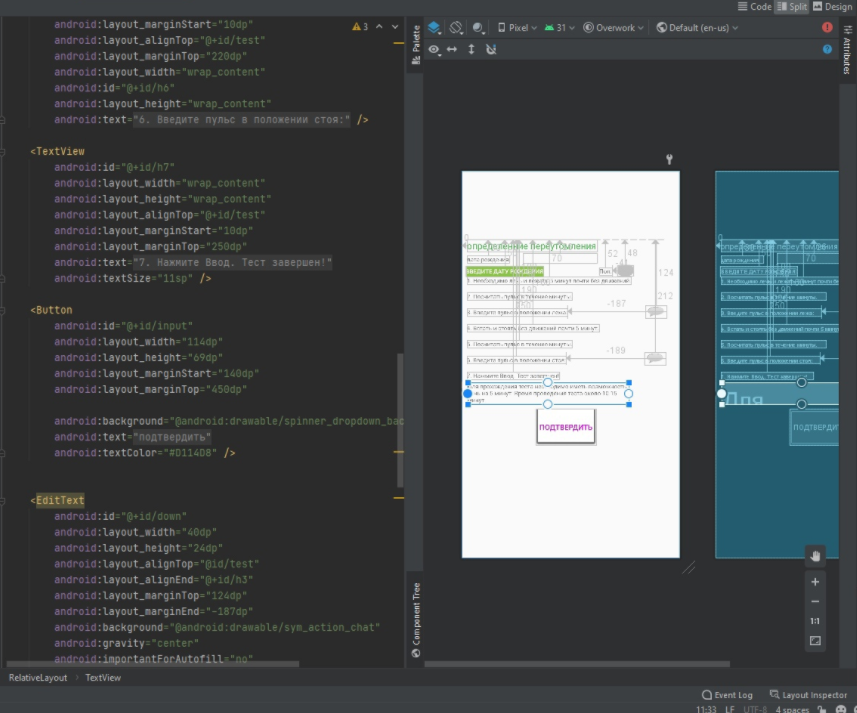
Pattern pattern = Pattern.*compile*("([А**-**Я].\*\\.)"); // шаблон для поиска ответа  
 Matcher matcher = pattern.matcher(parsedResponse); // ответ  
 if (matcher.find()) { // если матчер нашел, то заново запускаем парсинг для нахождения нужного слова  
 String answer = matcher.group(1); // сравнение строки со словом  
 pattern = Pattern.*compile*("([т][ ][а**-**я]\*[ ])");  
 matcher = pattern.matcher(answer);  
 matcher.find();  
 String value = matcher.group(1);  
 value = value.substring(2, value.length() - 1);  
 if (value.equals("высокому")) {  
 result = 2;  
 } else if (value.equals("небольшому")) {  
 result = 3;  
 } else {  
 result = 4;  
 }  
 } else {  
 result = 1;//error }  
  
 } catch (UnsupportedEncodingException e) { // "поимка" ошибки  
 e.printStackTrace();  
 } catch (ProtocolException e) {  
 e.printStackTrace();  
 } catch (MalformedURLException e) {  
 e.printStackTrace();  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 };  
}

second\_activity

package com.example.overwork;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.os.Bundle;  
import android.widget.ImageView;  
import android.widget.ProgressBar;  
import android.widget.TextView;  
  
public class ResultActivity extends AppCompatActivity {  
  
 ProgressBar progress;  
 TextView res;  
 int result;  
 ImageView Image;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_result*);  
 res = findViewById(R.id.*result*);  
 Image = findViewById(R.id.*smile*);  
 progress = findViewById(R.id.*progressBar*);  
 Bundle arguments = getIntent().getExtras(); // словарь  
 result = (int) arguments.get("result");  
 switch (result){  
 case 4:  
 res.setText(R.string.*cool*);  
 Image.setImageResource(android.R.drawable.*presence\_online*);  
 progress.setProgress(100);  
 break;  
 case 3:  
 res.setText(R.string.*normal*);  
 Image.setImageResource(android.R.drawable.*toast\_frame*);  
 progress.setProgress(66);  
 break;  
 case 2:  
 res.setText(R.string.*bad*);  
 Image.setImageResource(android.R.drawable.*ic\_notification\_overlay*);  
 progress.setProgress(33);  
 break;  
 case 1:  
 res.setText(R.string.*worst*);  
 progress.setProgress(0);  
 break;  
 }  
  
 }  
}

xml





xml 2

