cleaning-student

April 15, 2020

0.1 Gather

```
In [248]: import pandas as pd
          import numpy as np

In [249]: patients = pd.read_csv('patients.csv')
          treatments = pd.read_csv('treatments.csv')
          adverse_reactions = pd.read_csv('adverse_reactions.csv')
```

0.2 Assess

```
In [250]: patients.head()
```

Out[250]:

	patient_id	assigned_sex	given_name	surname	address	city	state	zip_code	country	contact	birthdate	weight	height	bmi
0	1	female	Zoe	Wellish	576 Brown Bear Drive	Rancho California	California	92390.0	United States	951-719-9170ZoeWellish@superrito.com	7/10/1976	121.7	66	19.6
1	2	female	Pamela	Hill	2370 University Hill Road	Armstrong	Illinois	61812.0	United States	PamelaSHill@cuvox.de+1 (217) 569-3204	4/3/1967	118.8	66	19.2
2	3	male	Jae	Debord	1493 Poling Farm Road	York	Nebraska	68467.0	United States	402-363-6804JaeMDebord@gustr.com	2/19/1980	177.8	71	24.8
3	4	male	Liêm	Phan	2335 Webster Street	Woodbridge	NJ	7095.0	United States	PhanBaLiem@jourrapide.com+1 (732) 636-8246	7/26/1951	220.9	70	31.7
4	5	male	Tim	Neudorf	1428 Turkey Pen Lane	Dothan	AL	36303.0	United States	334-515-7487TimNeudorf@cuvox.de	2/18/1928	192.3	27	26.1

```
In [251]: treatments.head()
```

Out[251]:

	given_name	surname	auralin	novodra	hba1c_start	hba1c_end
0	veronika	jindrová	41u - 48u	-	7.63	7.20

```
1     elliot  richardson          -  40u - 45u        7.56       7.09
2   yukitaka    takenaka          -  39u - 36u        7.68       7.25
3       skye  gormanston  33u - 36u          -        7.97       7.62
4     alissa      montez          -  33u - 29u        7.78       7.46

   hba1c_change
0           NaN
1          0.97
2           NaN
3          0.35
4          0.32
```

```
In [252]: adverse_reactions.head()
```

Out[252]:

	given_name	surname	adverse_reaction
0	berta	napolitani	injection site discomfort
1	lena	baer	hypoglycemia
2	joseph	day	hypoglycemia
3	flavia	fiorentino	cough
4	manouck	wubbels	throat irritation

```
In [253]: patients.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503 entries, 0 to 502
Data columns (total 14 columns):
patient_id      503 non-null int64
assigned_sex    503 non-null object
given_name      503 non-null object
surname         503 non-null object
address         491 non-null object
city            491 non-null object
state           491 non-null object
zip_code        491 non-null float64
country         491 non-null object
contact         491 non-null object
birthdate       503 non-null object
weight          503 non-null float64
height          503 non-null int64
bmi             503 non-null float64
dtypes: float64(3), int64(2), object(9)
memory usage: 55.1+ KB
```

```
In [254]: treatments.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 280 entries, 0 to 279
Data columns (total 7 columns):
```

```
given_name      280 non-null object
surname         280 non-null object
auralin         280 non-null object
novodra         280 non-null object
hba1c_start     280 non-null float64
hba1c_end       280 non-null float64
hba1c_change    171 non-null float64
dtypes: float64(3), object(4)
memory usage: 15.4+ KB
```

```
In [255]: adverse_reactions.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34 entries, 0 to 33
Data columns (total 3 columns):
given_name          34 non-null object
surname             34 non-null object
adverse_reaction    34 non-null object
dtypes: object(3)
memory usage: 896.0+ bytes
```

```
In [256]: all_columns = pd.Series(list(patients) + list(treatments) + list(adverse_reactions))
          all_columns[all_columns.duplicated()]
```

```
Out[256]: 14    given_name
          15       surname
          21    given_name
          22       surname
          dtype: object
```

```
In [257]: list(patients)
```

```
Out[257]: ['patient_id',
           'assigned_sex',
           'given_name',
           'surname',
           'address',
           'city',
           'state',
           'zip_code',
           'country',
           'contact',
           'birthdate',
           'weight',
           'height',
           'bmi']
```

```
In [258]: patients[patients['address'].isnull()]
```

Out[258]:

	patient_id	assigned_sex	given_name	surname	address	city	state
209	210	female	Lalita	Eldarkhanov	NaN	NaN	NaN
219	220	male	M	Quynh	NaN	NaN	NaN
230	231	female	Elisabeth	Knudsen	NaN	NaN	NaN
234	235	female	Martina	Tománková	NaN	NaN	NaN
242	243	male	John	O'Brian	NaN	NaN	NaN
249	250	male	Benjamin	Mehler	NaN	NaN	NaN
257	258	male	Jin	Kung	NaN	NaN	NaN
264	265	female	Wafiyyah	Asfour	NaN	NaN	NaN
269	270	female	Flavia	Fiorentino	NaN	NaN	NaN
278	279	female	Generosa	Cabán	NaN	NaN	NaN
286	287	male	Lewis	Webb	NaN	NaN	NaN
296	297	female	Ch	Lâm	NaN	NaN	NaN

	zip_code	country	contact	birthdate	weight	height	bmi
209	NaN	NaN	NaN	8/14/1950	143.4	62	26.2
219	NaN	NaN	NaN	4/9/1978	237.8	69	35.1
230	NaN	NaN	NaN	9/23/1976	165.9	63	29.4
234	NaN	NaN	NaN	4/7/1936	199.5	65	33.2
242	NaN	NaN	NaN	2/25/1957	205.3	74	26.4
249	NaN	NaN	NaN	10/30/1951	146.5	69	21.6
257	NaN	NaN	NaN	5/17/1995	231.7	69	34.2
264	NaN	NaN	NaN	11/3/1989	158.6	63	28.1
269	NaN	NaN	NaN	10/9/1937	175.2	61	33.1
278	NaN	NaN	NaN	12/16/1962	124.3	69	18.4
286	NaN	NaN	NaN	4/1/1979	155.3	68	23.6
296	NaN	NaN	NaN	5/14/1990	181.1	63	32.1

In [259]:
```
patients.describe()
```

Out[259]:

	patient_id	zip_code	weight	height	bmi
count	503.000000	491.000000	503.000000	503.000000	503.000000
mean	252.000000	49084.118126	173.434990	66.634195	27.483897
std	145.347859	30265.807442	33.916741	4.411297	5.276438
min	1.000000	1002.000000	48.800000	27.000000	17.100000
25%	126.500000	21920.500000	149.300000	63.000000	23.300000
50%	252.000000	48057.000000	175.300000	67.000000	27.200000
75%	377.500000	75679.000000	199.500000	70.000000	31.750000
max	503.000000	99701.000000	255.900000	79.000000	37.700000

In [260]:
```
treatments.describe()
```

Out[260]:

	hba1c_start	hba1c_end	hba1c_change
count	280.000000	280.000000	171.000000
mean	7.985929	7.589286	0.546023
std	0.568638	0.569672	0.279555
min	7.500000	7.010000	0.200000
25%	7.660000	7.270000	0.340000
50%	7.800000	7.420000	0.380000

```
75%       7.970000    7.570000      0.920000
max       9.950000    9.580000      0.990000
```

In [261]:
```
patients.sample(5)
```

Out[261]:

	patient_id	assigned_sex	given_name	surname	address	city	state	zip_code	country	contact	birthdate	weight	height	bmi
144	145	male	Mile	Stani	4640 Windy Ridge Road	Fort Wayne	IN	46804.0	United States	260-591-5755MileStanic@dayrep.com	10/31/1961	244.9	71	34.2
34	35	female	Mariana	Souza	577 Chipmunk Lane	Orrington	ME	4474.0	United States	207-825-8634MarianaGomesSouza@superrito.com	3/6/1948	152.9	63	27.1
328	329	female	Anja	Hueber	3216 Lodgeville Road	Minneapolis	MN	55402.0	United States	AnjaHueber@teleworm.us+1 (612) 342-6065	4/16/1987	151.8	65	25.3
341	342	female	Fatimah	Khoury	883 Oakwood Circle	Fullerton	CA	93632.0	United States	FatimahAqilahKhoury@superrito.com1 949 290 0728	1/23/1950	120.3	67	18.8
352	353	male	Marek	Dvoák	633 Better Street	Savannah	GA	31401.0	United States	912-988-6655MarekDvorak@gustr.com	12/19/1966	227.7	67	35.7

In [262]:
```
patients.surname.value_counts()
```

Out[262]:
```
Doe            6
Taylor         3
Jakobsen       3
Liu            2
Lng            2
Berg           2
Batukayev      2
Lund           2
Johnson        2
Cabrera        2
Kowalczyk      2
Schiavone      2
Tucker         2
Cindri         2
```

```
Gersten        2
Kadyrov        2
Parker         2
Aranda         2
Correia        2
Lâm            2
Ogochukwu      2
Souza          2
Hueber         2
Dratchev       2
Bùi            2
Grímsdóttir    2
Woniak         2
T              2
Collins        2
Nilsen         2
               ..
Baer           1
Lindström      1
Wysocki        1
Resanovi       1
Hill           1
Hunter         1
Schmitt        1
Woodward       1
Chung          1
Selassie       1
Wellish        1
Lavrentyev     1
Czerwinska     1
Gyenes         1
Chidi          1
Vukeli         1
Luoma          1
Compagnon      1
Ferreira       1
Miller         1
Obinna         1
Glaser         1
Bogolyubova    1
MacDonald      1
Tsukada        1
Knutsen        1
Fournier       1
Piirainen      1
Gunnarsson     1
Scholz         1
Name: surname, Length: 466, dtype: int64
```

```
In [263]: patients.address.value_counts()
```

```
Out[263]: 123 Main Street               6
          2778 North Avenue             2
          648 Old Dear Lane             2
          2476 Fulton Street            2
          1027 Tenmile Road             1
          846 Copperhead Road           1
          3209 Crowfield Road           1
          3613 Lodgeville Road          1
          1904 Granville Lane           1
          2970 Forest Avenue            1
          3115 May Street               1
          945 Maple Avenue              1
          4458 Stark Hollow Road        1
          2146 Willow Greene Drive      1
          2831 Milford Street           1
          3977 Jail Drive               1
          3893 Eva Pearl Street         1
          3130 Jessie Street            1
          4277 Mutton Town Road         1
          3113 Timber Ridge Road        1
          260 Derek Drive               1
          4040 Linda Street             1
          4148 Callison Lane            1
          1346 Nicholas Street          1
          2127 Elk City Road            1
          3427 Gerald L. Bates Drive    1
          3411 Pyramid Valley Road      1
          631 Isaacs Creek Road         1
          323 Platinum Drive            1
          3251 Radio Park Drive         1
                                        ..
          1257 Elsie Drive              1
          2126 Pearl Street             1
          1072 Bird Spring Lane         1
          4689 Briarhill Lane           1
          479 Elmwood Avenue            1
          3390 Hidden Meadow Drive      1
          1956 Rosemont Avenue          1
          3303 Anmoore Road             1
          115 Frank Avenue              1
          1736 Parrill Court            1
          4707 Parkway Street           1
          2121 Liberty Avenue           1
          4111 Thunder Road             1
          475 Preston Street            1
          2775 Single Street            1
```

```
4220 Simpson Square           1
3781 Hamill Avenue            1
3434 Holt Street              1
2645 Moore Avenue             1
547 Weekley Street            1
707 Gateway Avenue            1
3094 Oral Lake Road           1
4704 Edsel Road               1
4243 Hidden Meadow Drive      1
1815 Garrett Street           1
2595 Feathers Hooves Drive    1
149 Marion Drive              1
3686 Meadowcrest Lane         1
1493 Randolph Street          1
1813 Lindale Avenue           1
Name: address, Length: 483, dtype: int64
```

In [264]:
```
patients[patients.address.duplicated()]
```

Out[264]:

	patient_id	assigned_sex	given_name	surname	address	city	state	zip_code	country
29	30	male	Jake	Jakobsen	648 Old Dear Lane	Port Jervis	New York	12771.0	United States
219	220	male	M	Quynh	NaN	NaN	NaN	NaN	NaN
229	230	male	John	Doe	123 Main Street	New York	NY	12345.0	United States
230	231	female	Elisabeth	Knudsen	NaN	NaN	NaN	NaN	NaN
234	235	female	Martina	Ománková	NaN	NaN	NaN	NaN	NaN
237	238	male	John	Doe	123 Main Street	New York	NY	12345.0	United States
242	243	male	John	O'Brian	NaN	NaN	NaN	NaN	NaN
244	245	male	John	Doe	123 Main Street				
249	250	male	Benjamin	Mehler	NaN				
251	252	male	John	Doe	123 Main Street				
257	258	male	Jin	Kung	NaN				
264	265	female	Wafiyyah	Asfour	NaN				
269	270	female	Flavia	Fiorentino	NaN				
277	278	male	John	Doe	123 Main Street				
278	279	female	Generosa	Cabán	NaN				
282	283	female	Sandy	Taylor	2476 Fulton Street				
286	287	male	Lewis	Webb	NaN				
296	297	female	Ch	Lâm	NaN				
502	503	male	Pat	Gersten	2778 North Avenue				

```
244     New York        NY   12345.0  United States
249          NaN       NaN       NaN            NaN
251     New York        NY   12345.0  United States
257          NaN       NaN       NaN            NaN
264          NaN       NaN       NaN            NaN
269          NaN       NaN       NaN            NaN
277     New York        NY   12345.0  United States
278          NaN       NaN       NaN            NaN
282     Rainelle        WV   25962.0  United States
286          NaN       NaN       NaN            NaN
296          NaN       NaN       NaN            NaN
502         Burr  Nebraska   68324.0  United States

                                        contact   birthdate  weight  height  \
29   JakobCJakobsen@einrot.com+1 (845) 858-7707    8/1/1985   155.8      67
219                                         NaN    4/9/1978   237.8      69
229                johndoe@email.com1234567890    1/1/1975   180.0      72
230                                         NaN   9/23/1976   165.9      63
234                                         NaN    4/7/1936   199.5      65
237                johndoe@email.com1234567890    1/1/1975   180.0      72
242                                         NaN   2/25/1957   205.3      74
244                johndoe@email.com1234567890    1/1/1975   180.0      72
249                                         NaN  10/30/1951   146.5      69
251                johndoe@email.com1234567890    1/1/1975   180.0      72
257                                         NaN   5/17/1995   231.7      69
264                                         NaN   11/3/1989   158.6      63
269                                         NaN   10/9/1937   175.2      61
277                johndoe@email.com1234567890    1/1/1975   180.0      72
278                                         NaN  12/16/1962   124.3      69
282       304-438-2648SandraCTaylor@dayrep.com  10/23/1960   206.1      64
286                                         NaN    4/1/1979   155.3      68
296                                         NaN   5/14/1990   181.1      63
502       PatrickGersten@rhyta.com402-848-4923    5/3/1954   138.2      71

      bmi
29   24.4
219  35.1
229  24.4
230  29.4
234  33.2
237  24.4
242  26.4
244  24.4
249  21.6
251  24.4
257  34.2
264  28.1
269  33.1
```

```
277  24.4
278  18.4
282  35.4
286  23.6
296  32.1
502  19.3
```

In [265]:
```
patients.weight.sort_values()
```

Out[265]:
```
210     48.8
459    102.1
335    102.7
74     103.2
317    106.0
171    106.5
51     107.1
270    108.1
198    108.5
48     109.1
478    109.6
141    110.2
38     111.8
438    112.0
14     112.0
235    112.2
307    112.4
191    112.6
408    113.1
49     113.3
326    114.0
338    114.1
253    117.0
321    118.4
168    118.8
1      118.8
350    119.0
207    119.2
265    120.0
341    120.3
       ...
332    224.0
252    224.2
12     224.2
222    224.8
166    225.3
111    225.9
101    226.2
150    226.6
```

```
352    227.7
428    227.7
88     227.7
13     228.4
339    229.0
182    230.3
121    230.8
257    231.7
395    231.9
246    232.1
219    237.8
11     238.7
50     238.9
441    239.1
499    239.6
439    242.0
487    242.4
144    244.9
61     244.9
283    245.5
118    254.5
485    255.9
Name: weight, Length: 503, dtype: float64
```

```
In [266]: weight_lbs = patients[patients.surname == 'Zaitseva'].weight * 2.20462
          height_in = patients[patients.surname == 'Zaitseva'].height
          bmi_check = 703 * weight_lbs / (height_in * height_in)
          bmi_check
```

```
Out[266]: 210    19.055827
          dtype: float64
```

```
In [267]: patients[patients.surname == 'Zaitseva'].bmi
```

```
Out[267]: 210    19.1
          Name: bmi, dtype: float64
```

```
In [268]: sum(treatments.auralin.isnull())
```

```
Out[268]: 0
```

```
In [269]: sum(treatments.novodra.isnull())
```

```
Out[269]: 0
```

Quality

`patients` **table**

- Zip code is a float not a string
- Zip code has four digits sometimes
- Tim Neudorf height is 27 in instead of 72 in
- Full state names sometimes, abbreviations other times
- Dsvid Gustafsson
- Missing demographic information (address - contact columns) ***(can't clean)***
- Erroneous datatypes (assigned sex, state, zip_code, and birthdate columns)
- Multiple phone number formats
- Default John Doe data
- Multiple records for Jakobsen, Gersten, Taylor
- kgs instead of lbs for Zaitseva weight

`treatments` **table**

- Missing HbA1c changes
- The letter 'u' in starting and ending doses for Auralin and Novodra
- Lowercase given names and surnames
- Missing records (280 instead of 350)
- Erroneous datatypes (auralin and novodra columns)
- Inaccurate HbA1c changes (leading 4s mistaken as 9s)
- Nulls represented as dashes (-) in auralin and novodra columns

`adverse_reactions` **table**

- Lowercase given names and surnames

Tidiness

- Contact column in `patients` table should be split into phone number and email
- Three variables in two columns in `treatments` table (treatment, start dose and end dose)
- Adverse reaction should be part of the `treatments` table
- Given name and surname columns in `patients` table duplicated in `treatments` and `adverse_reactions` tables

0.3 Clean

```
In [270]: patients_clean = patients.copy()
          treatments_clean = treatments.copy()
          adverse_reactions_clean = adverse_reactions.copy()
```

0.3.1 Missing Data

Complete the following two "Missing Data" **Define, Code, and Test** sequences after watching the *"Address Missing Data First"* video.

`treatments`**: Missing records (280 instead of 350)**

Define *Your definition here. Note: the missing* `treatments` *records are stored in a file named* `treatments_cut.csv`*, which you can see in this Jupyter Notebook's dashboard (click the* ***jupyter*** *logo in the top lefthand corner of this Notebook). Hint: documentation page for the function used in the solution.*

Code

```
In [271]: treatments_clean = pd.concat([treatments_clean,  pd.read_csv('treatments_cut.csv')], i

In [272]: treatments_clean.head()
```

Out[272]:

	given_name	surname	auralin	novodra	hba1c_start	hba1c_end	hba1c_change
0	veronika	jindrová	41u - 48u	-	7.63	7.20	NaN
1	elliot	richardson	-	40u - 45u	7.56	7.09	0.97
2	yukitaka	takenaka	-	39u - 36u	7.68	7.25	NaN
3	skye	gormanston	33u - 36u	-	7.97	7.62	0.35
4	alissa	montez	-	33u - 29u	7.78	7.46	0.32

Test

```
In [273]: # Your testing code here
          # Append 70 rows on to original 280 rows
          assert treatments_clean.shape == (350, 7)
          # Bug fix: unique indexes
          assert treatments_clean.index.duplicated().any() == False
```

`treatments`**: Missing HbA1c changes and Inaccurate HbA1c changes (leading 4s mistaken as 9s)** *Note: the "Inaccurate HbA1c changes (leading 4s mistaken as 9s)" observation, which is an accuracy issue and not a completeness issue, is included in this header because it is also fixed by the cleaning operation that fixes the missing "Missing HbA1c changes" observation. Multiple observations in one* ***Define, Code, and Test*** *header occurs multiple times in this notebook.*

Define

- Turn `hba1c_change` column into a derived column `|hba1c_start - hba1c_end|`

Code

```
In [274]: # Your cleaning code here
          treatments_clean['hba1c_change'] = (treatments_clean['hba1c_start'] - treatments_clean
```

Test

```
In [275]: # Your testing code here
          assert (treatments_clean['hba1c_change'] == (treatments_clean['hba1c_start'] - treatme
```

0.3.2 Tidiness

Complete the following four "Tidiness" **Define, Code, and Test** sequences after watching the *"Cleaning for Tidiness"* video.

Contact column in `patients` table contains two variables: phone number and email

Define *Your definition here. Hint 1: use regular expressions with pandas' `str.extract` method. Here is an amazing regex tutorial. Hint 2: various phone number regex patterns. Hint 3: email address regex pattern, which you might need to modify to distinguish the email from the phone number.*

Code

```
In [276]: patients_clean.loc[215]
```

```
Out[276]: patient_id                              216
          assigned_sex                           male
          given_name                             John
          surname                                 Doe
          address                     123 Main Street
          city                               New York
          state                                    NY
          zip_code                              12345
          country                       United States
          contact         johndoe@email.com1234567890
          birthdate                          1/1/1975
          weight                                  180
          height                                   72
          bmi                                    24.4
          Name: 215, dtype: object
```

```
In [277]: # Your cleaning code here
          # patients.contact.str.extract()
          # import re
          phone_reg = r'(\(?\d{3}.*?\d{3}.*?\d{4})'
          patients_clean['phone_number'] = patients_clean.contact.str.extract(phone_reg)
          patients_clean['phone_number'] = patients_clean.phone_number.str.strip().replace(regex
```

```
In [278]: email_reg = r'([A-Za-z].*\.[A-Za-z]*)'
          # email_reg = r'([A-Za-z].*)'
          patients_clean['email'] = patients_clean.contact.str.extract(email_reg)
```

```
In [279]: patients_clean = patients_clean.drop('contact', axis=1)
```

```
In [280]: patients_clean.head()
```

Out[280]:

	patient_id	assigned_sex	given_name	surname	address	city	state	zip_code	country	birthdate	weight	height	bmi	phone_number	email
0	1	female	Zoe	Wellish	576 Brown Bear Drive	Rancho California	California	92390.0	United States	7/10/1976	121.7	66	19.6	951-719-9170	ZoeWellish@superrito.com
1	2	female	Pamela	Hill	2370 University Hill Road	Armstrong	Illinois	61812.0	United States	4/3/1967	118.8	66	19.2	217-569-3204	PamelaSHill@cuvox.de
2	3	male	Jae	Debord	1493 Poling Farm Road	York	Nebraska	68467.0	United States	2/19/1980	177.8	71	24.8	402-363-6804	JaeMDebord@gustr.com
3	4	male	Liêm	Phan	2335 Webster Street	Woodbridge	NJ	7095.0	United States	7/26/1951	220.9	70	31.7	732-636-8246	PhanBaLiem@jourrapide.com
4	5	male	Tim	Neudorf	1428 Turkey Pen Lane	Dothan	AL	36303.0	United States	2/18/1928	192.3	27	26.1	334-515-7487	TimNeudorf@cuvox.de

Test

```
In [281]: # Your testing code here
          one, _, three = list(patients_clean["phone_number"].str.len().unique())
          assert one == 12
          assert three == 10

In [282]: # Test for email
          assert patients_clean.email.shape[0] == patients_clean.phone_number.shape[0]

In [283]: assert patients_clean.shape[1] == (patients.shape[1] + 1)
```

Three variables in two columns in `treatments` table (treatment, start dose and end dose)

Define *Your definition here. Hint: use pandas' melt function and `str.split()` method. Here is an excellent `melt` tutorial.*

Code

```
In [284]: constant_cols = [col for col in treatments_clean.columns if col not in ('auralin', 'no

In [285]: # Your cleaning code here
          melt_prac = pd.melt(treatments_clean,id_vars= constant_cols, var_name="treatment", val
          melt_prac = melt_prac.query("dose != '-'")
          melt_prac['dose'] = melt_prac.dose.str.split('-').apply(lambda x: (x[0][:-2], x[1][:-1
```

```
melt_prac['start'] = melt_prac.dose.apply(lambda x: x[0]).astype(int)
melt_prac['end'] = melt_prac.dose.apply(lambda x: x[1]).astype(int)
melt_prac = melt_prac.drop('dose', axis=1)
melt_prac = melt_prac.reset_index(drop=True)
treatments_clean = melt_prac
treatments_clean.head()
```

Out[285]:

	given_name	surname	hba1c_start	hba1c_end	hba1c_change	treatment	\
0	veronika	jindrová	7.63	7.20	0.43	auralin	
1	skye	gormanston	7.97	7.62	0.35	auralin	
2	sophia	haugen	7.65	7.27	0.38	auralin	
3	eddie	archer	7.89	7.55	0.34	auralin	
4	asia	woniak	7.76	7.37	0.39	auralin	

	start	end
0	41	48
1	33	36
2	37	42
3	31	38
4	30	36

Test

```
In [286]: # Your testing code here
          assert (treatments_clean.start.dtype ==  treatments_clean.end.dtype == np.int64)
```

Adverse reaction should be part of the `treatments` table

Define *Your definition here. Hint: tutorial for the function used in the solution.*

- merge adverse_reactions with treatments using `merge`

Code

```
In [288]: # Your cleaning code here
          treatments_clean = treatments_clean.merge(adverse_reactions, on=['given_name', 'surnam
```

Test

```
In [291]: # Your testing code here
          treatments_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 350 entries, 0 to 349
Data columns (total 9 columns):
given_name          350 non-null object
surname             350 non-null object
hba1c_start         350 non-null float64
```

```
hba1c_end           350 non-null float64
hba1c_change        350 non-null float64
treatment           350 non-null object
start               350 non-null int64
end                 350 non-null int64
adverse_reaction    35 non-null object
dtypes: float64(3), int64(2), object(4)
memory usage: 27.3+ KB
```

Given name and surname columns in `patients` table duplicated in `treatments` and `adverse_reactions` tables and Lowercase given names and surnames

Define *Your definition here. Hint: tutorial for one function used in the solution and tutorial for another function used in the solution.*

Code

```
In [73]: # Your cleaning code here
```

Test

```
In [74]: # Your testing code here
```

0.3.3 Quality

Complete the remaining "Quality" **Define, Code, and Test** sequences after watching the *"Cleaning for Quality"* video.

Zip code is a float not a string and Zip code has four digits sometimes

Define *Your definition here. Hint: see the "Data Cleaning Process" page.*

Code

```
In [75]: # Your cleaning code here
```

Test

```
In [76]: # Your testing code here
```

Tim Neudorf height is 27 in instead of 72 in

Define *Your definition here.*

Code

```
In [77]: # Your cleaning code here
```

Test

```
In [78]: # Your testing code here
```

Full state names sometimes, abbreviations other times

Define *Your definition here. Hint: tutorial for method used in solution.*

Code

```
In [79]: # Your cleaning code here
```

Test

```
In [80]: # Your testing code here
```

Dsvid Gustafsson

Define *Your definition here.*

Code

```
In [81]: # Your cleaning code here
```

Test

```
In [82]: # Your testing code here
```

Erroneous datatypes (assigned sex, state, zip_code, and birthdate columns) and Erroneous datatypes (auralin and novodra columns) and The letter 'u' in starting and ending doses for Auralin and Novodra

Define *Your definition here. Hint: documentation page for one method used in solution, documentation page for one function used in the solution, and documentation page for another method used in the solution.*

Code

```
In [83]: # Your cleaning code here
```

Test

```
In [84]: # Your testing code here
```

Multiple phone number formats

Define *Your definition here. Hint: helpful Stack Overflow answer.*

Code

```
In [85]: # Your cleaning code here
```

Test

```
In [86]: # Your testing code here
```

Default John Doe data

Define *Your definition here. Recall that it is assumed that the data that this John Doe data displaced is not recoverable.*

Code

```
In [87]: # Your cleaning code here
```

Test

```
In [88]: # Your testing code here
```

Multiple records for Jakobsen, Gersten, Taylor

Define *Your definition here.*

Code

```
In [89]: # Your cleaning code here
```

Test

```
In [90]: # Your testing code here
```

kgs instead of lbs for Zaitseva weight

Define *Your definition here.*

Code

```
In [91]: # Your cleaning code here
```

Test

```
In [92]: # Your testing code here
```