

# pandas Dataframes - Examining Data

---

## lesson\_2\_2\_1

Import packages

```
import pandas as pd
```

Creating a Basic Dataframe From JSON

```
# define the data as a list
data = [
    ("Dexter","Johnsons","dog","shiba inu","red
sesame",1.5,35,"m",False,"both",True),
    ("Alfred","Johnsons","cat","mix","tuxedo",4,12,"m",True,"indoor",True),
    ("Petra","Smith","cat","ragdoll","calico",6,None,"f",False,"both",True),
    ("Ava","Smith","dog","mix","blk/wht",12,32,"f",True,"both",False),
    ("Schroder","Brown","cat","mix","orange",13,15,"m",False,"indoor",True),
    ("Blackbeard","Brown","bird","parrot","multi",5,3,"f",False,"indoor",),
]

# define the labels
labels =
["name","owner","type","breed","color","age","weight","gender","health
issues","indoor/outdoor","vaccinated"]

# create dataframe
vet_records = pd.DataFrame.from_records(data, columns=labels)
```

Examining the Data in a Dataframe

There are several different ways to examine data using a pandas dataframe. Two are `.head()` and `.tail()`. These show the first five and the last five rows of the dataframe respectively.

```
# displays the first five rows in the dataframe
vet_records.head()
```

```
# displays the first five rows in the dataframe
vet_records.tail()
```

```
# displays all the records of the dataframe
vet_records
```

**.dtypes** show you the types of data in the dataframe by column. If the **dtype** is **object**, this indicates that pandas is seeing that data as more than one type.

```
# object means a mixed type column
vet_records.dtypes
```

Notice all the **string** columns are listed as **object**. This is because a **string** type takes a maximum length argument, so when importing from CSV, they are imported as a **object** so they can be variable length.

**.describe** shows statistical operations on columns that these operations can be performed on.

```
# `.describe` shows statistical information on columns that the operations
can be performed on
vet_records.describe()
```

```
# to show all columns in `.describe` add `include="all"`
vet_records.describe(include="all")
```

**.at** allows the user to change the value of a specific cell

```
# change a specific value with `.at`
vet_records.at[0, "weight"] = 34.7
```

```
# notice the weight was changed for Dexter
vet_records
```

**.assign** is used to add another column of data

```
# we are going to add the ratio age:weight as a column to the dataframe
# notice that this method iterates through the dataframe
vet_records = vet_records.assign(age_weight=
(vet_records['age']/vet_records['weight']))
```

```
# review the new dataframe
vet_records
```