それでは顧客台帳の顧客名に対してスペースの除去を実施します。
行う処理は商品名の補正で用いた手法とほぼ同じです。

```
kokyaku_data["顧客名"] = kokyaku_data["顧客名"].str.replace("　", "")
kokyaku_data["顧客名"] = kokyaku_data["顧客名"].str.replace(" ", "")
kokyaku_data["顧客名"].head()
```

図2-16：補正後の顧客台帳の顧客名

```
In [17]: kokyaku_data["顧客名"] = kokyaku_data["顧客名"].str.replace("　", "")
         kokyaku_data["顧客名"] = kokyaku_data["顧客名"].str.replace(" ", "")
         kokyaku_data["顧客名"].head()

Out[17]: 0    須賀ひとみ
         1     岡田敏也
         2      芳賀希
         3      荻野愛
         4     栗田憲一
         Name: 顧客名, dtype: object
```

一度商品名の補正でも行っている.str.replace()による全角・半角スペースの除去を行い、結果を表示しています。
無事スペースが除去され、売上履歴と同じ体系にする事ができました。

今回はテストデータなので、非常にシンプルな揺れを題材としていますが、実際のデータの名前項目については、名前の誤変換などの複雑な揺れが存在する事が多々あります。
名前の誤変換などの場合、それが誤変換なのか別人なのかが判断できないため、単純にプログラムで補正する事ができません。現場の運用スタッフ等とヒアリングをし、地道に名寄せ作業を行う必要があります。また、同姓同名のデータが存在する場合は登録日や生年月日等、他の情報を用いて区別していく必要があります。

ノック17：日付の揺れを補正しよう

次に顧客台帳の登録日の揺れを補正していきましょう。
「**ノック11：データを読み込んでみよう**」で表示した**図2-2**を再確認してみましょう。

登録日を見ると「42782」のように日付ではない数字がいくつか見られます。
原因は取込元データ(Excel)にありますので、kokyaku_daicho.xlsxの該当部分を表示します。

図2-17：取込元のExcelデータ

	A	B	C	D	E
1	顧客名	かな	地域	メールアドレス	登録日
2	須賀ひとみ	すが ひとみ	H市	suga_hitomi@example.co	2018/01/04
3	岡田　敏也	おかだ としや	E市	okada_toshiya@example.c	2017年2月16日
4	芳賀 希	はが のぞみ	A市	haga_nozomi@example.co	2018/01/07
5	荻野 愛	おぎの あい	F市	ogino_ai@example.com	2017年5月17日
6	栗田 憲一	くりた けんいち	E市	kurita_kenichi@example.c	2018年1月27日
7	梅沢 麻緒	うめざわ まお	A市	umezawa_mao@example.	2017/06/20

右端の登録日に違う書式の日付が混在している事が確認できました。
Excelデータを取り扱う際、注意すべき点として、「書式が違うデータが混在する」事が挙げられます。
プログラム言語によっては、上記のような書式揺れを自動的に吸収してくれるものもありますが、今回のケースではyyyy年mm月dd日のデータを正しく日付として認識していないようです。

それでは、この日付を統一フォーマットに補正していきましょう。

```
flg_is_serial = kokyaku_data["登録日"].astype("str").str.isdigit()
flg_is_serial.sum()
```

図2-18：数値となってしまっている箇所の特定

```
In [18]: flg_is_serial = kokyaku_data["登録日"].astype("str").str.isdigit()
         flg_is_serial.sum()

Out[18]: 22
```

まずは、「42782」のように「数値」として取り込まれてしまっているデータを特定します。
1行目のflg_is_serial = kokyaku_data["登録日"].astype("str").str.isdigit()では、顧客台帳の登録日が数値かどうかを.str.isdigit()で判定してい