Piñata Game: Keeping Score in Alice

By Maggie Bashford
Professor Susan Rodger
Duke University
July 2009
www.cs.duke.edu/ced/alice/aliceInSchools

Adding the Text Object

Click add objects and add a new 3D Text Object to the world.

When it asks what to say, type score: 0.0.

Adding the Text Object

Download the Alice World called piñataGameStart.a2w that goes along with this tutorial.

We will use this world to create a game that allows players to hit a swinging piñata with a bat. Many of the objects have already been created and placed in the world. We must create an object to display and update the score, and we will also write the code to play the game.

Click add objects and add a new 3D Text Object to the world.

When it asks what to say, type score: 0.0.

Once you have created the new text object, right click on it in the object tree and rename it score.

Reposition the score where you can see it. Use resize and turn to face the camera.
Creating Variables

Click on `score` in your object tree, and then go to the `properties` tab. Click on the `create new variable` button.

Name the variable `scoreValue`, select the type `Number`, and click the arrow next to value to select `other` and type `0`.

Add Points Method

Now we need to write a method that will add one point to `scoreValue` variable when the piñata is hit. Create another class-level method, called `addPoints`.

Find `scoreValue` in the `properties` tab and drag it into the method, and choose to `increment` it by 1. Change the `duration` to 0 seconds.

Creating Variables

We need to keep track of one more thing during our game. Go to the `world properties` tab and notice two existing variables. The first is a list called `pinataParts`. It is used in the `explode` method to break open the piñata. There is also a variable called `time` that sets the number of seconds given to play the game. You can change it by clicking the arrow next to the value 15.

Now let's create a variable to remember the number of hits needed to break the piñata. Create a `world variable` called `numWinningHits`, and set its value to 10. We will use this later to determine if the player has won the game.

Add Points Method

Next drag `text` into the `addPoints` method, set its value to "score:" and set its duration to 0 seconds.

Next, click on `world` in the object tree, and find its `functions` tab.

Find `a` joined with `b` and drag it into where you typed "score:" and select `default string` for `b`.

Then drag what as a string into default string, and select `score.scoreValue` from expressions.
Set Up Method

Open `setUp` under `world methods`. Try playing it with the `When the world starts` event, but remember to set it back to `my first method` when you are done!

```
world setup: No parameters

// No variables.
```

Instructions:
- Set the option to `true`.
- Duration = 2 seconds.
- More...

```
instructions: setListening to true - duration = 2 seconds - more...
```

```
Do in order:
```

### Instructions 1

- Set the option to `true`.
- Duration = 2 seconds.
- More...

```
instructions: set to true - duration = 2 seconds - more...
```

### Instructions 2

- Set to true.
- Duration = 2 seconds.
- More...

```
instructions: set true - duration = 2 seconds - more...
```

- Set the option to `false`.
- Duration = 0 seconds.
- More...

```
instructions: setListening to false - duration = 0 seconds - more...
```

- Initialize time = worldTime.

```
initialize time = worldTime.
```

Play Game Method

Open the completed `world method` called `playGame` and try to figure out what it does.

```
world play game: No parameters

// No variables.
```

```
instructions: setListening to false - duration = 0 seconds - more...
```

Type in order:

- Check the true or false.
- Duration = 0 seconds.
- More...

```
instructions: check true or false - duration = 0 seconds - more...
```

- Check the true or false.
- Duration = 0 seconds.
- More...

```
instructions: check true or false - duration = 0 seconds - more...
```

- Check the true or false.
- Duration = 0 seconds.
- More...

```
instructions: check true or false - duration = 0 seconds - more...
```

```
initialize time = worldTime.
```

Game Over Method

Now open the `gameOver` method. A text object called `gameOverText` appears when the player wins, and it has been set to say “¡Felicidades!”.

The text changes to “¡Qué lástima!” if the player loses. Notice that there is no condition in the `if/else` statement. We need to create a condition that will check if the player won.

```
world gameOver: No variables

// No variables.
```

Instructions:
- Set the option to `false`.
- Duration = 0 seconds.
- More...

```
instructions: set true or false to false - duration = 0 seconds - more...
```

Game Over Text

If you don't like "¡Felicidades!" (which means "Congratulations!" in Spanish), you can change the text and color in the `gameOverText` properties.

Typing Accents

You can change the text that appears when the game is over. Here are the keyboard codes for typing accents.

PC: (use the keypad and make sure you have num lock on)

- á = Alt + 0225
- é = Alt + 0233
- í = Alt + 0237
- ó = Alt + 0243
- ù = Alt + 0250
- ñ = Alt + 0241
- ñ = Option + n, then n
- ñ = Alt + 0161
- ¿ = Option + 1
- ¿ = Option + Shift + ?

Mac:

- á = Option + e, then a
- é = Option + e, then e
- í = Option + e, then i
- ó = Option + e, then o
- ù = Option + e, then u
- ñ = Option + n, then n
- ñ = Option + 1
- ¿ = Option + Shift + ?

These are just some of the many different foreign characters you can create. Try searching for "keyboard accent codes" to find more.
Game Over Method

We need the condition to check if the player’s score is greater than the required number of hits. Go to the world functions and drag \( a \geq b \) into the true of the if condition.

From expressions, select `score.scoreValue` then `world.numWinningHits`.

Bat Hit Method

Next we need to drag in one line of code to finish the `batHit` method. Click on `score` in the object tree, then drag the `addPoints` method into the Do Nothing of the if statement. This will give the player points only when the bat has hit the piñata.

Bat Hit Method

If you can’t see the entire `batHit` method, scroll to the right. Try to figure out what the If/Else statement is asking. You can also play around with the durations to change how quickly the bat swings.

Events

Now we need to create two events: one to recognize when the player wins and another to swing the bat when the player presses the up key.

Click create new event in the top of the window, and select While something is true.

Right click on the blue area of the new event and select change to then When something becomes true.
Events

Go to the world functions and drag a>=b into the condition. Select score.scoreValue from expressions then world.numWinningHits from expressions.

Click on timer in the object tree then go to the properties tab. Drag timeLeft into the do Nothing and select set value to and choose 0. This will end the game when the player reaches the required number of hits.

Finishing Up

Now you are ready to play the game! Click play and try your best to break the piñata.

Keeping track of a score can be used in many games. Some games require a certain score to reach the next level. Others may keep track of the players' high scores.

You can create different events and methods to update your score. All you need to do is keep track of your score and update its display.