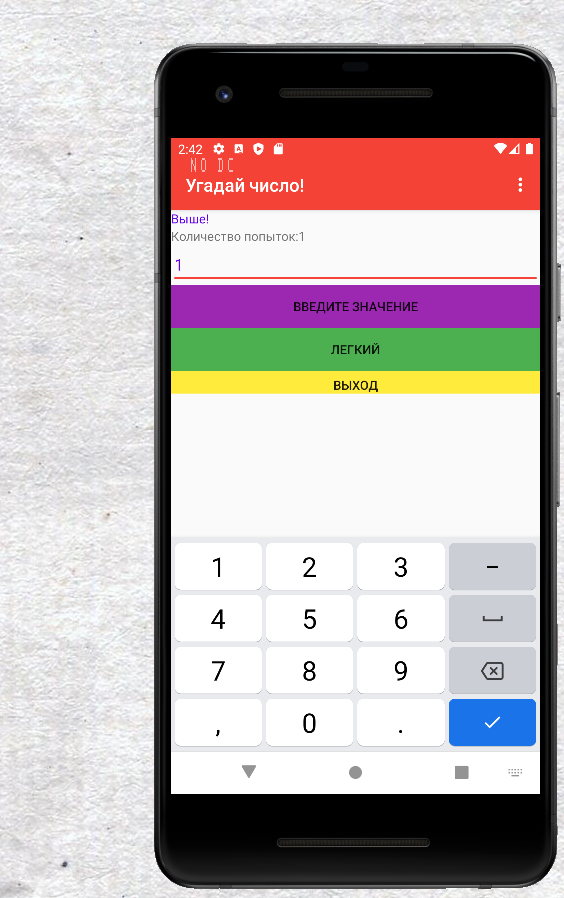
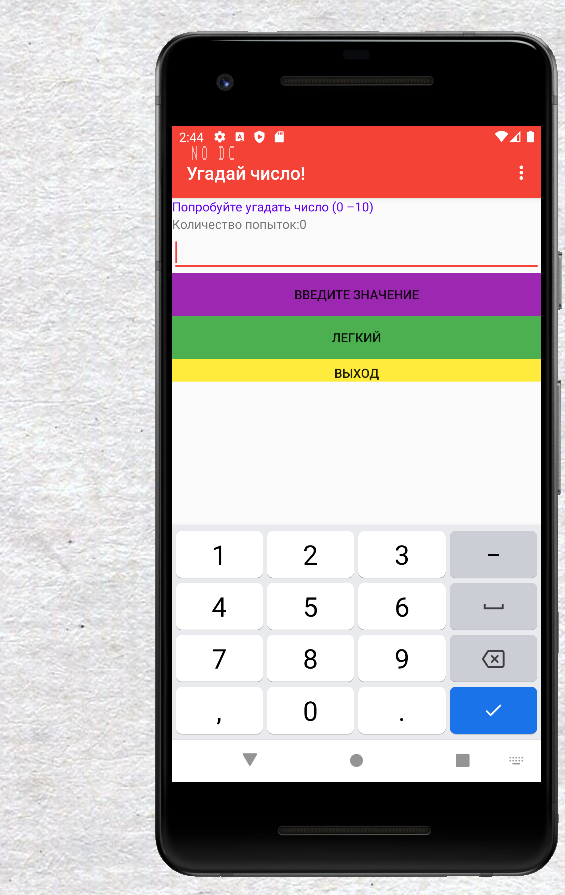
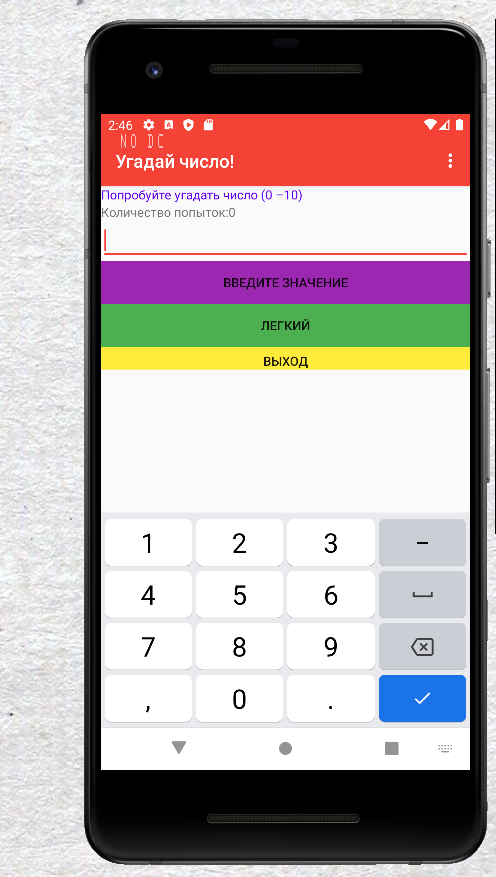
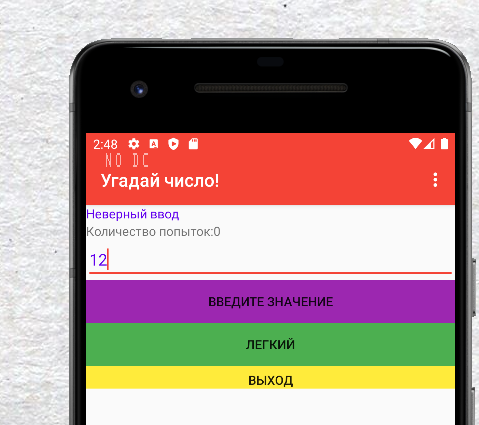
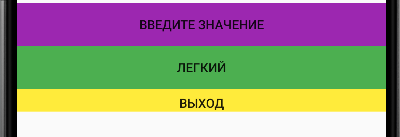
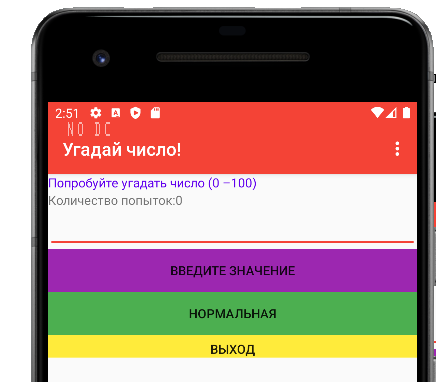
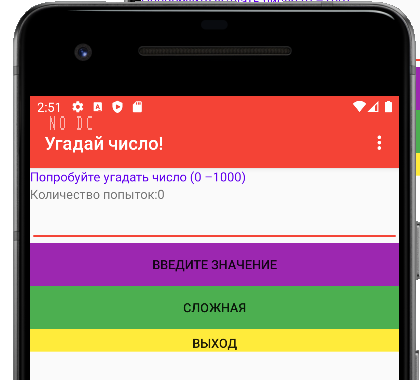
Егоров Сергей ПИ20-5 «Угадай число»

Демонстрация работы:

1. Была сделана работающая панель с функцией «Ввод числа»:  
     
    
2. При вводе числа, которое не соответствует нужным параметрам, будет создаваться соответствующее оповещение о том, что было введено неправильное число:  
    
3. Также была реализовна возможность «Выход» для того, чтобы выйти из приложения:  
   
4. Была сделана возможность для пользователя выбрать «Уровень сложности»:  
     
     
   

Далее будет показан процесс работы из файлов:

1. Процесс работы из «activity\_main.xml»:

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
  
 <TableLayout   
  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="horizontal"  
 android:stretchColumns="\*"  
 android:textColor="#34BFBE">  
  
 <TableRow  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/try\_to\_guess"  
 android:textColor="@color/purple\_500" />  
  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content" />  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content">  
  
 <EditText  
 android:id="@+id/editTextNumber"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:inputType="number"  
 android:textColor="@color/purple\_500"  
 android:visibility="visible" />  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content">  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
  
 android:background="#9C27B0"  
 android:onClick="onClick"  
 android:text="@string/input\_value" />  
  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:id="@+id/button3"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#4CAF50"  
 android:onClick="onDifficult"  
 android:text="@string/medium" />  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="194dp"  
 android:layout\_height="25dp"  
 android:alpha="12"  
 android:background="#FFEB3B"  
 android:gravity="bottom|center"  
 android:onClick="onExit"  
 android:paddingBottom="-100dp"  
 android:text="@string/exit" />  
  
 </TableRow>  
  
 </TableLayout>  
  
</LinearLayout>

1. Процесс работы из «activity\_main.java»:

import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.\*;  
  
public class MainActivity extends AppCompatActivity {  
 TextView tvInfo, tvAttempts;  
 EditText etInput;  
 Button bControl, bDifficult;  
 int guess\_number, current\_value, attempts, difficult = 10;  
 boolean game\_over = false;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 tvInfo = (TextView)findViewById(R.id.*textView*);  
 etInput = (EditText)findViewById(R.id.*editTextNumber*);  
 bControl = (Button)findViewById(R.id.*button*);  
 bDifficult = (Button)findViewById(R.id.*button3*);  
 tvAttempts = (TextView)findViewById(R.id.*textView2*);  
 attempts = 0;  
 guess\_number = (int) (Math.*random*() \* (10 \* difficult + 1));  
 tvAttempts.setText(getResources().getString(R.string.*attempts*) + attempts);  
 tvInfo.setText(getResources().getString(R.string.*try\_to\_guess*) + 10 \* difficult + ")");  
 }  
 public void onClick(View v) {  
 if (game\_over) {  
 guess\_number = (int) (Math.*random*() \* (10 \* difficult + 1));  
 attempts = 0;  
 game\_over = false;  
 bControl.setText(getResources().getString(R.string.*input\_value*));  
 tvInfo.setText(getResources().getString(R.string.*try\_to\_guess*) + 10 \* difficult + ")");  
 } else {  
 if (!etInput.getText().toString().isEmpty()) {  
 current\_value = Integer.*parseInt*(etInput.getText().toString());  
 if ((0 <= current\_value) && (10 \* difficult >= current\_value)) {  
 ++attempts;  
 if (current\_value == guess\_number) {  
 tvInfo.setText(getResources().getString(R.string.*bingo*));  
 game\_over = true;  
 bControl.setText(getResources().getString(R.string.*play\_more*));  
 etInput.getText().clear();  
 } else {  
 if (current\_value > guess\_number) {  
 tvInfo.setText(getResources().getString(R.string.*lower*));  
 } else {  
 tvInfo.setText(getResources().getString(R.string.*up*));  
 }  
 }  
 } else {  
 tvInfo.setText(getResources().getString(R.string.*error*));  
 }  
 } else {  
 tvInfo.setText(getResources().getString(R.string.*try\_to\_guess*) + 10 \* difficult + ")");  
 }  
  
 }  
 tvAttempts.setText(getResources().getString(R.string.*attempts*) + attempts);  
 }  
 public void onExit(View v) {  
 finish();  
 }  
  
  
 public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.*mymenu*, menu);  
 return super.onCreateOptionsMenu(menu);  
 }  
 public void Rewrite() {  
 tvInfo.setText(getResources().getString(R.string.*try\_to\_guess*) + 10 \* difficult + ")");  
 guess\_number = (int) (Math.*random*() \* (10 \* difficult+1));  
 attempts = 0;  
 game\_over = false;  
 bControl.setText(getResources().getString(R.string.*input\_value*));  
 tvAttempts.setText(getResources().getString(R.string.*attempts*) + attempts);  
 etInput.getText().clear();  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 switch (item.getItemId()) {  
 case R.id.*restart*:  
 Rewrite();  
 break;  
  
 case R.id.*exit*:  
 finish();  
 break;  
 default:  
 switch (item.getItemId()) {  
 case R.id.*easy*:  
 difficult = 1;  
 bDifficult.setText(getResources().getString(R.string.*easy*));  
 break;  
 case R.id.*medium*:  
 difficult = 10;  
 bDifficult.setText(getResources().getString(R.string.*medium*));  
 break;  
 case R.id.*hard*:  
 difficult = 100;  
 bDifficult.setText(getResources().getString(R.string.*hard*));  
 break;  
 };  
 Rewrite();  
 break;  
  
  
 };  
 return super.onOptionsItemSelected(item);  
 }  
  
 public void onDifficult(View v) {  
 if (difficult == 10) {  
 difficult = 100;  
 bDifficult.setText(getResources().getString(R.string.*hard*));  
 } else if (difficult == 100) {  
 difficult = 1;  
 bDifficult.setText(getResources().getString(R.string.*easy*));  
 } else {  
 difficult = 10;  
 bDifficult.setText(getResources().getString(R.string.*medium*));  
 }  
 Rewrite();  
 }  
}

3) Процесс работы из «strings.xml":

<resources>  
 <string name="app\_name">Угадай число</string>  
 <string name="up">Выше</string>  
 <string name="lower">Ниже</string>  
 <string name="bingo">Молодец!</string>  
 <string name="input\_value">Введите значение</string>  
 <string name="play\_more">Сыграть ещё раз</string>  
 <string name="try\_to\_guess">Попробуйте угадать число (0 &#8211; </string>  
 <string name="error">Неверный ввод</string>  
 <string name="exit">Выход</string>  
 <string name="attempts">Количество попыток: </string>  
 <string name="easy">Легкий</string>  
 <string name="medium">Нормальная</string>  
 <string name="hard">Сложная</string>  
</resources>