**package** com.google.engedu.wordstack;  
  
**import** android.content.res.AssetManager;  
**import** android.graphics.Color;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.DragEvent;  
**import** android.view.MotionEvent;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.LinearLayout;  
**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.util.ArrayList;  
**import** java.util.EmptyStackException;  
**import** java.util.HashSet;  
**import** java.util.Random;  
**import** java.util.Stack;  
  
**import static** android.R.attr.***x***;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 **private static final int** WORD\_LENGTH = 5;  
 **public static final int** LIGHT\_BLUE = Color.rgb(176, 200, 255);  
 **public static final int** LIGHT\_GREEN = Color.rgb(200, 255, 200);  
 **private** ArrayList<String> words = **new** ArrayList<>();  
 **private** Random random = **new** Random();  
 **private** StackedLayout stackedLayout;  
 **private** Stack<LetterTile> placedTiles;  
 **private** String word1, word2, playerWord1, playerWord2;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 AssetManager assetManager = getAssets();  
 **try** {  
 InputStream inputStream = assetManager.open(**"words.txt"**);  
 BufferedReader in = **new** BufferedReader(**new** InputStreamReader(inputStream));  
 String line = **null**;  
 **while**((line = in.readLine()) != **null**) {  
 String word = line.trim();  
 **if** (word.length() == WORD\_LENGTH)  
 words.add(word);  
 */\*\*my code above\*/* }  
 } **catch** (IOException e) {  
 Toast toast = Toast.makeText(**this**, **"Could not load dictionary"**, Toast.LENGTH\_LONG);  
 toast.show();  
 }  
 LinearLayout verticalLayout = (LinearLayout) findViewById(R.id.vertical\_layout);  
 stackedLayout = **new** StackedLayout(**this**);  
 verticalLayout.addView(stackedLayout, 3);  
  
 View word1LinearLayout = findViewById(R.id.word1);  
 *//word1LinearLayout.setOnTouchListener(new TouchListener());* word1LinearLayout.setOnDragListener(**new** DragListener());  
 View word2LinearLayout = findViewById(R.id.word2);  
 *//word2LinearLayout.setOnTouchListener(new TouchListener());* word2LinearLayout.setOnDragListener(**new** DragListener());  
  
 placedTiles = **new** Stack<>();  
 playerWord1 = **""**;  
 playerWord2 = **""**;  
 }  
  
 **private class** TouchListener **implements** View.OnTouchListener {  
  
 @Override  
 **public boolean** onTouch(View v, MotionEvent event) {  
  
  
 **if** (event.getAction() == MotionEvent.ACTION\_DOWN && !stackedLayout.empty()) {  
 LetterTile tile = (LetterTile) stackedLayout.peek();  
 tile.moveToViewGroup((ViewGroup) v);  
 **if** (stackedLayout.empty()) {  
 TextView messageBox = (TextView) findViewById(R.id.message\_box);  
 messageBox.setText(word1 + **" "** + word2);  
 }  
  
 placedTiles.push(tile);  
  
 **return true**;  
 }  
 **return false**;  
 }  
 }  
  
 **private class** DragListener **implements** View.OnDragListener {  
  
 **public boolean** onDrag(View v, DragEvent event) {  
 **int** action = event.getAction();  
 **switch** (action) {  
 **case** DragEvent.ACTION\_DRAG\_STARTED:  
 v.setBackgroundColor(LIGHT\_BLUE);  
 v.invalidate();  
 **return true**;  
 **case** DragEvent.ACTION\_DRAG\_ENTERED:  
 v.setBackgroundColor(LIGHT\_GREEN);  
 v.invalidate();  
 **return true**;  
 **case** DragEvent.***ACTION\_DRAG\_EXITED***:  
 v.setBackgroundColor(***LIGHT\_BLUE***);  
 v.invalidate();  
 **return true**;  
 **case** DragEvent.***ACTION\_DRAG\_ENDED***:  
 v.setBackgroundColor(Color.***WHITE***);  
 v.invalidate();  
 **return true**;  
 **case** DragEvent.***ACTION\_DROP***:  
 *// Dropped, reassign Tile to the target Layout* LetterTile tile = (LetterTile) event.getLocalState();  
 **if**(v.getId() == R.id.***word1***)  
 **playerWord1** += tile.moveToViewGroup((ViewGroup) v);  
 **else  
 playerWord2** += tile.moveToViewGroup((ViewGroup) v);  
 **if** (**stackedLayout**.empty()) {  
 checkWin();  
 }  
 **placedTiles**.push(tile);  
 **return true**;  
 }  
 **return false**;  
 }  
 }  
  
 **protected boolean** onStartGame(View view) {  
  
 ViewGroup word1LinearLayout = (ViewGroup)findViewById(R.id.***word1***);  
 ViewGroup word2LinearLayout = (ViewGroup)findViewById(R.id.***word2***);  
  
 word1LinearLayout.removeAllViews();  
 word2LinearLayout.removeAllViews();  
 **try** {  
 stackedLayout.clear();  
 } **catch**(EmptyStackException e){}  
  
 TextView messageBox = (TextView) findViewById(R.id.message\_box);  
 messageBox.setText(**"Game started"**);  
  
 **int** index1 = random.nextInt(words.size());  
 **int** index2;  
 **do** {  
 index2 = random.nextInt(words.size());  
 }**while**(index2 == index1);  
  
 word1 = words.get(index1);  
 word2 = words.get(index2);  
  
*// word1 = "dates";  
// word2 = "loved";* String word3 = **""**;  
 **int** word1Count = 0;  
 **int** word2Count = 0;  
 **while**(word1Count < WORD\_LENGTH || word2Count < WORD\_LENGTH){  
 **if**(random.nextInt(2) == 1 && word1Count < WORD\_LENGTH) {  
 word3 += word1.charAt(word1Count);  
 word1Count++;  
 }  
 **else if** (word2Count < WORD\_LENGTH) {  
 word3 += word2.charAt(word2Count);  
 word2Count++;  
 }  
  
 }  
  
 *//messageBox.setText(word3);* **for**(**int** i = word3.length()-1; i >= 0; --i){  
 stackedLayout.push(**new** LetterTile(**this**, word3.charAt(i)));  
 }  
  
 **return true**;  
 }  
  
 **protected boolean** onUndo(View view) {  
  
 **if**(!placedTiles.isEmpty()) {  
  
 **if** (((View)placedTiles.peek().getParent()).getId() == R.id.word1){  
 playerWord1 = **new** StringBuilder(playerWord1).deleteCharAt(playerWord1.length()-1).toString();  
 placedTiles.pop().moveToViewGroup(stackedLayout);  
 }  
 **else** {  
 playerWord2 = **new** StringBuilder(playerWord2).deleteCharAt(playerWord2.length()-1).toString();  
 placedTiles.pop().moveToViewGroup(stackedLayout);  
 }  
 }  
  
 **return true**;  
 }  
  
 **protected void** checkWin() {  
  
 TextView messageBox = (TextView) findViewById(R.id.message\_box);  
 **if**(word1.equals(playerWord1) && word2.equals(playerWord2))  
 messageBox.setText(**"You win! "** + word1 + **" "** + word2);  
 **else if**(words.contains(playerWord1) && words.contains(playerWord2)){  
 messageBox.setText(**"You found alternative words! "** + playerWord1 + **" "** + playerWord2);  
 }  
 **else**{  
 messageBox.setText(**"Try again"**);  
 }  
 }  
}