



You only need to complete one problem (either 1-1 or 1-2) in order to receive full credit!

**1-1:** Count the number of slashes in a string:

Table tbl\_Pets contains a field named Colors containing multiple color values separated by slashes. You are task to write a procedure to count the number of colors.

**Steps:**

Create a new function named intCountColors in mdl\_Class2 and pass a string variable named strColors into the function.

Understand the problem you are trying to solve:

Using the InStr function to find the first occurrence of a slash in a given string. Use the immediate window to test InStr function so that you gain a better understanding.

```
Immediate
?InStr("Black/Blue/Green", "/" )
6
```

Once the first slash is found, how do we find the next one? The InStr function contains an optional argument, the start argument. This argument tells the InStr function from which position to start searching. So to find the second slash, start searching after the position of the first slash, therefore increment the previous slash position by one (6+1):

```
Immediate
?InStr(7, "Black/Blue/Green", "/" )
11
```

For any further slashes, continue this process until no more slash is found. In this case the Instr function returns 0 (see below). This is the condition to use to stop looping.

```
Immediate
?InStr(12, "Black/Blue/Green", "/" )
0
```



Write Pseudo Code:

Variables:

intPos → position of slash found

intCountSlashes → count of slashes

Set the intPos variable to 1.

Create a loop (For Next loop (use intCountSlashes as counter variable ,start at 0, maximum number of loops = 5)

Within each loop, using the InStr function with intPos as the start argument to determine the position of the slash and assign it into intPos

If intPos equals 0, then exit the for loop

Increment intPos by 1

End Loop

Assign intCountSlashes + 1 into intCountColors (0 slashes mean 1 color, 1 slash means 2 colors, etc.)

Test the function:

Test the function using the immediate window

Additional Assignment (optional, not required):

Replace the For Next loop with a Do Loop

**1-2:** Create a sub routine to determine the FollowUpDate of a given visit date and follow up type

Business Rule:

The veterinarian will decide whether a follow up date is needed, if yes, then the follow up date is determined based on the following rule:

Follow Up Type	Follow Up Date (time interval from visit date)
Illness, Injury	7 days
Physical, Routine	1 month
Grooming	90 days
All other	1 year

Name the sub routine FollowUpDate and save it in module mdl\_class2. There are two arguments needed, dteVisitDate (date)and strFollowUpType (string)).



The first thing is to ask the user whether a follow up date is needed, use the message box function as shown below:



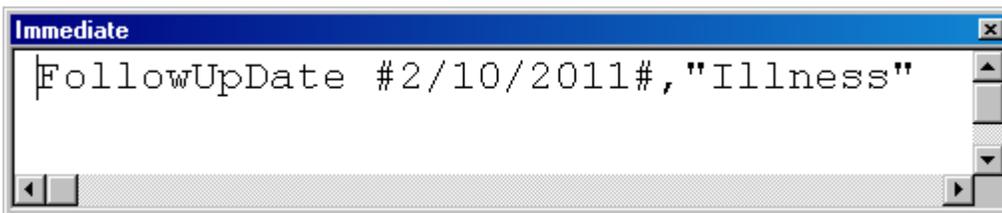
If the user clicks on Yes, then determine the follow up date based on the above rules (use DateAdd function). Display the follow up date in a message box statement as shown below:



If the user clicks on No, then display the following message box:

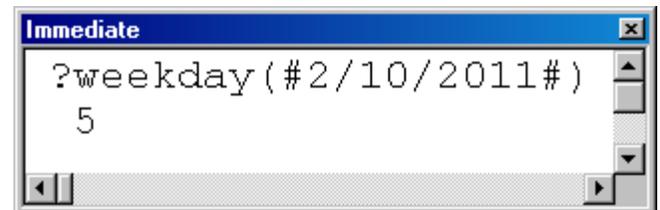


To test this procedure, use the Immediate Window as shown below:



Additional Assignment (optional, not required):

What if the follow up date falls on a weekend? Enhance this procedure by using the weekday function. The weekday function returns 1 for Sunday, 2 for Monday, ... 7 for Saturday.



The business rule is to move the follow up date to the next possible weekday.