

Meter Estimated Next Due Date

Call this procedure and print the result using the following lines:

```
do METER_DATE($ReadingDate1, #MeterReading1, #AvgMeterUnit1, #MeterFrequency1,  
             #LastMeterReading1, $UseTargetDate, $LastStartDate, $LastCompDate, 'DD-MON-  
YYYY', $MeterESTNextDate)
```

```
Print $MeterESTNextDate (,457)
```

```
!*****
```

```
! METER - Estimated Next Due Date
```

```
!
```

```
! This procedure calculates the Meter Estimated Next Due Date.
```

```
!
```

```
!
```

```
! Arguments:
```

```
!
```

```
! $EQReadingDate - Equipment.ReadingDate , Last Equipment Meter Reading Date
```

```
! #EQMeterReading - Equipment.MeterReading, Last Equipment Meter Reading
```

```
! #EQAvgMeterUnit - Equipment.AvgMeterUnit, Average Units per day
```

```
! #PMMeterFrequency - PM.MeterFrequency, PM Meter Frequency
```

```
! #PMLastMeterReading - PM.LastMeterReading , Reading at Last Workorder
```

```
! $UseTargetStart - PM.UseTargetDate (Y/N)
```

```
! $TargetStartDate - PM.LastStartDate
```

```
! $CompletionDate - PM.LastCOMPDate
```

```
!
```

```
! $dateformat - the date format to be returned
```

```
!
```

```
! $EstNextDate is passed back to calling procedure
```

```
!
```

```
!*****
```

```
begin-procedure METER_DATE($EQReadingDate, #EQMeterReading, #EQAvgMeterUnit,  
#PMMeterFrequency,
```

```
    #PMLastMeterReading, $UseTargetStart, $xTargetStartDate,
```

```
    $xCompletionDate, $dateformat, :$ESTNextDate )
```

```
declare-variable
```

```
date $EQReadingDate
```

```
date $Targetstartdate
```

```
date $Completiondate
```

```
date $xESTNextDate
```

```
End-Declare
```

```
move " to $EstNextDate
```

```
!Check AVGMeterUnit > 0
```

```
If #EQAvgMeterUnit > 0
```

```
IF #PMLastMeterReading = #EQMeterReading
```

```
!Check if Meter Reading taken since last Workorder.
```

```
If isnull($EQReadingDate)=0
```

```

move $xEQReadingDate to $EQReadingDate
Move #PMMeterFrequency to #Adddays
Subtract #EQMeterReading from #Adddays
Add #PMLastMeterReading to #Adddays
Divide #EQAvgMeterUnit into #Adddays

Let $xEstNextDate = dateadd($EQReadingDate,'day', #Adddays )
Let $EstNextDate = Datetostr($xEstNextDate,$dateformat)

End-if
Else
!No Reading Taken Since Last Workorder.
If $UseTargetStart = 'Y'
If isnull($xTargetStartDate)=0
Move $xTargetStartDate to $TargetStartDate
Move #PMMeterFrequency to #Adddays
Divide #EQAvgMeterUnit into #Adddays
Let $xEstNextDate = dateadd($TargetStartDate,'day', #Adddays )
Let $EstNextDate = Datetostr($xEstNextDate,$dateformat)
End-IF
Else
If isnull($xCompletionDate)=0
Move $xCompletionDate to $CompletionDate
Move #PMMeterFrequency to #Adddays
Divide #EQAvgMeterUnit into #Adddays
Let $xEstNextDate = dateadd($CompletionDate,'day', #Adddays )
Let $EstNextDate = Datetostr($xEstNextDate,$dateformat)
End-IF
End-if

End-if

End-Procedure
|*****

```