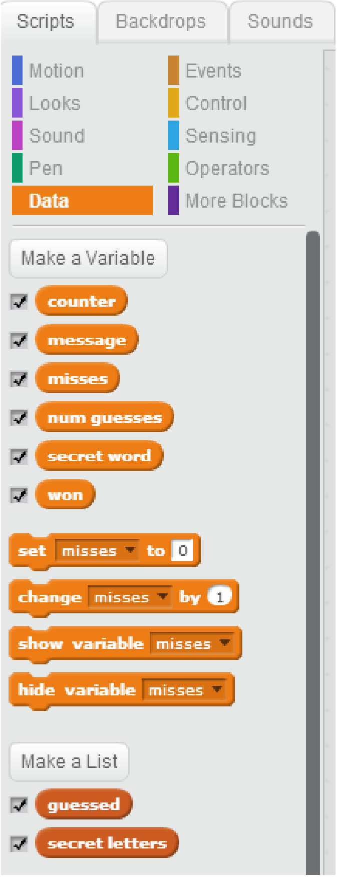
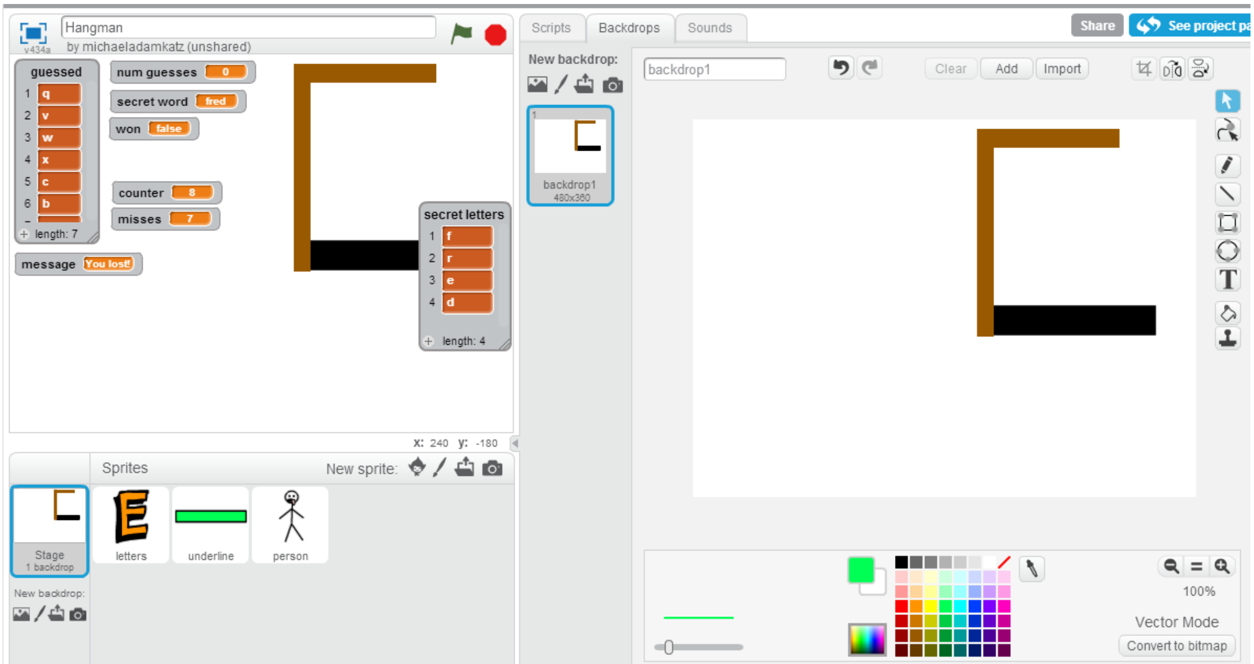
Hangman, er, Hangperson, er, Nonviolent progressive word game

1. Programmers often think about variables first. "What variables do I need to keep track of the state of this game?" This project uses six normal variables, and two lists. Lists are really just a kind of variable that can hold a list of values in one variable.



2. The stage.

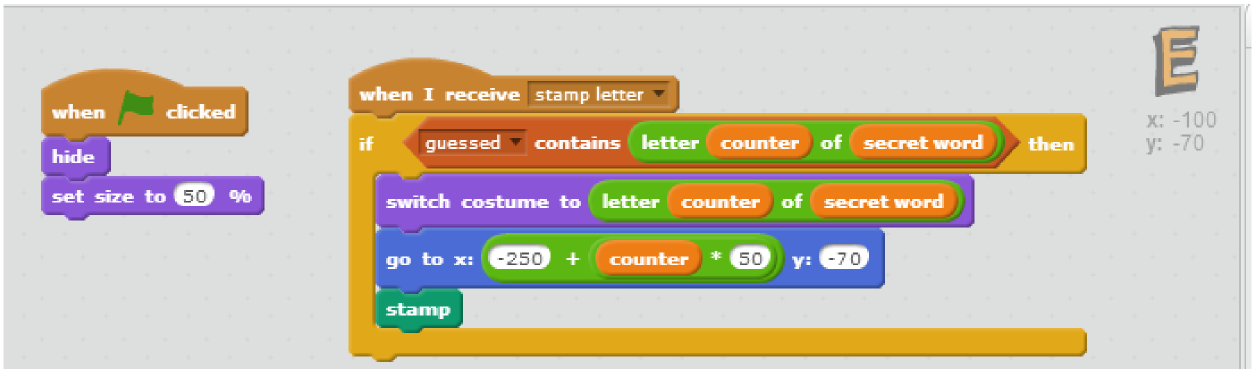
   Here is a picture that shows the stage, which has one simple backdrop. You can also see that there are three sprites. You can also see how I've laid out the display of the variables, so you can see what values they contain, but can still play the game.



```
Hangman
by michaeladamkatz (unshared)

guessed: 1 q, 2 v, 3 w, 4 x, 5 c, 6 b   length: 7
num guesses: 0
secret word: fred
won: false
counter: 8
misses: 7
message: You lost!
secret letters: 1 f, 2 r, 3 e, 4 d   length: 4
```

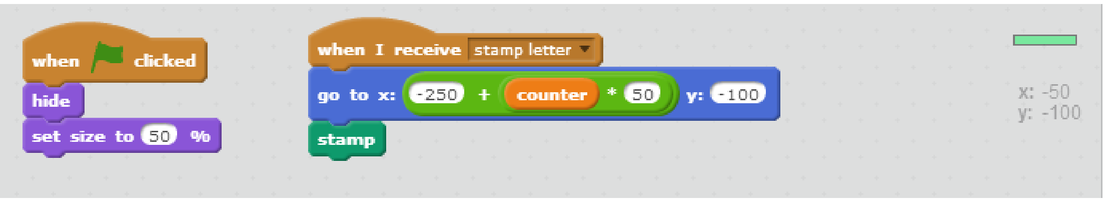
3. The "letters" sprite already has all of its costumes (I did this for you because it is a time-consuming and not very interesting task). But you need to add the script:



```
when green flag clicked
hide
set size to (50) %

when I receive [stamp letter v]
if <[guessed v] contains (letter (counter) of (secret word))> then
    switch costume to (letter (counter) of (secret word))
    go to x: ((-250) + ((counter) * (50))) y: (-70)
    stamp
end
```

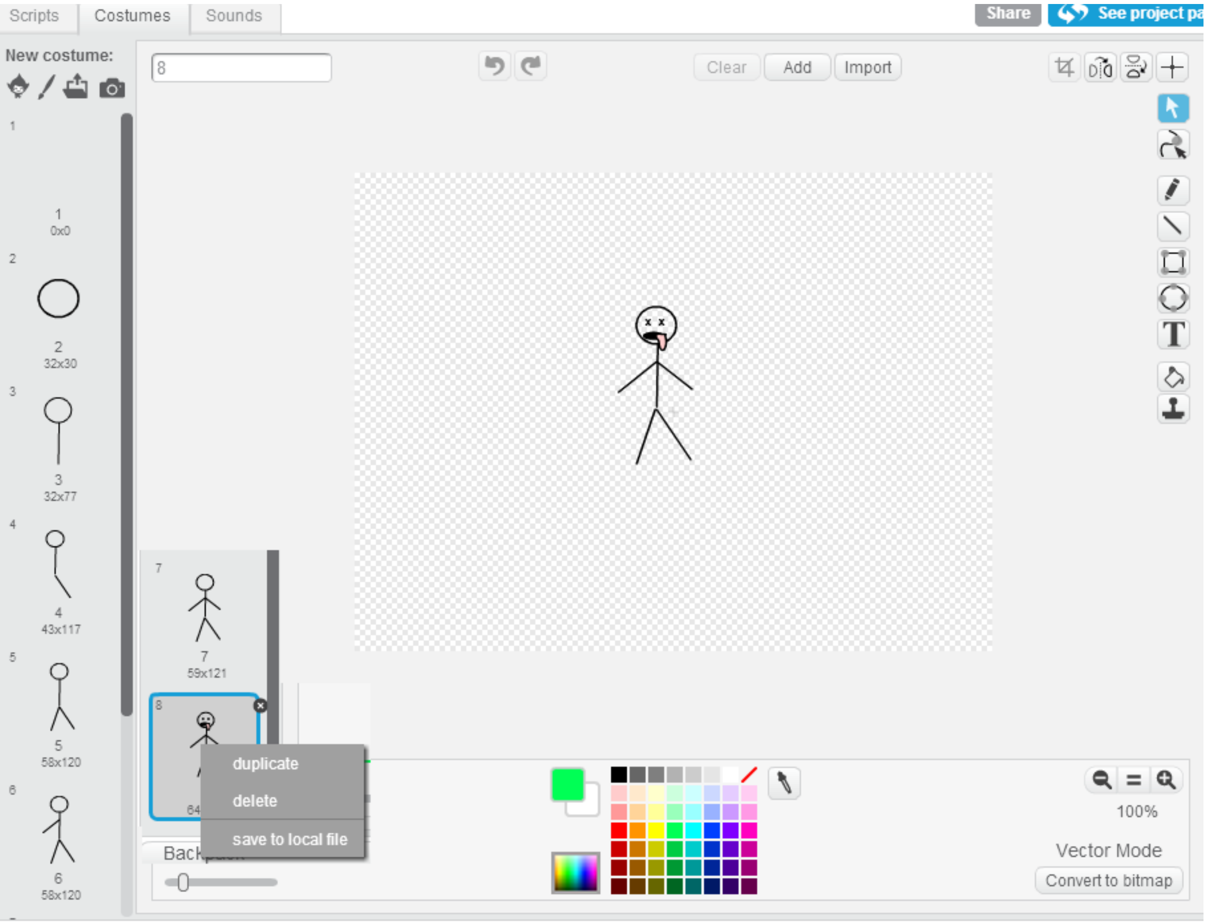
4. For the “underline” sprite, draw a little bar. Here is its script:



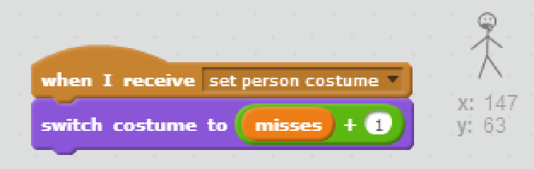
```
when [flag] clicked
hide
set size to (50) %
```

```
when I receive [stamp letter v]
go to x: ((-250) + ((counter) * (50))) y: (-100)
stamp
```

5. For the “person” sprite, make 8 costumes. Start by making costume “8”, which is the full person. Then use the “duplicate” (right-click”) costume function to make 7 copies, each time removing one more piece, until you get down to nothing for costume “1”. Make sure the sprites are named “1” through “8”. (Remember you don’t have to make a person. You can make some other picture that “builds up” or “falls apart”.)



Here is the very simple script for the “person” sprite.

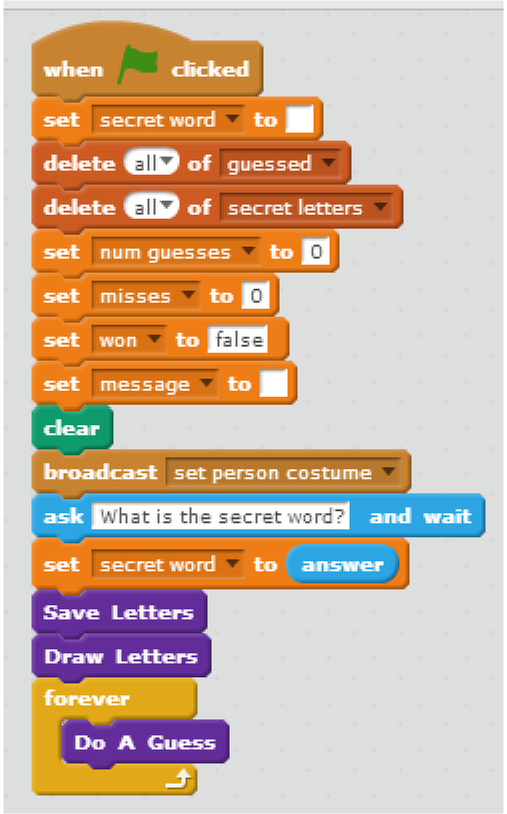


```
when I receive [set person costume v]
switch costume to ((misses) + (1))
```

6. Now let's go back to the Stage and add its script. The Stage has most of the script for this game.

   First, here is the code that starts the game by

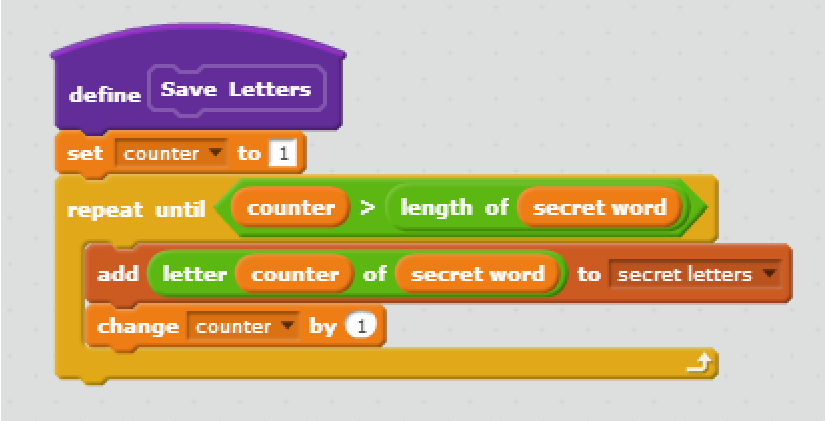
   - Initializing the variables
   - Clearing the drawing on the screen
   - Setting the person to its hidden costume
   - Asking for the secret work
   - Doing a "forever" loop to let the player guess until the game is either won or lost



```
when 🏁 clicked
set secret word to
delete all of guessed
delete all of secret letters
set num guesses to 0
set misses to 0
set won to false
set message to
clear
broadcast set person costume
ask What is the secret word? and wait
set secret word to answer
Save Letters
Draw Letters
forever
    Do A Guess
```

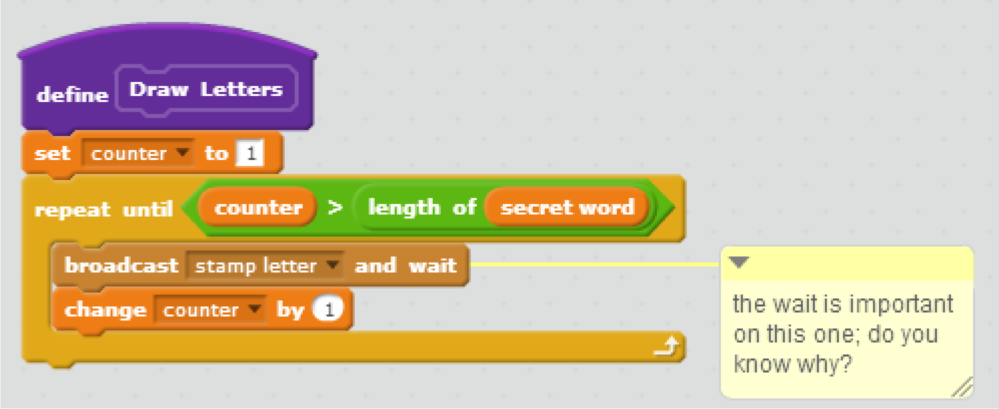
Next we use the **More Blocks** category to create the “Save Letters” and “Draw Letters” functions:

Save letters puts each letter of the secret word into the “secret letters” list. We need the letters in a list for the “Count Misses” function we’ll write soon.



```
define Save Letters
set counter to 1
repeat until (counter > length of secret word)
    add letter counter of secret word to secret letters
    change counter by 1
```

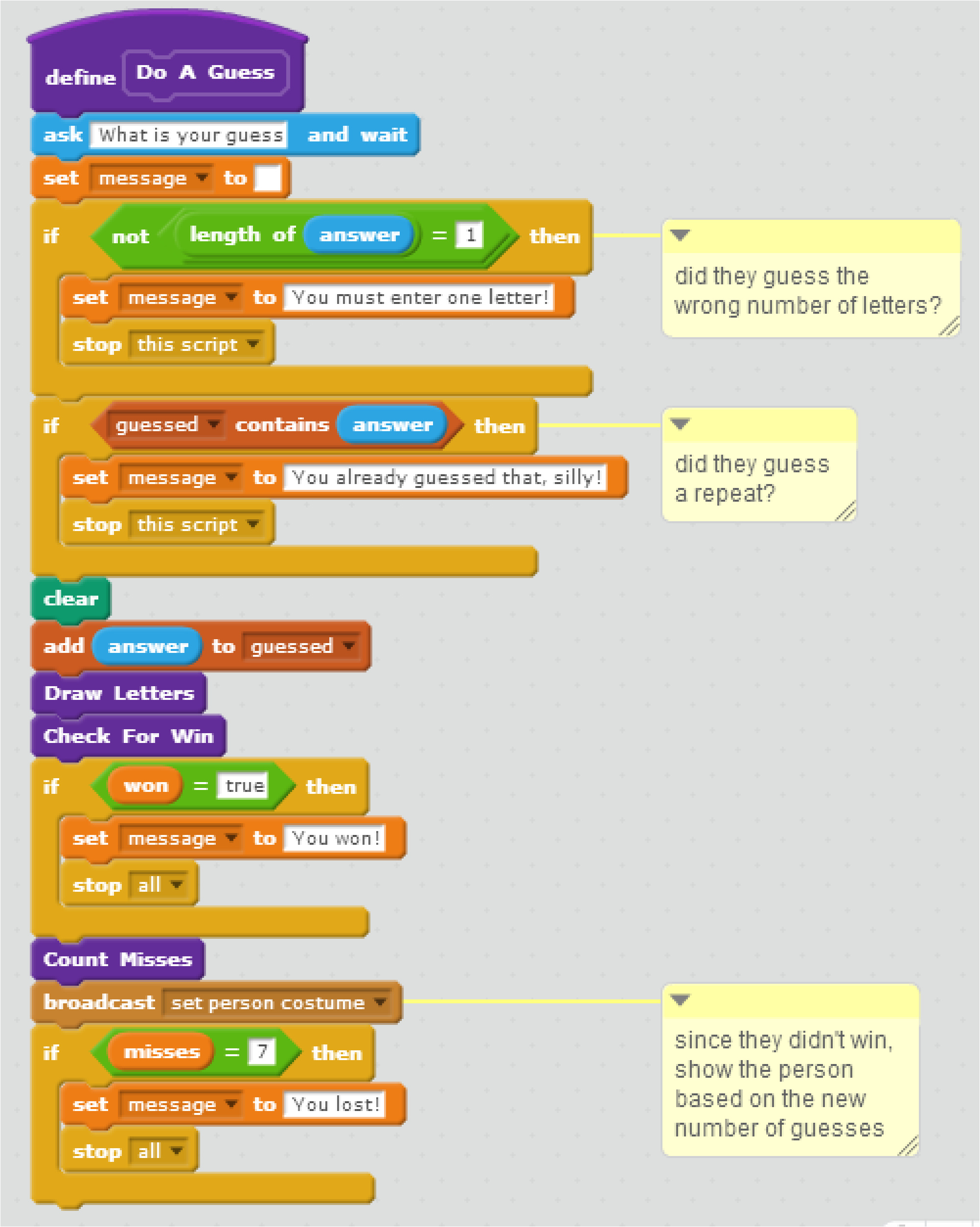
Draw letters tells the “letters” and “underline” sprites to stamp out each letter of the secret word. (Except the “letters” sprite only shows letters that have been guessed.)



```
define Draw Letters
set counter to 1
repeat until (counter > length of secret word)
    broadcast stamp letter and wait
    change counter by 1
```

the wait is important on this one; do you know why?

Next we write the long “Do A Guess” function. Don’t be scared of how long this function is. Read it section by section to see what each section does, and then see if you can understand each line:



```
define Do A Guess
ask What is your guess and wait
set message to 
if not length of answer = 1 then
    set message to You must enter one letter!
    stop this script
if guessed contains answer then
    set message to You already guessed that, silly!
    stop this script
clear
add answer to guessed
Draw Letters
Check For Win
if won = true then
    set message to You won!
    stop all
Count Misses
broadcast set person costume
if misses = 7 then
    set message to You lost!
    stop all
```

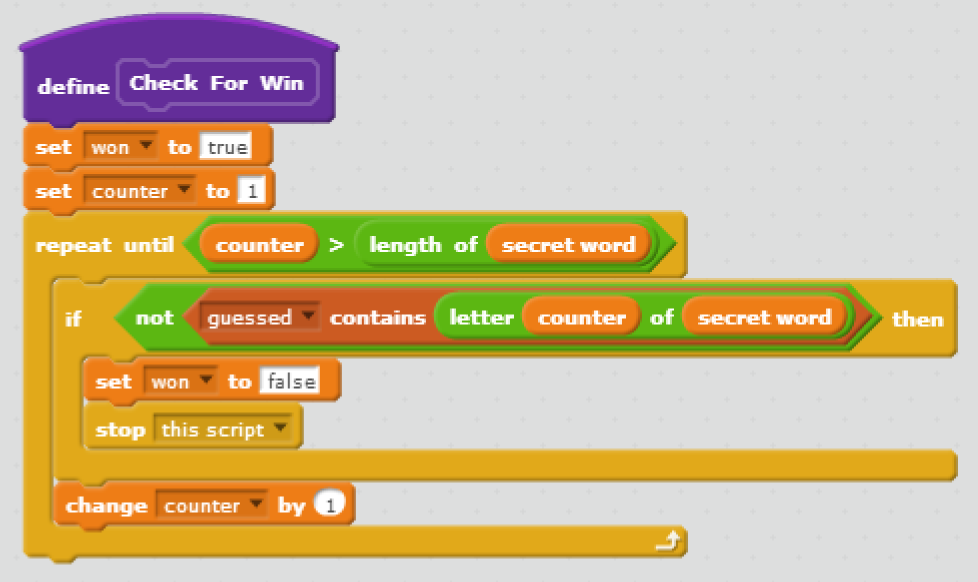
did they guess the wrong number of letters?

did they guess a repeat?

since they didn't win, show the person based on the new number of guesses

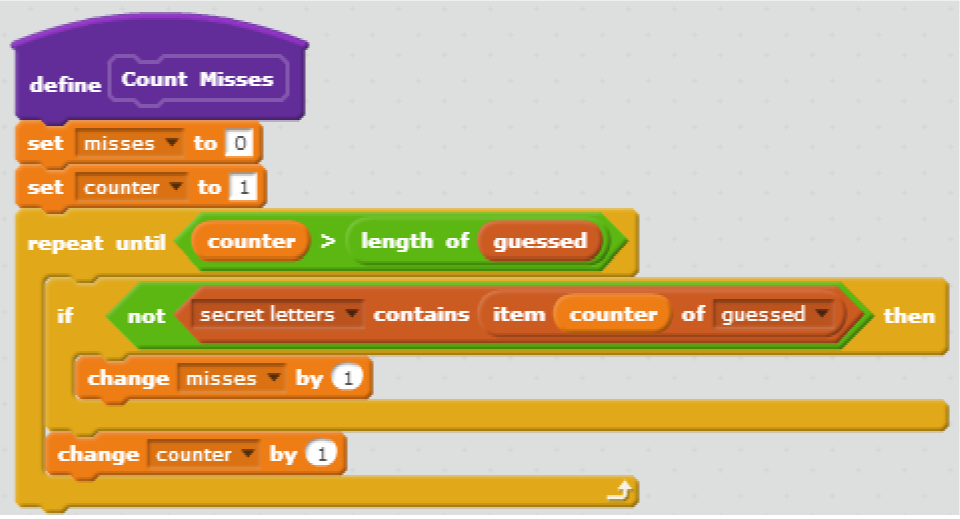
And finally we add the two remaining functions.

“Check For Win” sees if the player has won by seeing if there are in letters in the secret word that have not been guessed. (If there are any, the player has not won.)



```
define Check For Win
set won to true
set counter to 1
repeat until counter > length of secret word
    if not guessed contains letter counter of secret word then
        set won to false
        stop this script
    change counter by 1
```

“Count Misses” is used both for knowing which costume of the “person” sprite to show, and for checking if the player has lost.



```
define Count Misses
set misses to 0
set counter to 1
repeat until counter > length of guessed
    if not secret letters contains item counter of guessed then
        change misses by 1
    change counter by 1
```

Now you should actually be able to click the green flag to play the game. Remember that if it doesn’t work there are various debugging techniques to find out what’s wrong. Ask for help!