## Создание главного меню приложения

### Цель работы

Познакомиться с инструментами и приемами создания меню мобильного приложения и способами взаимодействия с ним.

### Дополнительные задания

1. Попробуйте добавить еще несколько пунктов в меню, чтобы их стало больше шести. И обратите внимание, как они отобразятся.
2. Реализуйте меню с пунктом “выход”, при выборе которого приложение должно закрыться.
3. Модифицируйте игру “Угадай число” таким образом, чтобы в меню можно было выбирать уровень сложности, начинать новую игру и выходить из приложения.

MainActivity.java

package com.example.guessthenumbermenu;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 protected final int EASY\_MODE\_MAX = 10;  
 protected final int MEDIUM\_MODE\_MAX = 100;  
 protected final int HARD\_MODE\_MAX = 1000;  
  
 private int random;  
 private int max;  
  
 TextView info;  
 EditText input;  
 Button button;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 info = findViewById(R.id.*info*);  
 input = findViewById(R.id.*input*);  
 button = findViewById(R.id.*button*);  
  
 startNewGame(MEDIUM\_MODE\_MAX);  
  
 button.setOnClickListener(  
 new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String text = input.getText().toString();  
 if (text.equals("")) {  
 setTextRes(R.string.*error*);  
 } else {  
 int number = Integer.*parseInt*(text);  
 if (number > max) {  
 setTextRes(R.string.*error*);  
 } else if (number < random) {  
 setTextRes(R.string.*ahead*);  
 } else if (number > random) {  
 setTextRes(R.string.*behind*);  
 } else {  
 onFinish();  
 }  
 if (number != random) {  
 clearInput();  
 }  
 }  
 }  
 }  
 );  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.*main\_menu*, menu);  
 return super.onCreateOptionsMenu(menu);  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 if (item.getItemId() == R.id.*newGame*) {  
 startNewGame(max);  
 }  
 else if (item.getItemId() == R.id.*easyMode*) {  
 startNewGame(EASY\_MODE\_MAX);  
 }  
 else if (item.getItemId() == R.id.*mediumMode*) {  
 startNewGame(MEDIUM\_MODE\_MAX);  
 }  
 else if (item.getItemId() == R.id.*hardMode*) {  
 startNewGame(HARD\_MODE\_MAX);  
 }  
 else if (item.getItemId() == R.id.*quit*) {  
 this.finishAffinity();  
 }  
 return super.onOptionsItemSelected(item);  
 }  
  
 private void setRandom() {  
 random = (int) (Math.*random*() \* max);  
 }  
  
 private void setTextRes(int strResId) {  
 info.setText(getResources().getString(strResId));  
 }  
  
 private void startNewGame(int max) {  
 this.max = max;  
 setRandom();  
 info.setText(getTryToGuessMessage());  
 input.setEnabled(true);  
 button.setEnabled(true);  
 clearInput();  
 }  
  
 private String getTryToGuessMessage() {  
 return "Попробуйте угадать число (от 0 до " + max + ")";  
 }  
  
 private void onFinish() {  
 setTextRes(R.string.*hit*);  
 input.setEnabled(false);  
 button.setEnabled(false);  
 }  
  
 private void clearInput() {  
 input.getText().clear();  
 }  
}

activity\_main.xml

*<?*xml version="1.0" encoding="utf-8"*?>*<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".MainActivity">  
  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <TextView  
 android:id="@+id/info"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/try\_to\_guess"  
 android:textAlignment="center" />  
  
 <EditText  
 android:id="@+id/input"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:inputType="number"  
 android:minHeight="48dp" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:text="@string/enter\_button" />  
  
 </LinearLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

main\_menu.xml

*<?*xml version="1.0" encoding="utf-8"*?>*<menu xmlns:android="http://schemas.android.com/apk/res/android">  
 <item  
 android:id="@+id/newGame"  
 android:title="@string/new\_game">  
 </item>  
 <item  
 android:id="@+id/easyMode"  
 android:title="@string/easy">  
 </item>  
 <item  
 android:id="@+id/mediumMode"  
 android:title="@string/medium">  
 </item>  
 <item  
 android:id="@+id/hardMode"  
 android:title="@string/hard">  
 </item>  
 <item  
 android:id="@+id/quit"  
 android:title="@string/quit">  
 </item>  
</menu>

strings.xml

<resources>  
 <string name="app\_name">Угадай число</string>  
  
 <string name="behind">Загаданное число меньше</string>  
 <string name="ahead">Загаданное число больше</string>  
 <string name="hit">Вы угадали!</string>  
  
 <string name="error">Неверный ввод</string>  
  
 <string name="enter\_button">Ввести</string>  
  
 <string name="try\_to\_guess">Попробуйте угадать число (от 0 до 100)</string>  
  
 <string name="new\_game">Новая игра</string>  
 <string name="easy">Легкий уровень</string>  
 <string name="medium">Средний уровень</string>  
 <string name="hard">Сложный уровень</string>  
 <string name="quit">Выйти</string>  
</resources>



  