quality-solution

April 15, 2020

0.1 Gather

```
In [1]: import pandas as pd
        import numpy as np
```

```
In [2]: patients = pd.read_csv('patients.csv')
        treatments = pd.read_csv('treatments.csv')
        adverse_reactions = pd.read_csv('adverse_reactions.csv')
```

0.2 Assess

```
In [3]: patients
```

Out[3]:

	patient_id	assigned_sex	given_name	surname
0	1	female	Zoe	Wellish
1	2	female	Pamela	Hill
2	3	male	Jae	Debord
3	4	male	Liêm	Phan
4	5	male	Tim	Neudorf
5	6	male	Rafael	Costa
6	7	female	Mary	Adams
7	8	female	Xiuxiu	Chang
8	9	male	Dsvid	Gustafsson
9	10	female	Sophie	Cabrera
10	11	female	Sandy	Gunnarsson
11	12	male	Abdul-Nur	Isa
12	13	male	Omeokachie	Ibeamaka
13	14	female	Anenechi	Chidi
14	15	female	Asia	Woniak
15	16	male	Søren	Lund
16	17	female	Tám	Liu
17	18	female	Roxanne	Andreyeva
18	19	male	William	Oates
19	20	male	Zak	Kelly
20	21	female	Sofia	Karlsen
21	22	male	Samúel	Guðbrandsson
22	23	male	Manchu	Su
23	24	male	Lovre	Gali

24	25	male	Jakob	Jakobsen
25	26	male	Gregor	Bole
26	27	female	Ella	Lund
27	28	male	Joseph	Tucker
28	29	male	Robert	Wolf
29	30	male	Jake	Jakobsen
..	...	...	...	...
473	474	female	Kate	Wilkinson
474	475	female	Esperanza	Labrosse
475	476	male	Malik	Vaneker
476	477	female	Berta	Napolitani
477	478	male	Juliusz	Majewski
478	479	female	Edelma	Villalpando
479	480	male	Tapa	Arsanukayev
480	481	male	Nasser	Mansour
481	482	male	Michael	Kristensen
482	483	male	Diogo	Souza
483	484	female	Angel	Grant
484	485	male	Placido	Udinesi
485	486	male	Trifon	Izmailov
486	487	male	Samuel	Blix
487	488	male	Ivar	Löfgren
488	489	male	Mika	Martinsson
489	490	female	Jasmine	Sykes
490	491	male	Jackson	Addison
491	492	female	Vanessa	Ferguson
492	493	male	Poldi	Tar
493	494	female	Fen	Chin
494	495	female	Sirkka	Piirainen
495	496	male	Hajime	Tsukada
496	497	male	Alexander	Hueber
497	498	male	Masataka	Murakami
498	499	male	Mustafa	Lindström
499	500	male	Ruman	Bisliev
500	501	female	Jinke	de Keizer
501	502	female	Chidalu	Onyekaozulu
502	503	male	Pat	Gersten

	address	city	state	zip_code	\
0	576 Brown Bear Drive	Rancho California	California	92390.0	
1	2370 University Hill Road	Armstrong	Illinois	61812.0	
2	1493 Poling Farm Road	York	Nebraska	68467.0	
3	2335 Webster Street	Woodbridge	NJ	7095.0	
4	1428 Turkey Pen Lane	Dothan	AL	36303.0	
5	1140 Willis Avenue	Daytona Beach	Florida	32114.0	
6	3145 Sheila Lane	Burbank	NV	84728.0	
7	2687 Black Oak Hollow Road	Morgan Hill	CA	95037.0	
8	1790 Nutter Street	Kansas City	MO	64105.0	

9	3303 Anmoore Road	New York	New York	10011.0
10	87 Wood Duck Drive	Rudyard	MI	49780.0
11	1092 Farm Meadow Drive	Brentwood	TN	37027.0
12	2544 Worley Avenue	Lynchburg	VA	24504.0
13	826 Broad Street	Birmingham	AL	35203.0
14	4970 Heather Sees Way	Tulsa	OK	74105.0
15	2438 Shady Pines Drive	Kingsport	VA	37660.0
16	2152 Heritage Road	Fresno	California	93706.0
17	2103 Edington Drive	Smyrna	GA	30082.0
18	441 Tibbs Avenue	Ekalaka	MT	59324.0
19	994 Hill Croft Farm Road	Oroville	California	95966.0
20	2931 Romano Street	Whitman	MA	2382.0
21	1904 Granville Lane	Elmsford	NJ	10523.0
22	1092 Deans Lane	Pleasantville	NY	10570.0
23	4941 Marion Drive	Winter Haven	Florida	33830.0
24	648 Old Dear Lane	Port Jervis	New York	12771.0
25	922 Chapmans Lane	Albuquerque	NM	87109.0
26	1207 Garfield Road	Peoria	IL	61602.0
27	4982 Wood Street	Venice	LA	70091.0
28	2386 Linda Street	Fort Washington	PA	19034.0
29	648 Old Dear Lane	Port Jervis	New York	12771.0
..	...	...	...	...
473	664 Lyon Avenue	South Boston	MA	2127.0
474	1370 Flint Street	Atlanta	GA	30303.0
475	1270 Haul Road	Mountain View	California	94041.0
476	1815 Garrett Street	Philadelphia	PA	19108.0
477	4435 Poe Road	Florence	SC	29501.0
478	312 Jim Rosa Lane	San Jose	CA	95134.0
479	4720 Gordon Street	Ontario	California	91762.0
480	547 Weekley Street	San Antonio	TX	78212.0
481	1614 Heather Sees Way	Tulsa	OK	74116.0
482	4033 White Avenue	Corpus Christi	TX	78401.0
483	990 Melville Street	Memphis	TN	38118.0
484	1094 Jones Avenue	Greensboro	NC	28716.0
485	3697 Drainer Avenue	Fort Walton Beach	FL	32548.0
486	3488 Clair Street	Waco	TX	76706.0
487	1346 Nicholas Street	Ottawa	KS	66067.0
488	962 George Street	Ocala	Florida	34471.0
489	2607 Water Street	Lafayette	California	94549.0
490	1160 Taylor Street	New Rochelle	New York	10801.0
491	241 Freshour Circle	San Antonio	TX	78205.0
492	3958 Liberty Avenue	Burbank	California	91505.0
493	1826 Poplar Chase Lane	Boise	ID	83702.0
494	4102 Ritter Avenue	Roseville	MI	48066.0
495	4111 Thunder Road	San Mateo	CA	94403.0
496	3868 Freed Drive	Stockton	California	95204.0
497	1179 Patton Lane	Tulsa	OK	74116.0
498	2530 Victoria Court	Milton Mills	ME	3852.0

```
499    494 Clarksburg Park Road             Sedona          AZ   86341.0
500           649 Nutter Street      Overland Park          MO   64110.0
501    3652 Boone Crockett Lane            Seattle          WA   98109.0
502           2778 North Avenue               Burr    Nebraska   68324.0

          country                                            contact  \
0    United States              951-719-9170ZoeWellish@superrito.com
1    United States             PamelaSHill@cuvox.de+1 (217) 569-3204
2    United States                  402-363-6804JaeMDebord@gustr.com
3    United States        PhanBaLiem@jourrapide.com+1 (732) 636-8246
4    United States                   334-515-7487TimNeudorf@cuvox.de
5    United States          386-334-5237RafaelCardosoCosta@gustr.com
6    United States                775-533-5933MaryBAdams@einrot.com
7    United States             XiuxiuChang@einrot.com1 408 778 3236
8    United States           816-265-9578DavidGustafsson@armyspy.com
9    United States      SophieCabreraIbarra@teleworm.us1 718 795 9124
10   United States            906-478-8949SandyGunnarsson@dayrep.com
11   United States         Abdul-NurMummarIsa@rhyta.com1 931 207 0839
12   United States          OmeokachieIbeamaka@einrot.com434-509-2614
13   United States         AnenechiChidi@armyspy.com+1 (205) 417-8095
14   United States                  AsiaWozniak@rhyta.com918-712-3469
15   United States                    276-225-1955SrenFLund@gustr.com
16   United States             LieuThiThuTam@dayrep.com1 559 765 7836
17   United States           RoxanneAndreyeva@armyspy.com678-829-8578
18   United States              406-775-2696WilliamVOates@armyspy.com
19   United States                   ZakKelly@rhyta.com1 530 532 8397
20   United States            SofiaTKarlsen@teleworm.us1 781 447 1763
21   United States          973-445-5341SamuelGubrandsson@teleworm.us
22   United States                    914-745-6108ManchuSu@einrot.com
23   United States                 LovreGalic@gustr.com1 813 355 9476
24   United States         JakobCJakobsen@einrot.com+1 (845) 858-7707
25   United States                  GregorBole@gustr.com505-828-4955
26   United States                  309-671-8852EllaLund@armyspy.com
27   United States               985-814-7603JosephNTucker@rhyta.com
28   United States               RobertWolf@fleckens.hu1 267 895 7462
29   United States         JakobCJakobsen@einrot.com+1 (845) 858-7707
..             ...                                                ...
473  United States            KateWilkinson@armyspy.com1 508 905 2371
474  United States          EsperanzaLabrosse@armyspy.com678-263-3564
475  United States             MalikVaneker@superrito.com650-962-7179
476  United States              267-972-3749BertaNapolitani@rhyta.com
477  United States     JuliuszMajewski@superrito.com+1 (843) 212-6421
478  United States  EdelmaVillalpandoSantillan@teleworm.us+1 (415)...
479  United States           TapaArsanukayev@dayrep.com1 909 458 2515
480  United States       NasserMazinMansour@fleckens.hu1 210 326 5509
481  United States          MichaelKristensen@gustr.com1 918 706 2776
482  United States        361-693-4960DiogoBarrosSouza@jourrapide.com
483  United States                731-577-0292AngelGrant@fleckens.hu
```

484	United States	336-697-2005PlacidoUdinesi@dayrep.com
485	United States	TrifonIzmailov@fleckens.hu1 850 659 0417
486	United States	254-681-4504SamuelBlix@dayrep.com
487	United States	IvarLofgren@armyspy.com1 785 229 1188
488	United States	352-453-4601MikaMartinsson@armyspy.com
489	United States	JasmineSykes@jourrapide.com925-283-5425
490	United States	914-636-9304JacksonAddison@armyspy.com
491	United States	210-222-8684VanessaFerguson@jourrapide.com
492	United States	714-496-2264TarPoldi@superrito.com
493	United States	FenChin@gustr.com+1 (208) 388-1065
494	United States	SirkkaPiirainen@teleworm.us+1 (586) 790-0975
495	United States	650-570-4896HajimeTsukada@dayrep.com
496	United States	AlexanderHueber@jourrapide.com1 209 762 2320
497	United States	MasatakaMurakami@einrot.com+1 (918) 984-9171
498	United States	207-477-0579MustafaLindstrom@jourrapide.com
499	United States	928-284-4492RumanBisliev@gustr.com
500	United States	816-223-6007JinkedeKeizer@teleworm.us
501	United States	ChidaluOnyekaozulu@jourrapide.com1 360 443 2060
502	United States	PatrickGersten@rhyta.com402-848-4923

	birthdate	weight	height	bmi
0	7/10/1976	121.7	66	19.6
1	4/3/1967	118.8	66	19.2
2	2/19/1980	177.8	71	24.8
3	7/26/1951	220.9	70	31.7
4	2/18/1928	192.3	27	26.1
5	8/31/1931	183.9	70	26.4
6	11/19/1969	146.3	65	24.3
7	8/13/1958	158.0	60	30.9
8	3/6/1937	163.9	66	26.5
9	12/3/1930	194.7	64	33.4
10	7/16/1974	199.3	62	36.4
11	2/3/1954	238.7	73	31.5
12	8/5/1957	224.2	69	33.1
13	3/7/1961	228.4	67	35.8
14	8/15/1997	112.0	65	18.6
15	8/23/1922	201.5	64	34.6
16	11/14/1952	183.9	61	34.7
17	7/24/1922	129.1	60	25.2
18	9/4/1949	202.2	64	34.7
19	12/13/1988	208.8	70	30.0
20	9/24/1934	153.1	66	24.7
21	4/12/1983	223.7	69	33.0
22	1/19/1936	130.7	65	21.7
23	5/26/1960	222.9	66	36.0
24	8/1/1985	155.8	67	24.4
25	6/19/1922	180.8	67	28.3
26	12/19/1933	144.8	61	27.4

```
27    4/10/1959   175.8      72  23.8
28    6/26/1937   206.6      70  29.6
29     8/1/1985   155.8      67  24.4
..          ...     ...     ...   ...
473   7/18/1998   175.3      65  29.2
474   10/7/1961   181.5      63  32.1
475   9/25/1953   214.4      67  33.6
476   12/2/1958   153.3      63  27.2
477   9/29/1966   212.1      69  31.3
478   6/24/1977   109.6      63  19.4
479   9/15/1955   220.0      65  36.6
480   3/25/1938   183.5      66  29.6
481   8/10/1930   154.7      65  25.7
482    3/3/1945   220.0      65  36.6
483   8/14/1987   123.9      61  23.4
484   5/31/1934   175.8      65  29.3
485   2/15/1973   255.9      74  32.9
486    7/6/1983   211.4      74  27.1
487   11/7/1962   242.4      77  28.7
488   1/27/1970   165.0      67  25.8
489   12/1/1988   187.2      63  33.2
490   5/29/1953   192.7      69  28.5
491   9/21/1950   149.8      67  23.5
492   5/23/1970   184.6      70  26.5
493   3/18/1997   195.1      68  29.7
494   1/16/1942   126.3      67  19.8
495    9/5/1972   168.1      66  27.1
496   9/12/1942   194.0      72  26.3
497   8/19/1937   155.1      72  21.0
498   4/10/1959   181.1      72  24.6
499   3/26/1948   239.6      70  34.4
500   1/13/1971   171.2      67  26.8
501   2/13/1952   176.9      67  27.7
502    5/3/1954   138.2      71  19.3

[503 rows x 14 columns]
```

In [4]:
```
treatments
```

Out[4]:

	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	
5	jasmine	sykes	-	42u - 44u	7.56	7.18	
6	sophia	haugen	37u - 42u	-	7.65	7.27	
7	eddie	archer	31u - 38u	-	7.89	7.55	

8	saber	ménard	-	54u - 54u	8.08	7.70
9	asia	woniak	30u - 36u	-	7.76	7.37
10	joseph	day	29u - 36u	-	7.70	7.19
11	kristiina	hyypiä	-	36u - 38u	7.87	7.49
12	roxanne	andreyeva	29u - 38u	-	9.54	9.14
13	gregor	bole	-	47u - 45u	7.61	7.16
14	simone	baumgaertner	27u - 37u	-	7.74	7.30
15	enco	ibrik	55u - 68u	-	7.78	7.34
16	camilla	zaitseva	28u - 37u	-	7.53	7.13
17	gina	cain	-	36u - 36u	7.88	7.40
18	addolorata	lombardi	-	49u - 46u	7.75	7.33
19	khalid	johnsrud	-	54u - 54u	8.35	7.94
20	mile	stani	-	47u - 48u	7.66	7.24
21	tekla	walczak	29u - 39u	-	7.61	7.29
22	brancaleone	russo	53u - 60u	-	8.61	8.18
23	chiemela	tobeolisa	-	43u - 47u	7.59	7.17
24	isac	berg	31u - 41u	-	9.68	9.29
25	benoît	bonami	-	44u - 43u	9.82	9.40
26	suhaim	rahal	-	49u - 47u	7.94	7.50
27	mizuki	iwata	-	45u - 46u	7.70	7.23
28	clinton	miller	42u - 51u	-	7.79	7.40
29	eugene	mironov	42u - 49u	-	7.81	7.48
..	...	...	...	...	...	...
250	chen	yao	-	56u - 57u	7.90	7.51
251	aksel	vestergaard	-	42u - 38u	9.62	9.29
252	ellen	luman	-	40u - 39u	9.27	8.77
253	albino	schiavone	35u - 43u	-	7.56	7.15
254	jose	combs	-	39u - 36u	7.89	7.42
255	jia li	teng	48u - 54u	-	7.66	7.32
256	ilija	horvat	42u - 50u	-	7.77	7.38
257	mathilde	nørgaard	-	27u - 28u	8.50	8.10
258	csilla	herczegh	-	43u - 46u	7.71	7.27
259	aaliyah	rice	-	31u - 31u	7.64	7.33
260	david	beauvais	-	26u - 23u	7.87	7.47
261	caroline	shuler	-	50u - 54u	7.63	7.27
262	alex	crawford	51u - 62u	-	7.69	7.30
263	rebecca	jephcott	53u - 63u	-	7.96	7.57
264	chukwumoge	ogochukwu	-	41u - 39u	7.95	7.56
265	fearne	mcgregor	-	27u - 29u	7.83	7.48
266	ursula	freud	42u - 54u	-	7.75	7.46
267	leon	scholz	-	38u - 32u	7.72	7.29
268	yasmin	araujo	-	51u - 54u	7.82	7.36
269	hiromu	horikawa	-	47u - 46u	7.77	7.28
270	mika	martinsson	34u - 43u	-	7.50	7.17
271	leo	vieira	-	30u - 33u	7.74	7.36
272	steven	roy	-	41u - 43u	7.87	7.43
273	kate	wilkinson	36u - 39u	-	7.72	7.20
274	naja	enoksen	43u - 50u	-	7.98	7.59

275	albina	zetticci	45u - 51u	-	7.93	7.73
276	john	teichelmann	-	49u - 49u	7.90	7.58
277	mathea	lillebø	23u - 36u	-	9.04	8.67
278	vallie	prince	31u - 38u	-	7.64	7.28
279	samúel	guðbrandsson	53u - 56u	-	8.00	7.64

	hba1c_change
0	NaN
1	0.97
2	NaN
3	0.35
4	0.32
5	0.38
6	0.38
7	0.34
8	NaN
9	NaN
10	NaN
11	0.38
12	NaN
13	0.95
14	NaN
15	NaN
16	NaN
17	0.98
18	NaN
19	NaN
20	0.92
21	0.32
22	NaN
23	NaN
24	0.39
25	0.92
26	0.94
27	0.97
28	0.39
29	0.33
..	...
250	0.39
251	NaN
252	0.50
253	NaN
254	NaN
255	0.34
256	0.39
257	0.90
258	NaN
259	0.31

```
260           NaN
261           NaN
262          0.39
263          0.39
264          0.39
265          0.35
266          0.29
267          0.93
268          0.96
269           NaN
270          0.33
271           NaN
272          0.94
273           NaN
274           NaN
275          0.20
276           NaN
277          0.37
278          0.36
279          0.36

[280 rows x 7 columns]
```

In [5]:
```
adverse_reactions
```

Out[5]:

	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
5	jasmine	sykes	hypoglycemia
6	louise	johnson	hypoglycemia
7	albinca	komavec	hypoglycemia
8	noe	aranda	hypoglycemia
9	sofia	hermansen	injection site discomfort
10	tegan	johnson	headache
11	abel	yonatan	cough
12	abdul-nur	isa	hypoglycemia
13	leon	scholz	injection site discomfort
14	gabriele	saenger	hypoglycemia
15	jia li	teng	nausea
16	jakob	jakobsen	hypoglycemia
17	christopher	woodward	nausea
18	ole	petersen	hypoglycemia
19	finley	chandler	headache
20	anenechi	chidi	hypoglycemia
21	miosaw	winiewski	injection site discomfort

```
22        lixue       hsueh  injection site discomfort
23        merci      leroux               hypoglycemia
24         kang         mai  injection site discomfort
25       elliot  richardson               hypoglycemia
26      clinton      miller          throat irritation
27       idalia       moore               hypoglycemia
28       xiuxiu       chang               hypoglycemia
29         alex    crawford               hypoglycemia
30       monika      lonar               hypoglycemia
31       steven         roy                    headache
32      cecilie      nilsen               hypoglycemia
33    krisztina      magyar               hypoglycemia
```

In [6]: `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 [7]: `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 [8]: 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 [9]: all_columns = pd.Series(list(patients) + list(treatments) + list(adverse_reactions))
        all_columns[all_columns.duplicated()]
```

```
Out[9]: 14    given_name
        15       surname
        21    given_name
        22       surname
        dtype: object
```

```
In [10]: list(patients)
```

```
Out[10]: ['patient_id',
          'assigned_sex',
          'given_name',
          'surname',
          'address',
          'city',
          'state',
          'zip_code',
          'country',
          'contact',
          'birthdate',
          'weight',
          'height',
          'bmi']
```

```
In [11]: patients[patients['address'].isnull()]
```

```
Out[11]:      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 [12]:
```
patients.describe()
```

Out[12]:

	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 [13]:
```
treatments.describe()
```

Out[13]:

	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 [14]:
```
patients.sample(5)
```

Out[14]:
```
     patient_id assigned_sex given_name surname                  address  \
23           24         male      Lovre   Gali         4941 Marion Drive
```

```
483         484       female      Angel   Grant       990 Melville Street
316         317       female       Chân     Bùi          115 Frank Avenue
391         392         male      Daimy   Tromp  522 Lamberts Branch Road
128         129         male   Muhammad  Hughes      1965 Crestview Manor

             city    state  zip_code        country  \
23   Winter Haven  Florida   33830.0  United States
483       Memphis       TN   38118.0  United States
316       Amherst       MA    1002.0  United States
391       Sunrise       FL   33323.0  United States
128  Indianapolis       IN   46214.0  United States

                                    contact   birthdate  weight  height   bmi
23       LovreGalic@gustr.com1 813 355 9476   5/26/1960   222.9      66  36.0
483      731-577-0292AngelGrant@fleckens.hu   8/14/1987   123.9      61  23.4
316    413-259-3637BuiQuynhChan@fleckens.hu  10/11/1932   162.1      61  30.6
391    786-970-4206DaimyTromp@superrito.com   5/23/1990   198.0      75  24.7
128  MuhammadHughes@rhyta.com1 317 292 2394   9/21/1938   202.2      72  27.4
```

In [15]:
```
patients.surname.value_counts()
```

Out[15]:
```
Doe             6
Taylor          3
Jakobsen        3
Ogochukwu       2
Johnson         2
Silva           2
Kadyrov         2
Parker          2
Kowalczyk       2
Batukayev       2
Lâm             2
Nilsen          2
Berg            2
Lund            2
Gersten         2
Liu             2
Hueber          2
Cabrera         2
Lng             2
Dratchev        2
Tucker          2
Correia         2
T               2
Woniak          2
Schiavone       2
Aranda          2
Bùi             2
```

```
Collins          2
Souza            2
Cindri           2
                ..
Adamski          1
Chinedum         1
Musliyevich      1
Bowden           1
Murakami         1
Miller           1
Wellish          1
Haraguchi        1
van der Lubbe    1
Combs            1
Gustafsson       1
Bonami           1
Borgen           1
McGregor         1
Sung             1
Kalb             1
Petersen         1
Bois             1
Fodor            1
Ferrari          1
Johnsrud         1
Mattila          1
Salib            1
Ferguson         1
Mathiesen        1
Werner           1
Lange            1
Fomin            1
Ruais            1
Sakai            1
Name: surname, Length: 466, dtype: int64
```

```
In [16]: patients.address.value_counts()
```

```
Out[16]: 123 Main Street             6
         648 Old Dear Lane           2
         2476 Fulton Street          2
         2778 North Avenue           2
         1962 Cabell Avenue          1
         2549 Pearlman Avenue        1
         2333 Hidden Pond Road       1
         3688 Adonais Way            1
         2235 Catherine Drive        1
         633 Better Street           1
```

```
4792 Maud Street            1
465 Southern Street         1
962 George Street           1
353 Whaley Lane             1
4519 Sussex Court           1
3094 Oral Lake Road         1
2816 Ashford Drive          1
3391 Marcus Street          1
2356 Myra Street            1
1525 Crestview Terrace      1
4104 Kennedy Court          1
1846 Joseph Street          1
321 Briercliff Road         1
57 Norman Street            1
2866 Myra Street            1
2945 Ferguson Street        1
3165 Upton Avenue           1
312 Jim Rosa Lane           1
4015 Juniper Drive          1
1510 Allison Avenue         1
                           ..
2970 Forest Avenue          1
1495 Post Farm Road         1
4476 Center Street          1
1343 Clair Street           1
2168 Butternut Lane         1
2055 Emeral Dreams Drive    1
1731 Chandler Drive         1
108 Griffin Street          1
2152 Heritage Road          1
602 Tator Patch Road        1
1368 Yorkshire Circle       1
4839 North Avenue           1
1463 Beechwood Avenue       1
3072 Braxton Street         1
1012 Lords Way              1
2531 Cantebury Drive        1
4213 Isaacs Creek Road      1
4237 Hamilton Drive         1
2127 Elk City Road          1
2121 Liberty Avenue         1
163 Hide A Way Road         1
3683 Gorby Lane             1
441 Tibbs Avenue            1
3595 Stuart Street          1
550 Cliffside Drive         1
1251 Clarence Court         1
4643 Reeves Street          1
```

```
4168 Coventry Court          1
4649 Joanne Lane             1
36 Heather Sees Way          1
Name: address, Length: 483, dtype: int64
```

```
In [17]: patients[patients.address.duplicated()]
```

Out[17]:

	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	Tománková	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	New York	NY	12345.0	United States
249	250	male	Benjamin	Mehler	NaN	NaN	NaN	NaN	NaN
251	252	male	John	Doe	123 Main Street	New York	NY	12345.0	United States
257	258	male	Jin	Kung	NaN	NaN	NaN	NaN	NaN
264	265	female	Wafiyyah	Asfour	NaN	NaN	NaN	NaN	NaN
269	270	female	Flavia	Fiorentino	NaN	NaN	NaN	NaN	NaN
277	278	male	John	Doe	123 Main Street	New York	NY	12345.0	United States
278	279	female	Generosa	Cabán	NaN	NaN	NaN	NaN	NaN
282	283	female	Sandy	Taylor	2476 Fulton Street	Rainelle	WV	25962.0	United States
286	287	male	Lewis	Webb	NaN	NaN	NaN	NaN	NaN
296	297	female	Ch	Lâm	NaN	NaN	NaN	NaN	NaN
502	503	male	Pat	Gersten	2778 North Avenue	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 [18]:
```
patients.weight.sort_values()
```

Out[18]:
```
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 [19]: 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[19]: 210    19.055827
         dtype: float64

In [20]: patients[patients.surname == 'Zaitseva'].bmi

Out[20]: 210    19.1
         Name: bmi, dtype: float64

In [21]: sum(treatments.auralin.isnull())

Out[21]: 0

In [22]: sum(treatments.novodra.isnull())

Out[22]: 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 yet)***
- 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 [23]: patients_clean = patients.copy()
         treatments_clean = treatments.copy()
         adverse_reactions_clean = adverse_reactions.copy()
```

0.3.1 Missing Data

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

Define Import the cut treatments into a DataFrame and concatenate it with the original treatments DataFrame.

Code

```
In [24]: treatments_cut = pd.read_csv('treatments_cut.csv')
         treatments_clean = pd.concat([treatments_clean, treatments_cut],
                                      ignore_index=True)
```

Test

```
In [25]: treatments_clean.head()
```

```
Out[25]:   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 [26]: treatments_clean.tail()
```

```
Out[26]:     given_name     surname    auralin    novodra  hba1c_start  hba1c_end  \
345     rovzan     kishiev  32u - 37u          -         7.75       7.41
346      jakob    jakobsen          -  28u - 26u         7.96       7.51
347      bernd   schneider  48u - 56u          -         7.74       7.44
348      berta  napolitani          -  42u - 44u         7.68       7.21
349     armina       sauvé  36u - 46u          -         7.86       7.40

     hba1c_change
345          0.34
346          0.95
347          0.30
348           NaN
349           NaN
```

`treatments`: **Missing HbA1c changes and Inaccurate HbA1c changes (leading 4s mistaken as 9s)**

Define Recalculate the `hba1c_change` column: `hba1c_start` minus `hba1c_end`.

Code

```
In [27]: treatments_clean.hba1c_change = (treatments_clean.hba1c_start -
                                          treatments_clean.hba1c_end)
```

Test

```
In [28]: treatments_clean.hba1c_change.head()
```

```
Out[28]: 0    0.43
         1    0.47
         2    0.43
         3    0.35
         4    0.32
         Name: hba1c_change, dtype: float64
```

0.3.2 Tidiness

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

Define Extract the *phone number* and *email* variables from the *contact* column using regular expressions and pandas' `str.extract` method. Drop the *contact* column when done.

Code

```
In [29]: patients_clean['phone_number'] = patients_clean.contact.str.extract('((?:\+\d{1,2}\s)?\
         patients_clean['email'] = patients_clean.contact.str.extract('([a-zA-Z][a-zA-Z0-9_.+-]+
         # Note: axis=1 denotes that we are referring to a column, not a row
         patients_clean = patients_clean.drop('contact', axis=1)
```

Test

```
In [30]: # Confirm contact column is gone
         list(patients_clean)
```

```
Out[30]: ['patient_id',
          'assigned_sex',
          'given_name',
          'surname',
          'address',
          'city',
          'state',
          'zip_code',
          'country',
          'birthdate',
          'weight',
          'height',
          'bmi',
          'phone_number',
          'email']
```

```
In [31]: patients_clean.phone_number.sample(25)
```

```
Out[31]: 162         916-555-7247
         157         920-849-0384
         395         336-677-8769
         261         617-297-0387
         388         318-621-7385
         12          434-509-2614
         244           1234567890
         219                  NaN
         51          309 912 9553
         48          312-719-7238
         300         706-616-0152
         394         903-939-1025
```

```
52          515-727-9324
404         409-944-6516
435    +1 (757) 736-7026
208    +1 (817) 909-5667
116         606-417-4332
63          619-299-1495
368         617 830 7216
30          303-910-2058
117    +1 (504) 528-8317
115         707 262 6503
224         802-614-0812
410         405 449 7960
453         281-556-4376
Name: phone_number, dtype: object
```

```
In [32]: # Confirm that no emails start with an integer (regex didn't match for this)
         patients_clean.email.sort_values().head()
```

```
Out[32]: 404               AaliyahRice@dayrep.com
         11          Abdul-NurMummarIsa@rhyta.com
         332               AbelEfrem@fleckens.hu
         258             AbelYonatan@teleworm.us
         305    AddoloratаLombardi@jourrapide.com
         Name: email, dtype: object
```

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

Define Melt the *auralin* and *novodra* columns to a *treatment* and a *dose* column (dose will still contain both start and end dose at this point). Then split the dose column on `'-'` to obtain *start_dose* and *end_dose* columns. Drop the intermediate *dose* column.

Code

```
In [33]: treatments_clean = pd.melt(treatments_clean, id_vars=['given_name', 'surname', 'hba1c_s
                                    var_name='treatment', value_name='dose')
         treatments_clean = treatments_clean[treatments_clean.dose != "-"]
         treatments_clean['dose_start'], treatments_clean['dose_end'] = treatments_clean['dose']
         treatments_clean = treatments_clean.drop('dose', axis=1)
```

Test

```
In [34]: treatments_clean.head()
```

```
Out[34]:   given_name     surname  hba1c_start  hba1c_end  hba1c_change treatment  \
         0   veronika    jindrová         7.63       7.20          0.43   auralin
         3       skye  gormanston         7.97       7.62          0.35   auralin
         6     sophia      haugen         7.65       7.27          0.38   auralin
         7      eddie      archer         7.89       7.55          0.34   auralin
```

```
9       asia     woniak         7.76       7.37          0.39    auralin

  dose_start dose_end
0        41u      48u
3        33u      36u
6        37u      42u
7        31u      38u
9        30u      36u
```

Adverse reaction should be part of the `treatments` table

Define Merge the *adverse_reaction* column to the `treatments` table, joining on *given name* and *surname*.

Code

```
In [35]: treatments_clean = pd.merge(treatments_clean, adverse_reactions_clean,
                                     on=['given_name', 'surname'], how='left')
```

Test

```
In [36]: treatments_clean
```

```
Out[36]:        given_name        surname  hba1c_start  hba1c_end  hba1c_change  \
         0        veronika       jindrová         7.63       7.20          0.43
         1            skye     gormanston         7.97       7.62          0.35
         2          sophia         haugen         7.65       7.27          0.38
         3           eddie         archer         7.89       7.55          0.34
         4            asia        woniak         7.76       7.37          0.39
         5          joseph            day         7.70       7.19          0.51
         6         roxanne      andreyeva         9.54       9.14          0.40
         7          simone   baumgaertner         7.74       7.30          0.44
         8            enco         ibrik         7.78       7.34          0.44
         9         camilla       zaitseva         7.53       7.13          0.40
         10          tekla        walczak         7.61       7.29          0.32
         11    brancaleone          russo         8.61       8.18          0.43
         12           isac           berg         9.68       9.29          0.39
         13        clinton         miller         7.79       7.40          0.39
         14         eugene        mironov         7.81       7.48          0.33
         15    szilveszter          totth         7.70       7.38          0.32
         16      alexander      mathiesen         7.96       7.55          0.41
         17             ch            lâm         7.68       7.24          0.44
         18       wadysaw      wieczorek         7.92       7.47          0.45
         19       kristján        ingason         7.92       7.57          0.35
         20          marija       grubii         7.53       7.15          0.38
         21           sauli     koivuniemi         7.67       7.37          0.30
         22         mariana          souza         7.86       7.51          0.35
         23     kristoffer      martinsen         9.18       8.64          0.54
```

24	m	quynh	7.61	7.16	0.45
25	oles	zhdanov	7.52	7.11	0.41
26	triana.	terrazas	7.71	7.34	0.37
27	gabry	tomaszewski	7.87	7.47	0.40
28	leixandre	alanis	7.74	7.32	0.42
29	onyekachukwu	obinna	7.58	7.12	0.46
..	...	...	...	...	...
320	jane	citizen	7.98	7.60	0.38
321	angela	lavrentyev	7.61	7.14	0.47
322	edelma	villalpando	7.99	7.56	0.43
323	annika	vaara	7.73	7.34	0.39
324	chiho	higa	7.71	7.30	0.41
325	beatrycze	woniak	7.54	7.17	0.37
326	miosaw	winiewski	7.51	7.08	0.43
327	firenze	fodor	7.89	7.55	0.34
328	zoe	wellish	7.71	7.30	0.41
329	una	traustadóttir	8.00	7.50	0.50
330	lubo	pecha	7.79	7.45	0.34
331	meaza	brhane	7.70	7.36	0.34
332	adlan	shishani	7.84	7.37	0.47
333	sofia	hermansen	8.90	8.57	0.33
334	guðni	heimisson	7.64	7.24	0.40
335	eufemio	rosario	7.54	7.26	0.28
336	dalmacia	madrid	7.67	7.21	0.46
337	daimy	tromp	9.41	8.94	0.47
338	jeremy	montagu	7.68	7.36	0.32
339	nebechi	ekechukwu	7.78	7.39	0.39
340	satsita	batukayev	7.63	7.25	0.38
341	timothy	cotton	7.92	7.52	0.40
342	bjørnar	nilsen	7.99	7.70	0.29
343	borna	lezinger	7.55	7.18	0.37
344	mary	adams	7.65	7.26	0.39
345	christopher	woodward	7.51	7.06	0.45
346	maret	sultygov	7.67	7.30	0.37
347	lixue	hsueh	9.21	8.80	0.41
348	jakob	jakobsen	7.96	7.51	0.45
349	berta	napolitani	7.68	7.21	0.47

	treatment	dose_start	dose_end	adverse_reaction
0	auralin	41u	48u	NaN
1	auralin	33u	36u	NaN
2	auralin	37u	42u	NaN
3	auralin	31u	38u	NaN
4	auralin	30u	36u	NaN
5	auralin	29u	36u	hypoglycemia
6	auralin	29u	38u	NaN
7	auralin	27u	37u	NaN
8	auralin	55u	68u	NaN

9	auralin	28u	37u	NaN
10	auralin	29u	39u	NaN
11	auralin	53u	60u	NaN
12	auralin	31u	41u	NaN
13	auralin	42u	51u	throat irritation
14	auralin	42u	49u	NaN
15	auralin	35u	39u	NaN
16	auralin	47u	58u	NaN
17	auralin	45u	48u	NaN
18	auralin	24u	37u	NaN
19	auralin	44u	55u	NaN
20	auralin	37u	43u	NaN
21	auralin	43u	47u	NaN
22	auralin	36u	42u	NaN
23	auralin	29u	37u	NaN
24	auralin	57u	64u	NaN
25	auralin	54u	67u	NaN
26	auralin	34u	42u	NaN
27	auralin	29u	37u	NaN
28	auralin	61u	67u	NaN
29	auralin	37u	46u	NaN
..	...	...	...	...
320	novodra	37u	38u	NaN
321	novodra	28u	24u	NaN
322	novodra	24u	26u	NaN
323	novodra	20u	21u	NaN
324	novodra	46u	46u	NaN
325	novodra	26u	27u	NaN
326	novodra	34u	33u	injection site discomfort
327	novodra	30u	35u	NaN
328	novodra	33u	33u	NaN
329	novodra	35u	34u	NaN
330	novodra	30u	27u	NaN
331	novodra	37u	41u	NaN
332	novodra	43u	40u	NaN
333	novodra	34u	34u	injection site discomfort
334	novodra	40u	36u	NaN
335	novodra	37u	40u	NaN
336	novodra	26u	23u	NaN
337	novodra	40u	45u	NaN
338	novodra	52u	52u	NaN
339	novodra	37u	39u	NaN
340	novodra	42u	42u	NaN
341	novodra	26u	25u	NaN
342	novodra	36u	33u	NaN
343	novodra	42u	41u	NaN
344	novodra	32u	33u	NaN
345	novodra	55u	51u	nausea

```
346   novodra        26u      23u                         NaN
347   novodra        22u      23u  injection site discomfort
348   novodra        28u      26u               hypoglycemia
349   novodra        42u      44u  injection site discomfort

[350 rows x 9 columns]
```

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

Define Adverse reactions table is no longer needed so ignore that part. Isolate the patient ID and names in the `patients` table, then convert these names to lower case to join with `treatments`. Then drop the given name and surname columns in the treatments table (so these being lowercase isn't an issue anymore).

Code

```
In [37]: id_names = patients_clean[['patient_id', 'given_name', 'surname']]
         id_names.given_name = id_names.given_name.str.lower()
         id_names.surname = id_names.surname.str.lower()
         treatments_clean = pd.merge(treatments_clean, id_names, on=['given_name', 'surname'])
         treatments_clean = treatments_clean.drop(['given_name', 'surname'], axis=1)
```

```
/opt/conda/lib/python3.6/site-packages/pandas/core/generic.py:3110: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#
  self[name] = value
```

Test

```
In [38]: # Confirm the merge was executed correctly
         treatments_clean
```

```
Out[38]:      hba1c_start  hba1c_end  hba1c_change treatment dose_start dose_end  \
         0           7.63       7.20          0.43   auralin        41u      48u
         1           7.97       7.62          0.35   auralin        33u      36u
         2           7.65       7.27          0.38   auralin        37u      42u
         3           7.89       7.55          0.34   auralin        31u      38u
         4           7.76       7.37          0.39   auralin        30u      36u
         5           7.70       7.19          0.51   auralin        29u      36u
         6           7.70       7.19          0.51   auralin        29u      36u
         7           9.54       9.14          0.40   auralin        29u      38u
         8           7.74       7.30          0.44   auralin        27u      37u
         9           7.78       7.34          0.44   auralin        55u      68u
         10          7.53       7.13          0.40   auralin        28u      37u
```

11	7.61	7.29	0.32	auralin	29u	39u
12	8.61	8.18	0.43	auralin	53u	60u
13	9.68	9.29	0.39	auralin	31u	41u
14	7.79	7.40	0.39	auralin	42u	51u
15	7.81	7.48	0.33	auralin	42u	49u
16	7.70	7.38	0.32	auralin	35u	39u
17	7.96	7.55	0.41	auralin	47u	58u
18	7.68	7.24	0.44	auralin	45u	48u
19	7.92	7.47	0.45	auralin	24u	37u
20	7.92	7.57	0.35	auralin	44u	55u
21	7.53	7.15	0.38	auralin	37u	43u
22	7.67	7.37	0.30	auralin	43u	47u
23	7.86	7.51	0.35	auralin	36u	42u
24	9.18	8.64	0.54	auralin	29u	37u
25	7.61	7.16	0.45	auralin	57u	64u
26	7.52	7.11	0.41	auralin	54u	67u
27	7.71	7.34	0.37	auralin	34u	42u
28	7.87	7.47	0.40	auralin	29u	37u
29	7.74	7.32	0.42	auralin	61u	67u
..	...	...	...	...	...	...
319	7.98	7.60	0.38	novodra	37u	38u
320	7.61	7.14	0.47	novodra	28u	24u
321	7.99	7.56	0.43	novodra	24u	26u
322	7.73	7.34	0.39	novodra	20u	21u
323	7.71	7.30	0.41	novodra	46u	46u
324	7.54	7.17	0.37	novodra	26u	27u
325	7.51	7.08	0.43	novodra	34u	33u
326	7.89	7.55	0.34	novodra	30u	35u
327	7.71	7.30	0.41	novodra	33u	33u
328	8.00	7.50	0.50	novodra	35u	34u
329	7.79	7.45	0.34	novodra	30u	27u
330	7.70	7.36	0.34	novodra	37u	41u
331	7.84	7.37	0.47	novodra	43u	40u
332	8.90	8.57	0.33	novodra	34u	34u
333	7.64	7.24	0.40	novodra	40u	36u
334	7.54	7.26	0.28	novodra	37u	40u
335	7.67	7.21	0.46	novodra	26u	23u
336	9.41	8.94	0.47	novodra	40u	45u
337	7.68	7.36	0.32	novodra	52u	52u
338	7.78	7.39	0.39	novodra	37u	39u
339	7.63	7.25	0.38	novodra	42u	42u
340	7.92	7.52	0.40	novodra	26u	25u
341	7.99	7.70	0.29	novodra	36u	33u
342	7.55	7.18	0.37	novodra	42u	41u
343	7.65	7.26	0.39	novodra	32u	33u
344	7.51	7.06	0.45	novodra	55u	51u
345	7.67	7.30	0.37	novodra	26u	23u
346	9.21	8.80	0.41	novodra	22u	23u

```
347         7.96       7.51          0.45   novodra        28u      26u
348         7.68       7.21          0.47   novodra        42u      44u

              adverse_reaction  patient_id
0                          NaN         225
1                          NaN         242
2                          NaN         345
3                          NaN         276
4                          NaN          15
5                  hypoglycemia          70
6                  hypoglycemia          70
7                          NaN          18
8                          NaN         424
9                          NaN         292
10                         NaN         211
11                         NaN         133
12                         NaN         316
13                         NaN         101
14           throat irritation         451
15                         NaN         335
16                         NaN         389
17                         NaN          71
18                         NaN         297
19                         NaN         188
20                         NaN         282
21                         NaN         174
22                         NaN         146
23                         NaN          35
24                         NaN         350
25                         NaN         220
26                         NaN         102
27                         NaN         181
28                         NaN         466
29                         NaN         205
..                         ...         ...
319                        NaN         187
320                        NaN         234
321                        NaN         479
322                        NaN          49
323                        NaN         356
324                        NaN         208
325  injection site discomfort         373
326                        NaN          63
327                        NaN           1
328                        NaN         291
329                        NaN         363
330                        NaN         465
331                        NaN         421
```

```
332  injection site discomfort         376
333                         NaN         463
334                         NaN          81
335                         NaN         322
336                         NaN         392
337                         NaN         262
338                         NaN          68
339                         NaN         152
340                         NaN         431
341                         NaN         450
342                         NaN         194
343                         NaN           7
344                      nausea         153
345                         NaN         420
346  injection site discomfort         336
347               hypoglycemia          25
348  injection site discomfort         477

[349 rows x 8 columns]
```

```
In [39]: # Patient ID should be the only duplicate column
         all_columns = pd.Series(list(patients_clean) + list(treatments_clean))
         all_columns[all_columns.duplicated()]
```

```
Out[39]: 22    patient_id
         dtype: object
```

0.3.3 Quality

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

Define Convert the zip code column's data type from a float to a string using `astype`, remove the '.0' using string slicing, and pad four digit zip codes with a leading 0.

Code

```
In [40]: patients_clean.zip_code = patients_clean.zip_code.astype(str).str[:-2].str.pad(5, fillc
         # Reconvert NaNs entries that were converted to '0000n' by code above
         patients_clean.zip_code = patients_clean.zip_code.replace('0000n', np.nan)
```

Test

```
In [41]: patients_clean.zip_code.head()
```

```
Out[41]: 0    92390
         1    61812
         2    68467
         3    07095
         4    36303
         Name: zip_code, dtype: object
```

Tim Neudorf height is 27 in instead of 72 in

Define Replace height for rows in the `patients` table that have a height of 27 in (there is only one) with 72 in.

Code

```
In [42]: patients_clean.height = patients_clean.height.replace(27, 72)
```

Test

```
In [43]: # Should be empty
         patients_clean[patients_clean.height == 27]
```

```
Out[43]: Empty DataFrame
         Columns: [patient_id, assigned_sex, given_name, surname, address, city, state, zip_code
         Index: []
```

```
In [44]: # Confirm the replacement worked
         patients_clean[patients_clean.surname == 'Neudorf']
```

```
Out[44]:    patient_id assigned_sex given_name  surname               address    city  \
         4           5         male        Tim  Neudorf  1428 Turkey Pen Lane  Dothan

           state zip_code        country  birthdate  weight  height   bmi  \
         4    AL    36303  United States  2/18/1928   192.3      72  26.1

            phone_number                email
         4  334-515-7487  TimNeudorf@cuvox.de
```

Full state names sometimes, abbreviations other times

Define Apply a function that converts full state name to state abbreviation for California, New York, Illinois, Florida, and Nebraska.

Code

```
In [45]: # Mapping from full state name to abbreviation
         state_abbrev = {'California': 'CA',
                         'New York': 'NY',
                         'Illinois': 'IL',
                         'Florida': 'FL',
                         'Nebraska': 'NE'}

         # Function to apply
         def abbreviate_state(patient):
             if patient['state'] in state_abbrev.keys():
                 abbrev = state_abbrev[patient['state']]
```

```
            return abbrev
        else:
            return patient['state']

    patients_clean['state'] = patients_clean.apply(abbreviate_state, axis=1)
```

Test

In [46]:
```
patients_clean.state.value_counts()
```

Out[46]:
```
CA    60
NY    47
TX    32
IL    24
MA    22
FL    22
PA    18
GA    15
OH    14
LA    13
OK    13
MI    13
NJ    12
VA    11
MS    10
WI    10
MN     9
AL     9
IN     9
TN     9
KY     8
WA     8
NC     8
MO     7
NE     6
ID     6
NV     6
KS     6
CT     5
IA     5
SC     5
ND     4
AZ     4
AR     4
RI     4
CO     4
ME     4
OR     3
```

```
WV     3
DE     3
SD     3
MD     3
VT     2
MT     2
DC     2
NM     1
AK     1
NH     1
WY     1
Name: state, dtype: int64
```

Dsvid Gustafsson

Define Replace given name for rows in the `patients` table that have a given name of 'Dsvid' with 'David'.

Code

```
In [47]: patients_clean.given_name = patients_clean.given_name.replace('Dsvid', 'David')
```

Test

```
In [48]: patients_clean[patients_clean.surname == 'Gustafsson']
```

```
Out[48]:    patient_id assigned_sex given_name     surname             address  \
         8           9         male      David  Gustafsson  1790 Nutter Street

                 city state zip_code        country birthdate  weight  height   bmi  \
         8  Kansas City    MO    64105  United States  3/6/1937   163.9      66  26.5

            phone_number                          email
         8  816-265-9578  DavidGustafsson@armyspy.com
```

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 Convert assigned sex and state to categorical data types. Zip code data type was already addressed above. Convert birthdate to datetime data type. Strip the letter 'u' in start dose and end dose and convert those columns to data type integer.

Code

```
In [49]: # To category
         patients_clean.assigned_sex = patients_clean.assigned_sex.astype('category')
         patients_clean.state = patients_clean.state.astype('category')

         # To datetime
         patients_clean.birthdate = pd.to_datetime(patients_clean.birthdate)

         # Strip u and to integer
         treatments_clean.dose_start = treatments_clean.dose_start.str.strip('u').astype(int)
         treatments_clean.dose_end = treatments_clean.dose_end.str.strip('u').astype(int)
```

Test

```
In [50]: patients_clean.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503 entries, 0 to 502
Data columns (total 15 columns):
patient_id      503 non-null int64
assigned_sex    503 non-null category
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 category
zip_code        491 non-null object
country         491 non-null object
birthdate       503 non-null datetime64[ns]
weight          503 non-null float64
height          503 non-null int64
bmi             503 non-null float64
phone_number    491 non-null object
email           491 non-null object
dtypes: category(2), datetime64[ns](1), float64(2), int64(2), object(8)
memory usage: 53.9+ KB


In [51]: treatments_clean.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 349 entries, 0 to 348
Data columns (total 8 columns):
hba1c_start         349 non-null float64
hba1c_end           349 non-null float64
hba1c_change        349 non-null float64
treatment           349 non-null object
dose_start          349 non-null int64
dose_end            349 non-null int64
```

```
adverse_reaction    35 non-null object
patient_id          349 non-null int64
dtypes: float64(3), int64(3), object(2)
memory usage: 24.5+ KB
```

Multiple phone number formats

Define Strip all " ", "-", "(", ")", and "+" and store each number without any formatting. Pad the phone number with a 1 if the length of the number is 10 digits (we want country code).

Code

```
In [52]: patients_clean.phone_number = patients_clean.phone_number.str.replace(r'\D+', '').str.p
```

Test

```
In [53]: patients_clean.phone_number.head()

Out[53]: 0    19517199170
         1    12175693204
         2    14023636804
         3    17326368246
         4    13345157487
         Name: phone_number, dtype: object
```

Default John Doe data

Define Remove the non-recoverable John Doe records from the `patients` table.

Code

```
In [54]: patients_clean = patients_clean[patients_clean.surname != 'Doe']
```

Test

```
In [55]: # Should be no Doe records
         patients_clean.surname.value_counts()

Out[55]: Jakobsen         3
         Taylor           3
         Hueber           2
         Souza            2
         Lund             2
         Gersten          2
         Liu              2
         Lng              2
         Silva            2
```

```
Kowalczyk        2
Johnson          2
Batukayev        2
Parker           2
Kadyrov          2
Schiavone        2
Nilsen           2
Lâm              2
Tucker           2
Woniak           2
Aranda           2
Ogochukwu        2
Grímsdóttir      2
T                2
Cabrera          2
Cindri           2
Collins          2
Bùi              2
Correia          2
Dratchev         2
Berg             2
                 ..
Adamski          1
Chinedum         1
Musliyevich      1
Bowden           1
Murakami         1
Miller           1
Wellish          1
Haraguchi        1
van der Lubbe    1
Combs            1
Gustafsson       1
Bonami           1
Borgen           1
McGregor         1
Sung             1
Kalb             1
Petersen         1
Bois             1
Fodor            1
Ferrari          1
Johnsrud         1
Mattila          1
Salib            1
Ferguson         1
Mathiesen        1
Werner           1
```

```
Lange            1
Fomin            1
Ruais            1
Sakai            1
Name: surname, Length: 465, dtype: int64
```

In [56]:
```
# Should be no 123 Main Street records
patients_clean.address.value_counts()
```

Out[56]:
```
648 Old Dear Lane           2
2476 Fulton Street          2
2778 North Avenue           2
1962 Cabell Avenue          1
2816 Ashford Drive          1
2333 Hidden Pond Road       1
3688 Adonais Way            1
2235 Catherine Drive        1
633 Better Street           1
4792 Maud Street            1
465 Southern Street         1
962 George Street           1
353 Whaley Lane             1
4519 Sussex Court           1
3094 Oral Lake Road         1
2549 Pearlman Avenue        1
2356 Myra Street            1
3391 Marcus Street          1
1525 Crestview Terrace      1
4104 Kennedy Court          1
1846 Joseph Street          1
321 Briercliff Road         1
57 Norman Street            1
2866 Myra Street            1
2945 Ferguson Street        1
3165 Upton Avenue           1
312 Jim Rosa Lane           1
4015 Juniper Drive          1
1510 Allison Avenue         1
3686 Meadowcrest Lane       1
                           ..
1965 Crestview Manor        1
1343 Clair Street           1
2531 Cantebury Drive        1
2168 Butternut Lane         1
2055 Emeral Dreams Drive    1
1731 Chandler Drive         1
108 Griffin Street          1
2152 Heritage Road          1
```

```
602 Tator Patch Road        1
1368 Yorkshire Circle       1
4839 North Avenue           1
1463 Beechwood Avenue       1
3072 Braxton Street         1
1012 Lords Way              1
4476 Center Street          1
4237 Hamilton Drive         1
2970 Forest Avenue          1
2127 Elk City Road          1
2121 Liberty Avenue         1
163 Hide A Way Road         1
3683 Gorby Lane             1
441 Tibbs Avenue            1
3595 Stuart Street          1
550 Cliffside Drive         1
1251 Clarence Court         1
4643 Reeves Street          1
4168 Coventry Court         1
4649 Joanne Lane            1
4213 Isaacs Creek Road      1
36 Heather Sees Way         1
Name: address, Length: 482, dtype: int64
```

Multiple records for Jakobsen, Gersten, Taylor

Define Remove the Jake Jakobsen, Pat Gersten, and Sandy Taylor rows from the `patients` table. These are the nicknames, which happen to also not be in the `treatments` table (removing the wrong name would create a consistency issue between the `patients` and `treatments` table). These are all the second occurrence of the duplicate. These are also the only occurences of non-null duplicate addresses.

Code

```
In [57]: # tilde means not: http://pandas.pydata.org/pandas-docs/stable/indexing.html#boolean-in
         patients_clean = patients_clean[~((patients_clean.address.duplicated()) & patients_clea
```

Test

```
In [58]: patients_clean[patients_clean.surname == 'Jakobsen']
```

```
Out[58]:      patient_id assigned_sex given_name   surname             address  \
         24           25         male      Jakob  Jakobsen   648 Old Dear Lane
         432         433       female      Karen  Jakobsen  1690 Fannie Street

                    city state zip_code        country  birthdate  weight  height  \
         24   Port Jervis    NY    12771  United States 1985-08-01   155.8      67
         432      Houston    TX    77020  United States 1962-11-25   185.2      67
```

```
      bmi phone_number                         email
24   24.4  18458587707     JakobCJakobsen@einrot.com
432  29.0  19792030438  KarenJakobsen@jourrapide.com
```

```
In [59]: patients_clean[patients_clean.surname == 'Gersten']
```

```
Out[59]:     patient_id assigned_sex given_name  surname           address  city  \
97           98         male    Patrick  Gersten  2778 North Avenue  Burr

    state zip_code        country  birthdate  weight  height   bmi  \
97     NE    68324  United States 1954-05-03   138.2      71  19.3

    phone_number                      email
97   14028484923  PatrickGersten@rhyta.com
```

```
In [60]: patients_clean[patients_clean.surname == 'Taylor']
```

```
Out[60]:     patient_id assigned_sex given_name surname             address      city  \
131          132       female     Sandra  Taylor  2476 Fulton Street  Rainelle
426          427         male    Rogelio  Taylor  4064 Marigold Lane     Miami

    state zip_code        country  birthdate  weight  height   bmi  \
131    WV    25962  United States 1960-10-23   206.1      64  35.4
426    FL    33179  United States 1992-09-02   186.6      69  27.6

    phone_number                        email
131  13044382648    SandraCTaylor@dayrep.com
426  13054346299  RogelioJTaylor@teleworm.us
```

kgs instead of lbs for Zaitseva weight

Define Use advanced indexing to isolate the row where the surname is Zaitseva and convert the entry in its weight field from kg to lbs.

Code

```
In [61]: weight_kg = patients_clean.weight.min()
         mask = patients_clean.surname == 'Zaitseva'
         column_name = 'weight'
         patients_clean.loc[mask, column_name] = weight_kg * 2.20462
```

Test

```
In [62]: # 48.8 shouldn't be the lowest anymore
         patients_clean.weight.sort_values()
```

```
Out[62]: 459    102.100000
         335    102.700000
         74     103.200000
         317    106.000000
         171    106.500000
         51     107.100000
         270    108.100000
         198    108.500000
         48     109.100000
         478    109.600000
         141    110.200000
         38     111.800000
         438    112.000000
         14     112.000000
         235    112.200000
         307    112.400000
         191    112.600000
         408    113.100000
         49     113.300000
         326    114.000000
         338    114.100000
         253    117.000000
         321    118.400000
         168    118.800000
         1      118.800000
         350    119.000000
         207    119.200000
         265    120.000000
         341    120.300000
         208    121.200000
                   ...
         12     224.200000
         252    224.200000
         222    224.800000
         166    225.300000
         111    225.900000
         101    226.200000
         150    226.600000
         88     227.700000
         352    227.700000
         428    227.700000
         13     228.400000
         339    229.000000
         182    230.300000
         121    230.800000
         257    231.700000
         395    231.900000
         246    232.100000
```

```
219    237.800000
11     238.700000
50     238.900000
441    239.100000
499    239.600000
439    242.000000
487    242.400000
61     244.900000
144    244.900000
283    245.500000
118    254.500000
485    255.900000
210    268.302254
Name: weight, Length: 494, dtype: float64
```