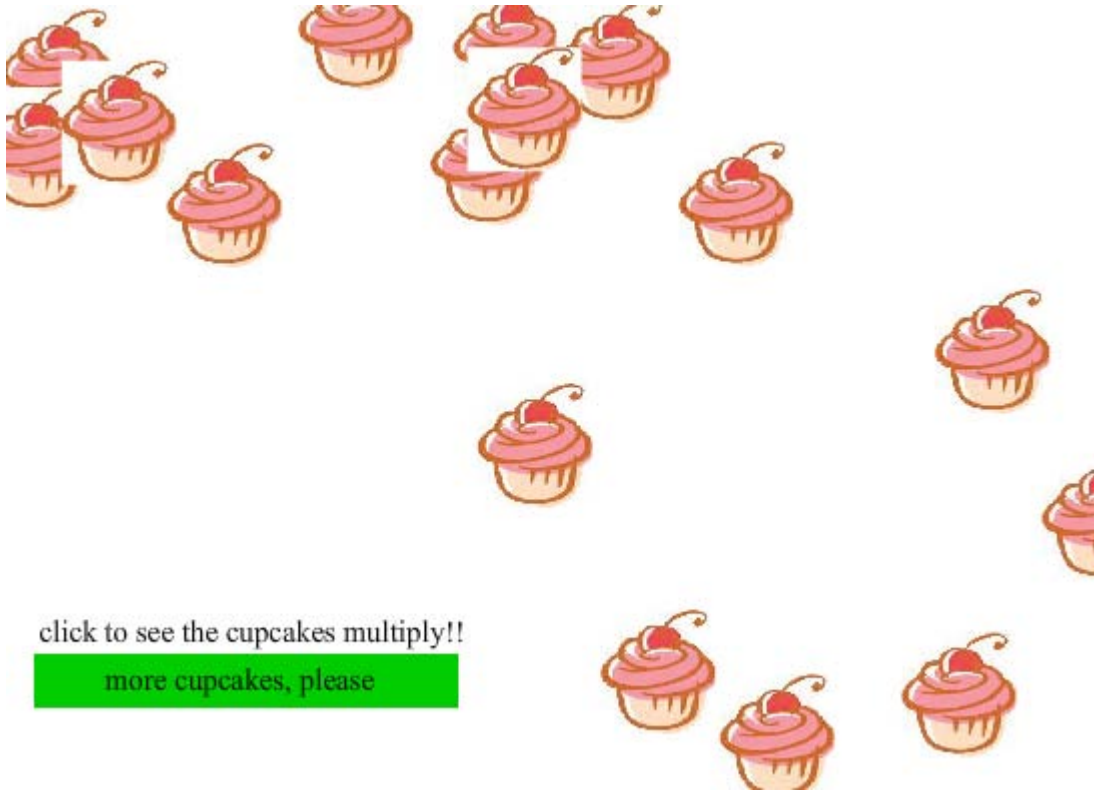


# Doubling Time World Population

---



click to see the cupcakes multiply!!

more cupcakes, please

## Contents

Overview .....	3
Graphics to develop .....	3
For level one.....	3
For level two .....	3
Level One Application .....	<b>Error! Bookmark not defined.</b>
Level Two Application .....	4
Level Three Application.....	6

## Overview

A key notion in biology and ecology is doubling time. How long does it take a given population to double in size? The doubling time for the human population on Earth is one of the most prominent features of the past few centuries. Whereas thousands of years characterized doubling times for many thousands of years, since 1900 the human population had doubled twice, first from 1 to 2 billion, then from 2 to 4 billion. During the early decades of the twenty-first century, it is expected to double once again.

Check out this video for visualization before beginning:

<http://www.youtube.com/watch?v=4BbkQiQyaYc>

## Graphics to develop

Check out the resource folder for starter files.

### *For level one*

A picture of a cupcake.

### *For level two*

Just the continents alone in black color with a white or transparent background (the color of the continent can be changed later as you explore the color transformations in the code)

## Level One Application

Create a new directory for Lesson 6 – Population Doubling. Open a new .fla file and save it in that directory as doubling01.

### Set Up

Create two layers in the timeline: display and code. Find a cool picture of a cupcake and import into the library, make sure it is a movie clip and its name is cupcake2.

### Display Layer

Make a button and pull an instance of it on to the screen – instance name of “doublingBtn.” Above it put a description for the user “click to see the cupcakes multiply!!” And on the button “more cupcakes, please.”

### Code Layer

First we need to add an event listener to the button:

```
doublingBtn.addEventListener(MouseEvent.CLICK, calcCupcakes);
```

Next we need some variable definitions:

```
var cupcakes:Number = 1;
```

```
var cake:cupcake2 = new cupcake2();
```

```
var cakes:Array = new Array();
```

Now, we can add a function to double the cupcakes:

```
function calcCupcakes(event:MouseEvent):void {
    cupcakes=cupcakes*2;
    trace("number of cupcakes: " + cupcakes);
}
```

Save and run the application. You should be able to see the output that shows how that cupcakes are doubling.

Finally, we can add a for statement that will add the number of cupcakes to the stage. Place this code inside the calcCupcakes function:

```
for (var i:int=0; i<cupcakes; i++) {
    cakes.push(new cupcake2());

    cakes[i].x=Math.random()*550;
    cakes[i].y=Math.random()*400;

    addChildAt(cakes[i],0);
}
```

Save and run. Mmmm, more cupcakes with each click!

## Level Two Application

### Set Up

Open a new .fla file and save it to the same directory as doubling02.fla. Create two layers: display and code.

### Display Layer

Begin by copying the button from level one into the library of this .fla file. Pull an instance of the button on the stage and call it “doublingBtn.” Over it, place static text that reads: click to calculate!

Next, create four static text boxes, three input text boxes and one input text box as follows:

base population

Input text box with instance name of "basePopulation"

number of years that will pass

Input text box with the instance name of "timetoPass"

current doubling time

Input text box with the instance name of "doublingTime"

Dynamic text box with instance name of "newPopulation"

click to calculate!

the pouplation will be....

Finally, import the graphic you made of the world map into the library or use the starter one in the Resource library and pull an instance of it onto the stage; call it worldMap.  
Save and run the application to make sure everything looks good.

### Code Layer

Place the following code in the first time frame:

```
doublingBtn.addEventListener(MouseEvent.CLICK, calcnewPopulation);
```

```
function calcnewPopulation(event:MouseEvent) :void {
    var basePopulation:Number=Number(basePopulation.text);
    var timet:Number=Number(timetoPass.text);
    var doublingt:Number=Number(doublingTime.text);
    var newvalue:Number;

    //insert equation here
}
```

Save and test the application by entering values in the boxes.

Next, we will add a color transformation to the world map. Add another event listener to the button with this line:

```
doublingBtn.addEventListener(MouseEvent.CLICK, makeAnimation1);
```

Add two new functions at the end of the code:

```
// This line defines a variable of type ColorTransform and naming it colorTransform
```

```
var colorTransform:ColorTransform = new ColorTransform();
```

```
function makeAnimation1(event:Event):void {
    colorTransform.redOffset=-255;
    this.addEventListener(Event.ENTER_FRAME, makeAnimation);
}
```

```
function makeAnimation(event:Event):void {
    colorTransform.redOffset+=10;
    worldMap.transform.colorTransform=colorTransform;
}
```

Functions explained:

-The line `colorTransform.redOffset=-255`

We set it to -255 because the values of the colors go from -255 to +255, so the transition will be adding 10 to the value -255 and so on until we get 255 which is max.

-The line `this.addEventListener(Event.ENTER_FRAME, makeAnimation)`

This adds an eventlistener to the stage, an enterframe event, to call the function `makeAnimation`. (Remember the `enter_frame` event is called as many times per second as the fps is set to in the flash settings (usually 12 frames per second)).

-The line `colorTransform.redOffset+=10`

This is what happens in the animation, first we add 10 to the redoffset value, this is done 12 times a second.

-The line `worldMap.transform.colorTransform=colorTransform`

This "attaches" the `colortransform` to our object (`worldMap`), without this there will be no effect

Save and run the completed application. The world map should color transform into red – as a visual representation of population growth.

## Level Three Application

A logistical growth model assumes that population growth gradually slows as the population approaches the carrying capacity.

When the population is small relative to the carrying capacity, logistical growth is exponential with a fractional growth rate close to the base growth rate  $r$ . As the population approaches the carrying capacity, the logistical growth rate approaches zero. The logistical growth rate at any particular time

depends on the population at that time, the carrying capacity, and the base growth rate  $r$  (Bennet and Briggs 539).

### Set Up

Save the population02 fla as population03 fla in the same directory.

Look at your math opportunity worksheet for the equation and change the boxes as well as the function. (Demo application on the website).

If Earth's carrying capacity is estimated at 12 billion people, can you create a calculation (much like level two) to find the logistical growth rate? What would it mean if the carrying capacity of Earth was 40 billion, or 4 billion?