Pokemon

Introduction:

This time you will create the data.

Step 1. Import the necessary libraries

```
import pandas as pd
```

Step 2. Create a data dictionary that looks like the DataFrame below

```
data = {
    "evolution": ["Ivysaur", "Charmeleon", "Wartortle", "Metapod"],
    "hp": [45, 39, 44, 45],
    "name": ["Bulbasaur", "Charmander", "Squirtle", "Caterpie"],
    "pokedex": ["yes", "no", "yes", "no"],
}
```

Step 3. Assign it to a variable called pokemon

```
pokemon = pd.DataFrame(data)

print(pokemon)
```

```
     evolution  hp        name pokedex
0      Ivysaur  45   Bulbasaur     yes
1   Charmeleon  39  Charmander      no
2    Wartortle  44    Squirtle     yes
3      Metapod  45    Caterpie      no
```

Step 4. Ops...it seems the DataFrame columns are in alphabetical order. Place the order of the columns as name, type, hp, evolution, pokedex

```
pokemon = pokemon.sort_values(by="evolution")
print(pokemon)
```

```
     evolution  hp        name pokedex
1   Charmeleon  39  Charmander      no
0      Ivysaur  45   Bulbasaur     yes
3      Metapod  45    Caterpie      no
2    Wartortle  44    Squirtle     yes
```

```
pokemon  = pokemon.sort_values(by="name")
print(pokemon)
```

```
     evolution  hp        name pokedex
0      Ivysaur  45   Bulbasaur     yes
3      Metapod  45    Caterpie      no
1   Charmeleon  39  Charmander      no
2    Wartortle  44    Squirtle     yes
```

```
pokemon = pokemon.sort_values(by="hp")
print(pokemon)
```

```
     evolution  hp        name pokedex
1   Charmeleon  39  Charmander      no
2    Wartortle  44    Squirtle     yes
3      Metapod  45    Caterpie      no
0      Ivysaur  45   Bulbasaur     yes
```

```
pokemon = pokemon.sort_values(by="pokedex")
print(pokemon)
```

```
     evolution  hp        name pokedex
1   Charmeleon  39  Charmander      no
3      Metapod  45    Caterpie      no
2    Wartortle  44    Squirtle     yes
0      Ivysaur  45   Bulbasaur     yes
```

```
pokemon = pokemon.sort_values(by="evolution", ascending=False).reset_index(drop=True)
print(pokemon)
```

```
     evolution  hp        name pokedex
0   Wartortle  44    Squirtle     yes
1     Metapod  45    Caterpie      no
2     Ivysaur  45   Bulbasaur     yes
3  Charmeleon  39  Charmander      no
```

Step 5. Add another column called place, and insert what you have in mind.

```
pokemon["place"] = ["forest", "desert", "lake", "cave"]
print(pokemon)
```

```
     evolution  hp        name pokedex   place
0   Wartortle  44    Squirtle     yes  forest
1     Metapod  45    Caterpie      no  desert
2     Ivysaur  45   Bulbasaur     yes    lake
3  Charmeleon  39  Charmander      no    cave
```

Step 6. Present the type of each column

```
print(pokemon.dtypes)
```

```
evolution    object
hp            int64
name         object
pokedex      object
place        object
dtype: object
```

BONUS: Create your own question and answer it.

How to read data from pokemon.csv and display all Pokémon with hp greater than or equal to 45?

```
import pandas as pd


pokemon = pd.read_csv("pokemon.csv")

pokemon = pokemon[pokemon["HP"] >= 45]

print(pokemon)
```

```
        #                   Name   Type 1  Type 2  Total  HP  Attack  Defense  \
0       1              Bulbasaur    Grass  Poison    318  45      49       49
1       2                Ivysaur    Grass  Poison    405  60      62       63
2       3               Venusaur    Grass  Poison    525  80      82       83
3       3  VenusaurMega Venusaur    Grass  Poison    625  80     100      123
5       5             Charmeleon     Fire     NaN    405  58      64       58
..    ...                    ...      ...     ...    ...  ..     ...      ...
795   719                Diancie     Rock   Fairy    600  50     100      150
796   719    DiancieMega Diancie     Rock   Fairy    700  50     160      110
797   720    HoopaHoopa Confined  Psychic   Ghost    600  80     110       60
798   720     HoopaHoopa Unbound  Psychic    Dark    680  80     160       60
799   721              Volcanion     Fire   Water    600  80     110      120

     Sp. Atk  Sp. Def  Speed  Generation  Legendary
0         65       65     45           1      False
1         80       80     60           1      False
2        100      100     80           1      False
3        122      120     80           1      False
5         80       65     80           1      False
..       ...      ...    ...         ...        ...
795      100      150     50           6       True
796      160      110    110           6       True
797      150      130     70           6       True
798      170      130     80           6       True
799      130       90     70           6       True

[698 rows x 13 columns]
```