



Tech Info Library

Pascal: Dollar amount formatter (1 of 2)

Revised: 11/7/84
Security: Everyone

Pascal: Dollar amount formatter (1 of 2)

=====

The following Pascal function allows you to format printed values as dollar amounts. The function uses the APPLESTUFF unit. GET_DOLLARS takes three integer parameters and returns a real value for possible future calculations:

GET_DOLLARS (HPOS, VPOS, LIMIT)

The whole dollar amount is restricted to "LIMIT" number of characters (4 whole numbers). Three additional characters are allowed for the decimal and cents. Character space not used by LIMIT may be used by cents; however, all entries are rounded to the closest whole cent. The resulting number is printed, right justified, on the screen on top of the input characters.

```
writeln ('Enter value:');
Value:= GET_DOLLARS (20, 10, 4)
```

"Enter value:" is printed on the screen as normal, while the input cursor is positioned at screen location 20,10. Some possible input and corresponding outputs are:

Inputs	Outputs
9999	9999.00
9.99999	10.00
.999	1.00
.99	.99
1234.56	1234.56
123.456	123.46

```
function GET_DOLLARS (HPos, VPos, Limit: integer): real;
var RealStr, Space: string;
    Count      : integer;
    Value      : real;
```

```
procedure GET_REAL_STR (Limit: integer; var InStr: string);
var Entry      : string;
    Decimal, Back_Space, Return, ch: char;
    Count      : integer;
```

```

Real_Keys, GoodOnes      : set of char;
GotIt                    : boolean;

```

```

function KEY_IN: char;
  var ch: char;
  begin
    ch:= ' '; {Initialize variable}
    repeat until KEYPRESS;
    UNITREAD (2, ch, 1,, 12);
    KEY_IN:= ch
  end; {Key_In}

begin
  InStr:= ''; {Beginning default}
  Count:= 0; {Beginning default}
  Entry:= ' '; {Initialize as one character}
  Decimal:= '.';
  Back_Space:= chr(8);
  Return:= chr(13);
  Real_Keys:= [Decimal, '0'..'9']; {Valid characters}
                                     {in real numbers}
  GoodOnes:= Real_Keys + [Back_Space, Return]; {Valid}
                                               {inputs}
  GotIt:= false; {Beginning default};
  repeat
    repeat ch:= KEY_IN until (ch in GoodOnes);
    if (ch = Decimal) then begin
      GoodOnes:= GoodOnes - [Decimal]; {Can't use but one}
      Limit:= Limit + 3 {OK to get cents now}
    end;
    if (ch in Real_Keys) and (Count < Limit) then begin
      Entry [1]:= ch; {Convert char to string}
      InStr:= concat (InStr, Entry);
      Count:= Count + 1;
      write (ch)
    end;
    if (ch = Back_Space) and (Count > 0) then begin
      write (Back_Space, ' ', Back_Space);
      ch:= InStr [Count]; {See what last character is}
      if (ch = Decimal) then begin
        GoodOnes:= GoodOnes + [Decimal]; {Can use decimal}
                                               {again}
        Limit:= Limit - 3
      end;
      if (Count = 1) then InStr:= '';
      if (Count > 1) then delete (InStr, Count, 1);
      Count:= Count - 1
    end;
    if (ch = Return) then GotIt:= true
  until GotIt
end; {Get_Real_Str}

procedure DOLLAR_FORMAT (var Sample: string);

```

Apple Technical Communications

Tech Info Library Article Number:319