

# unique

October 17, 2019

```
[3]: import pandas as pd
csv_path="https://people.sc.fsu.edu/~jburkardt/data/csv/biostats.csv"
df=pd.read_csv(csv_path)
df
```

```
[3]:
```

	Name	"Sex"	"Age"	"Height (in)"	"Weight (lbs)"
0	Alex	"M"	41	74	170
1	Bert	"M"	42	68	166
2	Carl	"M"	32	70	155
3	Dave	"M"	39	72	167
4	Elly	"F"	30	66	124
5	Fran	"F"	33	66	115
6	Gwen	"F"	26	64	121
7	Hank	"M"	30	71	158
8	Ivan	"M"	53	72	175
9	Jake	"M"	32	69	143
10	Kate	"F"	47	69	139
11	Luke	"M"	34	72	163
12	Myra	"F"	23	62	98
13	Neil	"M"	36	75	160
14	Omar	"M"	38	70	145
15	Page	"F"	31	67	135
16	Quin	"M"	29	71	176
17	Ruth	"F"	28	65	131

```
[6]: df["Age"].unique()
```

```

      □
↳ -----
KeyError                                Traceback (most recent call↳
↳ last)

<ipython-input-6-10090c31e9de> in <module>
----> 1 df[["Age"]].unique()
```

```

~/conda/envs/python/lib/python3.6/site-packages/pandas/core/frame.py in
↳ __getitem__(self, key)
    2984         if is_iterator(key):
    2985             key = list(key)
-> 2986         indexer = self.loc._convert_to_indexer(key, axis=1,
↳ raise_missing=True)
    2987
    2988         # take() does not accept boolean indexers

```

```

~/conda/envs/python/lib/python3.6/site-packages/pandas/core/indexing.py
↳ in _convert_to_indexer(self, obj, axis, is_setter, raise_missing)
    1283         # When setting, missing keys are not allowed, even
↳ with .loc:
    1284         kwargs = {"raise_missing": True if is_setter else
↳ raise_missing}
-> 1285         return self._get_listlike_indexer(obj, axis,
↳ **kwargs)[1]
    1286     else:
    1287         try:

```

```

~/conda/envs/python/lib/python3.6/site-packages/pandas/core/indexing.py
↳ in _get_listlike_indexer(self, key, axis, raise_missing)
    1090
    1091         self._validate_read_indexer(
-> 1092             keyarr, indexer, o._get_axis_number(axis),
↳ raise_missing=raise_missing
    1093         )
    1094         return keyarr, indexer

```

```

~/conda/envs/python/lib/python3.6/site-packages/pandas/core/indexing.py
↳ in _validate_read_indexer(self, key, indexer, axis, raise_missing)
    1175         raise KeyError(
    1176             "None of [{key}] are in the [{axis}]"
-> 1177             .format(key=key, axis=self.obj._get_axis_name(axis)
    1178             )
    1179         )

```

KeyError: "None of [Index(['Age'], dtype='object')] are in the [columns]"

[ ]: