Introduction

In this tutorial, you'll learn how to investigate data types within a DataFrame or Series. You'll also learn how to find and replace entries.

Dtypes

The data type for a column in a DataFrame or a Series is known as the **dtype**.

You can use the `dtype` property to grab the type of a specific column. For instance, we can get the dtype of the `price` column in the `reviews` DataFrame:

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

reviews = pd.read_csv("datasets/winemag-data-130k-v2.csv", index_col=0)
reviews
```

Out[1]:

	country	description	designation	points	price	province	region_1	regior
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	N
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	NaN	N
2	US	Tart and snappy, the flavors of lime flesh and...	NaN	87	14.0	Oregon	Willamette Valley	Willame Va
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	N
4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Oregon	Willamette Valley	Willame Va
...	...	...	...	...	...	...	...	
65494	France	Made from young vines from the Vaulorent porti...	Fourchaume Premier Cru	90	45.0	Burgundy	Chablis	N
65495	Australia	This is a big, fat, almost sweet-tasting Caber...	NaN	90	22.0	South Australia	McLaren Vale	N
65496	US	Much improved over the unripe 2005, Fritz's 20...	Estate	90	20.0	California	Dry Creek Valley	Sonc
65497	US	This wine wears its 15.8% alcohol	Block 24	90	31.0	California	Napa Valley	Na

	country	description	designation	points	price	province	region_1	regior
		better than ...						
65498	Spain	A unique take on Manzanilla Sherry, which is o...	Manzanilla	90	10.0	Andalucia	Jerez	N

65499 rows × 13 columns

In [2]:
```
reviews.price.dtype
```

Out[2]:
```
dtype('float64')
```

Alternatively, the `dtypes` property returns the `dtype` of *every* column in the DataFrame:

In [3]:
```
reviews.dtypes
```

Out[3]:
```
country                  object
description              object
designation              object
points                    int64
price                   float64
province                 object
region_1                 object
region_2                 object
taster_name              object
taster_twitter_handle    object
title                    object
variety                  object
winery                   object
dtype: object
```

Data types tell us something about how pandas is storing the data internally. `float64` means that it's using a 64-bit floating point number; `int64` means a similarly sized integer instead, and so on.

One peculiarity to keep in mind (and on display very clearly here) is that columns consisting entirely of strings do not get their own type; they are instead given the `object` type.

It's possible to convert a column of one type into another wherever such a conversion makes sense by using the `astype()` function. For example, we may transform the `points` column from its existing `int64` data type into a `float64` data type:

In [4]:
```
reviews.points.astype('float64')
```

Out[4]:
```
0        87.0
1        87.0
2        87.0
3        87.0
4        87.0
         ...
65494    90.0
65495    90.0
65496    90.0
65497    90.0
65498    90.0
Name: points, Length: 65499, dtype: float64
```

A DataFrame or Series index has its own `dtype`, too:

In [5]:
```
reviews.index.dtype
```

Out[5]:
```
dtype('int64')
```

Pandas also supports more exotic data types, such as categorical data and timeseries data. Because these data types are more rarely used, we will omit them until a much later section of this tutorial.

Missing data

Entries missing values are given the value `NaN`, short for "Not a Number". For technical reasons these `NaN` values are always of the `float64` dtype.

Pandas provides some methods specific to missing data. To select `NaN` entries you can use `pd.isnull()` (or its companion `pd.notnull()`). This is meant to be used thusly:

In [6]:
```
reviews.region_2
```

Out[6]:
```
0                    NaN
1                    NaN
2        Willamette Valley
3                    NaN
4        Willamette Valley
               ...
65494                NaN
65495                NaN
65496             Sonoma
65497               Napa
65498                NaN
Name: region_2, Length: 65499, dtype: object
```

In [7]:
```
reviews[pd.isnull(reviews.country)]
```

Out[7]:

	country	description	designation	points	price	province	region_1	region_
913	NaN	Amber in color, this wine has aromas of peach ...	Asureti Valley	87	30.0	NaN	NaN	Na
3131	NaN	Soft, fruity and juicy, this is a pleasant, si...	Partager	83	NaN	NaN	NaN	Na
4243	NaN	Violet-red in color, this semisweet wine has a...	Red Naturally Semi-Sweet	88	18.0	NaN	NaN	Na
9509	NaN	This mouthwatering blend starts with a nose of...	Theopetra Malagouzia-Assyrtiko	92	28.0	NaN	NaN	Na
9750	NaN	This orange-style wine has a cloudy yellow-gol...	Orange Nikolaevo Vineyard	89	28.0	NaN	NaN	Na
11150	NaN	A blend of 85% Melnik, 10% Grenache Noir and 5...	NaN	89	20.0	NaN	NaN	Na
11348	NaN	Light and fruity, this is a wine that has some...	Partager	82	NaN	NaN	NaN	Na
14030	NaN	This Furmint, grown in marl soils, has aromas ...	Márga	88	25.0	NaN	NaN	Na
16000	NaN	Jumpy, jammy aromas of foxy black fruits are s...	Valle de los Manantiales Vineyard	86	40.0	NaN	NaN	Na
16749	NaN	Winemaker: Bartho Eksteen. This wooded Sauvy s...	Cape Winemakers Guild Vloekskoot Wooded	91	NaN	NaN	NaN	Na
18075	NaN	Delicate white flowers and a spin of lemon pee...	Askitikos	90	17.0	NaN	NaN	Na

	country	description	designation	points	price	province	region_1	region
26485	NaN	This wine has aromas of black berry, dried red...	NaN	87	13.0	NaN	NaN	Na
26486	NaN	Aromas of green apple and white flowers prepar...	NaN	87	14.0	NaN	NaN	Na
26489	NaN	Balanced aromas of green herbs and citrus zest...	Aliwen Reserva	87	12.0	NaN	NaN	Na
27822	NaN	This is a reasonably rich, concentrated exampl...	NaN	86	19.0	NaN	NaN	Na
36112	NaN	An interesting blend of indigenous Bulgarian a...	Hrumki Melnik 55 Mourvèdre Marselan	89	25.0	NaN	NaN	Na
38240	NaN	Subdued citrus and pear notes on the nose find...	Steirische Klassik	89	24.0	NaN	NaN	Na
38898	NaN	Scents of clover, stem, green herb and red cur...	Wismer-Parke Vineyard	89	34.0	NaN	NaN	Na
44674	NaN	Crisp apple freshness almost tips into full ci...	Steirische Klassik	91	25.0	NaN	NaN	Na
44850	NaN	This blend of Gamay and Prokupe has aromas of ...	Amphora	84	6.0	NaN	NaN	Na
44851	NaN	This wine has aromas of honeysuckle and lemon ...	Royal	84	6.0	NaN	NaN	Na
45247	NaN	Just a whiff of citrus shows on the restrained...	Steirische Klassik	89	25.0	NaN	NaN	Na

	country	description	designation	points	price	province	region_1	region_
45402	NaN	Basic cherry aromas turn more earthy and soupy...	Reserva Estate Bottled	85	12.0	NaN	NaN	Na
46352	NaN	A dark color and rich, jammy, baked aromas of ...	Catalina	91	50.0	NaN	NaN	Na
49425	NaN	This blend is comprised of 55% Merlot, 21% Cab...	Getika Made With Organic Grapes	88	28.0	NaN	NaN	Na
49426	NaN	Enticing aromas of blueberry syrup open this b...	Getika Made With Organic Grapes	88	28.0	NaN	NaN	Na
49427	NaN	This dark-garnet wine has aromas of eucalyptus...	Hrumki Syrah Melnik 55 Mourvèdre Marselan	88	19.0	NaN	NaN	Na
49510	NaN	Aromas of cherry, blueberry and rose petal pre...	NaN	91	34.0	NaN	NaN	Na
54222	NaN	Almost caramel in color, this wine offers arom...	Babaneuri Valley	87	30.0	NaN	NaN	Na
57612	NaN	Winemaker: Gordon Newton Johnson. This is such...	Cape Winemakers Guild Windansea	92	NaN	NaN	NaN	Na
59670	NaN	The heady florality of damask rose is joined b...	Steintal	92	38.0	NaN	NaN	Na
60678	NaN	This wine was made for grilled meats, with its...	Dry	86	17.0	NaN	NaN	Na

Replacing missing values is a common operation. Pandas provides a really handy method for this problem: `fillna()`. `fillna()` provides a few different strategies for mitigating such data. For example, we can simply replace each `NaN` with an `"Unknown"`:

In [8]:
```
reviews.region_2.fillna("Unknown")
```

Out[8]:
```
0                  Unknown
1                  Unknown
2        Willamette Valley
3                  Unknown
4        Willamette Valley
               ...
65494              Unknown
65495              Unknown
65496               Sonoma
65497                 Napa
65498              Unknown
Name: region_2, Length: 65499, dtype: object
```

Or we could fill each missing value with the first non-null value that appears sometime after the given record in the database. This is known as the backfill strategy.

Alternatively, we may have a non-null value that we would like to replace. For example, suppose that since this dataset was published, reviewer Kerin O'Keefe has changed her Twitter handle from `@kerinokeefe` to `@kerino`. One way to reflect this in the dataset is using the `replace()` method:

In [9]:
```
reviews.taster_twitter_handle.replace("@kerinokeefe", "@kerino")
```

Out[9]:
```
0           @kerino
1         @vossroger
2        @paulgwine 
3                NaN
4        @paulgwine 
            ...
65494     @vossroger
65495         @JoeCz
65496            NaN
65497            NaN
65498    @wineschach
Name: taster_twitter_handle, Length: 65499, dtype: object
```

The `replace()` method is worth mentioning here because it's handy for replacing missing data which is given some kind of sentinel value in the dataset: things like `"Unknown"`, `"Undisclosed"`, `"Invalid"`, and so on.

Your turn

If you haven't started the exercise, you can start now.