Daily data are often generated by nondaily processes: for instance, financial markets are closed on weekends and holidays. Stata’s time-series date schemes ([U] 24.3 Time-series dates) allow for daily data, but gaps in time series may be problematic. A model that uses lags or differences will lose several observations every time a gap appears, discarding many of the original data points. Analysis of “business-daily” data often proceeds by assuming that Monday follows Friday, and so on. At the same time, we usually want data to be placed on Stata’s time-series calendar so that useful tools such as the `tsline` graph will work and label data points with readable dates; see [TS] `tsline`.

At a recent Stata Users Group presentation in Boston, David Drukker spoke to this point. His solution: generate two date variables, one containing the actual calendar dates, another numbering successive available observations consecutively. The former variable (`caldate`) is `tsset` (see [TS] `tsset`) when the calendar dates are to be used, whereas the latter (`seqdate`) is `tsset` when statistical analyses are to be performed.

We download daily data on the 3-month U.S. Treasury bill rate with Drukker’s `freduse` command (Drukker 2006) and retain the August 2005–present data for analysis. (We can also view the data graphically with `tsline`.)

```
. freduse DTB3
   (13764 observations read)
. rename daten caldate
. tsset caldate
   time variable:  caldate, 04jan1954 to 05oct2006, but with gaps
. keep if tin(1aug2005,)
   (13455 observations deleted)
. label var caldate date
```

These data do not contain observations for weekends and are missing for U.S. holidays. We may not want to drop the observations containing missing data, though, as we may have complete data for other variables: for instance, exchange rate data are available every day. If there were no missing data in our series—only missing observations—we could use Drukker’s suggestion and `generate seqdate = _n`. As we have observations for which `DTB3` is missing, we follow a more complex route:

```
. quietly generate byte notmiss = `DTB3' < .
. quietly generate seqdate = cond(notmiss, sum(notmiss),.)
. tsset seqdate
   time variable:  seqdate, 1 to 297
```
The variable `seqdate` is created as the sequential day number for every nonmissing day and is itself missing when `DTB3` is missing—allowing us to use this variable in `tsset` and then use time-series operators (see [U] 11.1 varlist) in `generate` or estimation commands such as `regress`. We may want to display the transformed data (or results from estimation, such as predicted values) on a time-series graph. We can just revert to the other `tsset`:

```
. quietly generate dDTB3 = D.DTB3
. quietly regress dDTB3 L(1/5).dDTB3
. predict double dDTB3hat, xb
(18 missing values generated)
. label var dDTB3 "Daily change in three-month Treasury rate"
. tsset caldate
    time variable: caldate, 01aug2005 to 05oct2006, but with gaps
. tsline dDTB3 dDTB3hat, yline(0) xlabel(, angle(45))
```

If we retain both the `caldate` and `seqdate` variables in our saved dataset, we will always be able to view these data either on a time-series calendar or as a sequential series. In my research, I need to know how many calendar days separate each observed point (1 for Thursday–Friday but 3 for Friday–Monday) and then sum `DTB3` by month, weighting each observation by the square root of the days of separation:

```
. tsset seqdate
    time variable: seqdate, 1 to 297
. quietly generate dcal = D.caldate if seqdate < .
. quietly generate month = mofd(caldate) if seqdate < .
. format %tm month
. sort month (seqdate)
. quietly by month: generate adjchange = sum(dDTB3/sqrt(dcal))
. quietly by month: generate sumchange = adjchange if _n==_N & month < .
```
. list month sumchange if sumchange < ., sep(0) noobs

<table>
<thead>
<tr>
<th>month</th>
<th>sumchange</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005m8</td>
<td>-.003812</td>
</tr>
<tr>
<td>2005m9</td>
<td>-.0810769</td>
</tr>
<tr>
<td>2005m10</td>
<td>.2424316</td>
</tr>
<tr>
<td>2005m11</td>
<td>-.063453</td>
</tr>
<tr>
<td>2005m12</td>
<td>.096188</td>
</tr>
<tr>
<td>2006m1</td>
<td>.2769615</td>
</tr>
<tr>
<td>2006m2</td>
<td>.099641</td>
</tr>
<tr>
<td>2006m3</td>
<td>.0142265</td>
</tr>
<tr>
<td>2006m4</td>
<td>.0938675</td>
</tr>
<tr>
<td>2006m5</td>
<td>.0350555</td>
</tr>
<tr>
<td>2006m6</td>
<td>.0327906</td>
</tr>
<tr>
<td>2006m7</td>
<td>.0304485</td>
</tr>
<tr>
<td>2006m8</td>
<td>-.083812</td>
</tr>
<tr>
<td>2006m9</td>
<td>-.123094</td>
</tr>
<tr>
<td>2006m10</td>
<td>.0442265</td>
</tr>
</tbody>
</table>

References