**package** com.amandamcnair.assignment3;  
  
**import** android.app.AlertDialog;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.content.SharedPreferences;  
**import** android.media.AudioAttributes;  
**import** android.media.SoundPool;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.util.Log;  
**import** android.view.MotionEvent;  
**import** android.view.View;  
**import** android.view.animation.AlphaAnimation;  
**import** android.view.animation.Animation;  
**import** android.view.animation.LinearInterpolator;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** java.util.HashSet;  
**import** java.util.Random;  
**import** java.util.Set;  
  
**public class** SimonOriginalGame **extends** AppCompatActivity {  
  
 **private** TextView **scoreText**;  
 **private** TextView **highestScoreText**;  
 **private int buttonClick**;  
 **private int choice**;  
 **private int numOfBlocksToClick** = 0;  
 **private int numOfClicks** = 0;  
  
 **private int plays**[] = **new int**[20];  
  
 **private int score** = 0;  
 **private int highestScore** = 0;  
 Random **random** = **new** Random();  
 **final** Handler **handler** = **new** Handler();  
 *//private SoundPool soundPool;  
 //private int lose = soundPool.load(this, R.raw.lose, 1);* **private** SoundPool **soundPool**;  
 **private** Set<Integer> **soundsLoaded**;  
 **private int bell**;  
 **private int ding**;  
 **private int dong**;  
 **private int high\_ding**;  
 **private int lose**;  
 **private** Animation **animation** = **new** AlphaAnimation(1, 0);  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***simonoriginalgame***);  
  
 getSupportActionBar().setDisplayShowHomeEnabled(**true**);  
 getSupportActionBar().setLogo(R.mipmap.***ic\_launcher***);  
 getSupportActionBar().setDisplayUseLogoEnabled(**true**);  
  
 SharedPreferences prefs = **this**.getSharedPreferences(**"GET\_HIGH\_SCORE"**, getApplicationContext().***MODE\_PRIVATE***);  
 **highestScore** = prefs.getInt(**"HIGH\_SCORE"**, 0);  
 runOnUiThread(**new** Runnable() {  
 **public void** run() {  
 TextView tv = findViewById(R.id.***highestscore\_textview***);  
 tv.setText(**"High score: "** + **highestScore**);  
 Log.*i*(**"HIGH SCORE"**, **"High score: "** + **highestScore**);  
 }  
 });  
  
  
 **soundsLoaded** = **new** HashSet<Integer>();  
  
 AudioAttributes.Builder attributeBuilder = **new** AudioAttributes.Builder();  
 attributeBuilder.setUsage(AudioAttributes.***USAGE\_GAME***);  
  
 SoundPool.Builder spBuilder = **new** SoundPool.Builder();  
 spBuilder.setAudioAttributes(attributeBuilder.build());  
  
 spBuilder.setMaxStreams(1);  
  
 **soundPool** = spBuilder.build();  
  
 **soundPool**.setOnLoadCompleteListener(**new** SoundPool.OnLoadCompleteListener() {  
 @Override  
 **public void** onLoadComplete(SoundPool soundPool, **int** sampleID, **int** status) {  
 **if** (status == 0) *// success* {  
 **soundsLoaded**.add(sampleID);  
 Log.*i*(**"SOUND"**, **"Sound loaded "** + sampleID);  
 } **else** {  
 Log.*i*(**"SOUND"**, **"Error cannot load sound status = "** + status);  
 }  
 }  
 });  
  
 **scoreText** = findViewById(R.id.***score\_textview***);  
 **highestScoreText** = findViewById(R.id.***highestscore\_textview***);  
  
 **bell** = **soundPool**.load(**this**, R.raw.***bell***, 1);  
 **ding** = **soundPool**.load(**this**, R.raw.***ding***, 1);  
 **dong** = **soundPool**.load(**this**, R.raw.***dong***, 1);  
 **high\_ding** = **soundPool**.load(**this**, R.raw.***high\_ding***, 1);  
 **lose** = **soundPool**.load(**this**, R.raw.***lose***, 1);  
  
 findViewById(R.id.***red\_button***).setOnTouchListener(**clicked**);  
 findViewById(R.id.***green\_button***).setOnTouchListener(**clicked**);  
 findViewById(R.id.***blue\_button***).setOnTouchListener(**clicked**);  
 findViewById(R.id.***yellow\_button***).setOnTouchListener(**clicked**);  
  
 findViewById(R.id.***red\_button***).setEnabled(**false**);  
 findViewById(R.id.***green\_button***).setEnabled(**false**);  
 findViewById(R.id.***blue\_button***).setEnabled(**false**);  
 findViewById(R.id.***yellow\_button***).setEnabled(**false**);  
  
 findViewById(R.id.***start\_button***).setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 playGame();  
 findViewById(R.id.***start\_button***).setEnabled(**false**);  
 Toast.*makeText*(getApplicationContext(), **"Game has begun!"**, Toast.***LENGTH\_SHORT***).show();  
  
 }  
 });  
  
 }  
  
 View.OnTouchListener **clicked** = **new** View.OnTouchListener() {  
 @Override  
 **public boolean** onTouch(View v, MotionEvent event) {  
 **if** (event.getAction() == MotionEvent.***ACTION\_UP***) {  
  
  
 *// this is the computer creating the buttons* **switch** (v.getId()) {  
 **case** R.id.***red\_button***:  
 **buttonClick** = 1;  
 playSound(**bell**);  
 */\*animation = new AlphaAnimation(1, 0);  
 animation.setDuration(300);  
 animation.setInterpolator(new LinearInterpolator());  
 findViewById(R.id.red\_button).startAnimation(animation);\*/* **break**;  
 **case** R.id.***green\_button***:  
 **buttonClick** = 2;  
 */\*animation = new AlphaAnimation(1, 0);  
 animation.setDuration(300);  
 animation.setInterpolator(new LinearInterpolator());  
 findViewById(R.id.green\_button).startAnimation(animation);\*/* **break**;  
 **case** R.id.***blue\_button***:  
 **buttonClick** = 3;  
 */\*animation = new AlphaAnimation(1, 0);  
 animation.setDuration(300);  
 animation.setInterpolator(new LinearInterpolator());  
 findViewById(R.id.blue\_button).startAnimation(animation);\*/* **break**;  
 **case** R.id.***yellow\_button***:  
 **buttonClick** = 4;  
 */\*animation = new AlphaAnimation(1, 0);  
 animation.setDuration(300);  
 animation.setInterpolator(new LinearInterpolator());  
 findViewById(R.id.yellow\_button).startAnimation(animation);\*/* **break**;  
 }  
  
 **if** (**plays**[**numOfClicks**] != **buttonClick**)  
 {  
 playSound(**lose**);  
 findViewById(R.id.***red\_button***).setEnabled(**false**);  
 findViewById(R.id.***green\_button***).setEnabled(**false**);  
 findViewById(R.id.***blue\_button***).setEnabled(**false**);  
 findViewById(R.id.***yellow\_button***).setEnabled(**false**);  
  
 findViewById(R.id.***start\_button***).setEnabled(**true**);  
 *//numOfClicks = 0;  
 //buttonClick = 0;* Toast.*makeText*(getApplicationContext(), **"GAME OVER!"**, Toast.***LENGTH\_SHORT***).show();  
 AlertDialog.Builder builder = **new** AlertDialog.Builder(SimonOriginalGame.**this**); *// need a new one because of running activity* builder.setTitle(**"GAME OVER!"**);  
 *//builder.setMessage("You lost :( \n Click 'Play again!' or 'home' to go back to home.");* builder.setMessage(**"You lost :( \n Your score was "** + **score** + **"\nClick 'home' to go back to home."**);  
  
 builder.setNegativeButton(**"HOME"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** choice) {  
 *// Dismiss Dialog* Intent i = **new** Intent(getApplicationContext(), MainActivity.**class**);  
 getApplicationContext().startActivity(i);  
 }  
 });  
  
 */\*builder.setPositiveButton("Play again!", new DialogInterface.OnClickListener() {  
 public void onClick(DialogInterface dialog, int choice) {  
 }  
 });\*/* AlertDialog dialog = builder.create();  
 dialog.show();  
 dialog.getWindow().setLayout(1100, 600);  
  
 **return true**;  
 }  
 *//if the user gets its right* **if** (v.getId() == R.id.***red\_button***)  
 {  
 playSound(**bell**);  
 *// when I click, it will animate* **animation** = **new** AlphaAnimation(1, 0);  
 **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***red\_button***).startAnimation(**animation**);  
 Log.*i*(**"PRESSED RED"**, **"Red"**);  
 }  
 **else if** (v.getId() == R.id.***green\_button***)  
 {  
 playSound(**ding**);  
 *// when I click, it will animate* **animation** = **new** AlphaAnimation(1, 0);  
 **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***green\_button***).startAnimation(**animation**);  
 Log.*i*(**"PRESSED GREEN"**, **"Green"**);  
 }  
 **else if** (v.getId() == R.id.***blue\_button***)  
 {  
 playSound(**dong**);  
 *// when I click, it will animate* **animation** = **new** AlphaAnimation(1, 0);  
 **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***blue\_button***).startAnimation(**animation**);  
 Log.*i*(**"PRESSED BLUE"**, **"Blue"**);  
 }  
 **else if** (v.getId() == R.id.***yellow\_button***)  
 {  
 playSound(**high\_ding**);  
 *// when I click, it will animate* **animation** = **new** AlphaAnimation(1, 0);  
 **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***yellow\_button***).startAnimation(**animation**);  
 Log.*i*(**"PRESSED YELLOW"**, **"Yellow"**);  
 }  
 **numOfClicks**++;  
  
  
 **if** (**numOfBlocksToClick** == **numOfClicks**) {  
  
 **score**++;  
 **scoreText**.setText(**"Score: "** + **score**);  
  
 **numOfClicks** = 0;  
 **if** (**numOfBlocksToClick** > **highestScore**) {  
 **highestScore** = **numOfBlocksToClick**;  
 SharedPreferences highScores = getSharedPreferences(**"GET\_HIGH\_SCORE"**, Context.***MODE\_PRIVATE***);  
 SharedPreferences.Editor editor = highScores.edit();  
 editor.putInt(**"HIGH\_SCORE"**, **highestScore**);  
 editor.commit();  
  
 **highestScoreText**.setText(**"High score: "** + **highestScore**);  
  
 }  
 **final** Runnable runnable = **new** Runnable() {  
 **public void** run() {  
 playGame();  
 }  
 };  
 **handler**.postDelayed(runnable, 1000);  
 }  
 }  
 **return true**;  
 }  
 };  
  
  
  
 */\*findViewById(R.id.red\_button).setOnTouchListener(new View.OnTouchListener() {  
 @Override  
 public boolean onTouch(View view, MotionEvent motionEvent) {  
 if(motionEvent.getAction() == MotionEvent.ACTION\_UP)  
 {  
 if(view.getId() == R.id.red\_button)  
 {  
 buttonClick = 0;  
 }  
  
 if(plays[numOfClicks] != buttonClick)  
 {  
 playSound(lose);  
 return true;  
 }  
  
 playSound(bell);  
  
 if(numOfBlocksToClick == numOfClicks)  
 {  
 score++;  
 scoreText.setText("Score: " + score);  
 numOfClicks = 0;  
 if (numOfBlocksToClick > highestScore) {  
 highestScore = numOfBlocksToClick;  
 SharedPreferences highScores = getSharedPreferences("HIGHSCORE", Context.MODE\_PRIVATE);  
 SharedPreferences.Editor editor = highScores.edit();  
 editor.putInt("HIGHSCORE", highestScore);  
 editor.commit();  
  
 highestScoreText.setText("High score: " + highestScore);  
  
 }  
 final Runnable runnable = new Runnable() {  
 public void run() {  
 playGame();  
 }  
 };  
 handler.postDelayed(runnable, 1500);  
 }  
 }  
 return true;  
 }  
 });  
  
 findViewById(R.id.green\_button).setOnTouchListener(new View.OnTouchListener() {  
 @Override  
 public boolean onTouch(View view, MotionEvent motionEvent) {  
 if(motionEvent.getAction() == MotionEvent.ACTION\_UP)  
 {  
 if(view.getId() == R.id.green\_button)  
 {  
 buttonClick = 1;  
 }  
 if(plays[numOfClicks] != buttonClick)  
 {  
 playSound(lose);  
 return true;  
 }  
 playSound(ding);  
 if(numOfBlocksToClick == numOfClicks)  
 {  
 score++;  
 scoreText.setText("Score: " + score);  
 numOfClicks = 0;  
 if (numOfBlocksToClick > highestScore) {  
 highestScore = numOfBlocksToClick;  
 SharedPreferences highScores = getSharedPreferences("HIGHSCORE", Context.MODE\_PRIVATE);  
 SharedPreferences.Editor editor = highScores.edit();  
 editor.putInt("HIGHSCORE", highestScore);  
 editor.commit();  
  
 highestScoreText.setText("High score: " + highestScore);  
  
 }  
 final Runnable runnable = new Runnable() {  
 public void run() {  
 playGame();  
 }  
 };  
 handler.postDelayed(runnable, 1500);  
 }  
 }  
 return true;  
 }  
 });  
  
 findViewById(R.id.blue\_button).setOnTouchListener(new View.OnTouchListener() {  
 @Override  
 public boolean onTouch(View view, MotionEvent motionEvent) {  
 if(motionEvent.getAction() == MotionEvent.ACTION\_UP)  
 {  
 if(view.getId() == R.id.blue\_button)  
 {  
 buttonClick = 2;  
 }  
 if(plays[numOfClicks] != buttonClick)  
 {  
 playSound(lose);  
 return true;  
 }  
 playSound(dong);  
 if(numOfBlocksToClick == numOfClicks)  
 {  
 score++;  
 scoreText.setText("Score: " + score);  
 numOfClicks = 0;  
 if (numOfBlocksToClick > highestScore) {  
 highestScore = numOfBlocksToClick;  
 SharedPreferences highScores = getSharedPreferences("HIGHSCORE", Context.MODE\_PRIVATE);  
 SharedPreferences.Editor editor = highScores.edit();  
 editor.putInt("HIGHSCORE", highestScore);  
 editor.commit();  
  
 highestScoreText.setText("High score: " + highestScore);  
  
 }  
 final Runnable runnable = new Runnable() {  
 public void run() {  
 playGame();  
 }  
 };  
 handler.postDelayed(runnable, 1500);  
 }  
 }  
 return true;  
 }  
 });  
  
 findViewById(R.id.yellow\_button).setOnTouchListener(new View.OnTouchListener() {  
 @Override  
 public boolean onTouch(View view, MotionEvent motionEvent) {  
 if(motionEvent.getAction() == MotionEvent.ACTION\_UP)  
 {  
 if(view.getId() == R.id.yellow\_button)  
 {  
 buttonClick = 3;  
 }  
  
 if(plays[numOfClicks] != buttonClick)  
 {  
 playSound(lose);  
 return true;  
 }  
 playSound(high\_ding);  
 if(numOfBlocksToClick == numOfClicks)  
 {  
 score++;  
 scoreText.setText("Score: " + score);  
 numOfClicks = 0;  
 if (numOfBlocksToClick > highestScore) {  
 highestScore = numOfBlocksToClick;  
 SharedPreferences highScores = getSharedPreferences("HIGHSCORE", Context.MODE\_PRIVATE);  
 SharedPreferences.Editor editor = highScores.edit();  
 editor.putInt("HIGHSCORE", highestScore);  
 editor.commit();  
  
 highestScoreText.setText("High score: " + highestScore);  
  
 }  
 final Runnable runnable = new Runnable() {  
 public void run() {  
 playGame();  
 }  
 };  
 handler.postDelayed(runnable, 1500);  
 }  
 }  
 return true;  
 }  
 });\*/* **private void** playSound(**int** soundId) {  
 **if** (**soundsLoaded**.contains(soundId)) {  
 **soundPool**.play(soundId, 1.0f, 1.0f, 0, 0, 1.0f);  
 }  
 }  
  
  
**private int j**;  
 **private void** playGame() {  
 */\*final int bell = soundPool.load(this, R.raw.bell, 1);  
 final int ding = soundPool.load(this, R.raw.ding, 1);  
 final int dong = soundPool.load(this, R.raw.dong, 1);  
 final int high\_ding = soundPool.load(this, R.raw.high\_ding, 1);  
 final int lose = soundPool.load(this, R.raw.lose, 1);\*/* findViewById(R.id.***red\_button***).setEnabled(**true**);  
 findViewById(R.id.***green\_button***).setEnabled(**true**);  
 findViewById(R.id.***blue\_button***).setEnabled(**true**);  
 findViewById(R.id.***yellow\_button***).setEnabled(**true**);  
  
 **for** (**int** i = 0; i < 20; i++) {  
 **if** (**plays**[i] == 0) *// assign button a number between 1-4* {  
 *//plays[i] = random.nextInt(4); // 0 - 3* **plays**[i] = **random**.nextInt(4) + 1; *// 1 -4* Log.*i*(**"RANDOM"**, **""** + **plays**[i] + **"\n"**);  
 **break**; *// need this so it goes back to the first button in order* }  
 }  
 **numOfBlocksToClick**++;  
  
 **for** (**j** = 0; **j** < **numOfBlocksToClick**; **j**++) {  
 **final int** newJ = **j**;  
 Log.*i*(**"J"**, **""** + **j**);  
  
 *//Log.i("newJ", "" + newJ);* Log.*i*(**"Num of Blocks to Click"**, **""** + **numOfBlocksToClick**);  
 **final** Runnable runnable = **new** Runnable() {  
 @Override  
 **public void** run() {  
  
 **if** (**plays**[newJ] == 1)  
 {  
 playSound(**bell**);  
 */\*new Handler().postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 findViewById(R.id.red\_button).performClick();  
 }  
 }, 1000);\*/  
 // when button auto clicks, it will animate* **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***red\_button***).startAnimation(**animation**);  
  
 *//Toast.makeText(getApplicationContext(), "Red!", Toast.LENGTH\_SHORT).show();* findViewById(R.id.***red\_button***).performClick();  
  
 }  
 **else if** (**plays**[newJ] == 2)  
 {  
 playSound(**ding**);  
 */\*new Handler().postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 findViewById(R.id.green\_button).performClick();  
 }  
 }, 1000);\*/  
 // when button auto clicks, it will animate* **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***green\_button***).startAnimation(**animation**);  
  
 *//Toast.makeText(getApplicationContext(), "Green!", Toast.LENGTH\_SHORT).show();* findViewById(R.id.***green\_button***).performClick();  
  
 }  
 **else if** (**plays**[newJ] == 3)  
 {  
 playSound(**dong**);  
 */\*new Handler().postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 findViewById(R.id.blue\_button).performClick();  
 }  
 }, 1000);\*/  
 // when button auto clicks, it will animate* **animation**.setDuration(300);  
 **animation**.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***blue\_button***).startAnimation(**animation**);  
  
 *//Toast.makeText(getApplicationContext(), "Blue!", Toast.LENGTH\_SHORT).show();* findViewById(R.id.***blue\_button***).performClick();  
  
 }  
 **else if** (**plays**[newJ] == 4)  
 {  
 playSound(**high\_ding**);  
 */\*new Handler().postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 findViewById(R.id.yellow\_button).performClick();  
 }  
 }, 1000);\*/  
 // when button auto clicks, it will animate* Animation animation = **new** AlphaAnimation(1, 0);  
 animation.setDuration(300);  
 animation.setInterpolator(**new** LinearInterpolator());  
 findViewById(R.id.***yellow\_button***).startAnimation(animation);  
  
 *//Toast.makeText(getApplicationContext(), "Yellow!", Toast.LENGTH\_SHORT).show();* findViewById(R.id.***yellow\_button***).performClick();  
 }  
  
 }  
  
 };  
 **handler**.postDelayed(runnable, (1000) \* **j**);  
 }  
 }  
}