

はすべて離散データといえます。例えば2018年4月「1日」の日足の四本値データの次のデータは2018年4月「2日」の日足四本値データです。前のデータから次のデータまで1日分ジャンプしており、それらの間にデータは存在しません。同様に1分足であれば、10時「55分」の次のデータは10時「56分」です。両者の間にデータは存在せず1分ジャンプしています。さらに短いインターバルであるティックデータも同様です。このようにデータ間に距離があり離れているため「離散」データと呼ばれます。

詳しくは第3章で解説しますが、トレストとEasyLanguageは原則として足の単位で計算を行うように設計されています（68ページ参照）。足の終了が契機となって次の足の冒頭でプログラムを1回だけ走らせます。つまり、その原則から足の始値と終値の間は休んでいる設計になっていることを前提にプログラムを開発します。なお、足の間（イントラバー）でプログラムを稼働させる方法も学びます（212ページ参照）。

典型的な例としては、日足での自動売買プログラムを開発すると前場の始値でプログラムは稼働しますが、その日の終値ではプログラムは稼働しません。次にプログラムが稼働するのは翌営業日の前場の始値がつく時になります。損切りや利益確定など場中にプログラムを稼働させるには、このトレストの設計に配慮したプログラミングを行います。

チャートの足が持つデータ

チャートに表示される1本1本の足毎に以下の各データが格納され、データウィンドウでその格納されているデータを確認できます。これらはすべての足種で共通の項目です。例えば1分足であっても時間だけではなく日付データを持ちますし、月足であっても日付だけではなく時間のデータを持ちます。

足が持つデータ

日付 (date)、時間 (time)

四本値: 始値 (open)、高値 (high)、安値 (low)、終値 (close)

出来高 (volume 、 ticks、 upticks、 downticks)

今度はプリント文でチャートの足が持つデータを印刷ログに表示し、データウィンドウの値と比較してみましょう。インジケーター形式でファイルを用意し、エディタで以下のプログラムを書いて検証します。

volume,ticks, upticks,downticksの違い

```
print(date[0]:0:0,spaces(2),volume[0]:0:0,
spaces(2),ticks[0]:0:0,spaces(2),upticks[0]:0:0,
spaces(2),downticks[0]:0:0,spaces(2),(upticks[0] + downticks[0]):0:0);
```

印刷ログ

1180131	554453	13794625	554453	13240172	13794625
1180201	525827	13157959	525827	12632132	13157959
1180202	593576	14773675	593576	14180099	14773675