

1. Declare a variable (say, x) in the main body of your program. Then declare one inside of a for loop. Is there a conflict? Is the old variable overwritten, or do you now have two variables of the same name?

The old variable is overwritten.

2. What if the other x is inside of a function?

If another x is declared inside the function with the keyword var it is a unique variable to that function and cannot be accessed from main unless it's returned. This is dependent on if you use the keyword var to show the program that you are declaring a new variable.

3. Can you have variables that are globally accessible? What are the rules for creating them?

Variables declared without being nested in any functions or conditionals are global. So if you repeat what was done in question 2 without the keyword var it would override the old variable.

4. Are some variables passed by value while others are passed by reference? Which ones are which?

JavaScript passes by value with functions.

5. If you run this code (or the equivalent) in your language, what is the output? What does that tell you about how the language handles assignments?

```
char [] a = {'c','a','t'}  
char [] b = {'d','o','g'}  
a=b  
b[1] = 'u'  
print a  
print b
```

The output:

```
[ 'd', 'u', 'g' ]  
[ 'd', 'u', 'g' ]
```

Because the changes to b were applied to a after they were set as equal to each other, you can tell that the assignment doesn't set a's value to a copy of b's but rather changes a's reference to point to where b's value is stored.