Subject: Skip file during loop based on subprocess
Posted by Kai Heckel on Thu, 30 Jul 2015 11:55:00 GMT
View Forum Message <> Reply to Message

Hey there!

Here is what I have:

```
FOR f=0, N_ELEMENTS(foldername)-1 DO BEGIN<-----
  IF test EQ 1 THEN BEGIN                     -
    IF test EQ 1 THEN BEGIN                   -
      IF test EQ 1 THEN BEGIN                 -
        PROBLEM  >-------------------------
      ENDIF
    ENDIF
  ENDIF
ENDFOR
```

What I'd like to do:

If an error in the third IF statement occurs/ if the statement does not match the criteria, then the FOR-loop should skip the current file (f=n) and continue with the next file (f=n+1).

I saw that there is a function called CONTINUE but I couldn't make it work yet.

Does anyone have solution for this problem?

Thank you very much!

Greetings

Subject: Re: Skip file during loop based on subprocess
Posted by Helder Marchetto on Thu, 30 Jul 2015 13:14:22 GMT
View Forum Message <> Reply to Message

On Thursday, July 30, 2015 at 1:55:03 PM UTC+2, Kai Heckel wrote:

```
>  Hey there!
>
>  Here is what I have:
>
>  FOR f=0, N_ELEMENTS(foldername)-1 DO BEGIN<-----
>    IF test EQ 1 THEN BEGIN                     -
>      IF test EQ 1 THEN BEGIN                   -
>        IF test EQ 1 THEN BEGIN                 -
>          PROBLEM  >-------------------------
>        ENDIF
```

>   ENDIF
>   ENDIF
> ENDFOR
>
>
> What I'd like to do:
>
> If an error in the third IF statement occurs/ if the statement does not match the criteria, then the FOR-loop should skip the current file (f=n) and continue with the next file (f=n+1).
>
> I saw that there is a function called CONTINUE but I couldn't make it work yet.
>
> Does anyone have solution for this problem?
>
> Thank you very much!
>
> Greetings

Hi,
I'm not sure what "PROBLEM" means, therefore I've made two versions. The easiest one:

```
pro testEasyContinue
for i=0,10 do begin
   if i eq 5 then begin
      if i eq 5 then begin
         if i eq 5 then begin
            print, 'skip this one!'
            continue
         endif
      endif
   endif
   print, i
endfor
end
```

and the complicated one that catches an error:

```
pro testContinue
for i=0,10 do begin
   if i eq 5 then begin
      if i eq 5 then begin
         if i eq 5 then begin
            Catch, theError
            if theError ne 0 then begin
               print, 'skip this one!'
               continue
            endif
            a = b  ;generate error
      endif
    endif
  endif
  print, i
endfor
end
```

Both give the same result:

```
       0
       1
       2
       3
       4
skip this one!
       6
       7
       8
       9
      10
```

I hope it helps.

Cheers,
Helder