実行すると、売上履歴(uriage.csv)と顧客台帳(kokyaku_daicho.xlsx)のデータの先頭5行のデータが確認できます。

ここで、売上履歴のデータに注目すると、item_nameやitem_priceに欠損値や表記の整合性がない事に気が付くかと思います。

このようにデータ等で顕在する入力ミスや表記方法の違い等が混在し、不整合を起こしている状態を「**データの揺れ**」と言います。

表2-2：データの揺れの例

分類	例	説明
日付	2019-10-10 2019/10/10 10/10/2019 2019年10月10日	同じ日付でもフォーマットの違いで別の文字列データとなります。この辺りの揺れを自動で補正してくれる言語もありますが、混在している場合等は注意が必要です。
名前	佐々木太郎 佐々木 太郎 佐々木　太郎 佐々木多郎 佐々木太郎	人間にとっては見た目や意味に大差のない半角・全角スペースの有無ですが、システムにとっては別のデータとなってしまいます。 入力時の変換ミス等により、本来同じ人物でも別の名前になってしまいます。

上記のように、同じ日付、同じ人名でも、フォーマットや入力ミス等により、別のデータとなってしまうケースが、手作業で作成されたデータには必ず付いて回る問題です。

人間はデータの揺れを補完してデータを理解してしまいますが、システムはそうはいきません。

また、データの揺れを解消し、整合性を担保する事はデータ分析を行うのに基礎となるべき重要な点です。ここをあやふやにしたまま分析しても結果の信憑性や信頼性は担保できません。

それでは、このようなケースはどのようにデータの揺れを解消し、整合性を取っていけばよいのでしょうか。

整合性を整えるために、まずはデータのもつ属性や意味を理解します。

売上履歴においては、「purchase_date」「item_name」「item_price」「customer_name」が格納されている事が確認できます。

整合性を整えるためには、まずデータの揺れを把握する事から始めます。

ノック12：データの揺れを見てみよう

まずは、売上履歴のitem_nameを抽出して、データの揺れを確認してみましょう。

```
uriage_data["item_name"].head()
```

図2-3：データの揺れ(商品名)

ノック１２：データの揺れを見てみよう

```
In [3]: uriage_data["item_name"].head()
Out[3]: 0      商品A
        1    商 品 S
        2    商 品 a
        3      商品Z
        4      商品a
        Name: item_name, dtype: object
```

uriage_data["item_name"].head()で、売上履歴からitem_nameのみを抽出し先頭5行のデータを表示しています。

データを見てみると「商品A」「商品a」「商 品 a」と、スペースが含まれていたり、アルファベットが小文字になっていたりというデータが確認できます。

このままデータ分析を行ってしまうと、「商品A」「商品a」「商 品 a」はそれぞれ別の商品として集計されてしまい、本来一つの商品である「商品A」の正確な集計が得られません。