

# Loading a DataFrame

## Exercise

We're still working hard to solve the kidnapping of Bayes, the Golden Retriever. Previously, we used a license plate spotted at the crime scene to narrow the list of suspects to:

Fred Frequentist

Ronald Aylmer Fisher

Gertrude Cox

Kirstine Smith

We've obtained credit card records for all four suspects. Perhaps some of them made suspicious purchases before the kidnapping?

The records are in a CSV called "credit\_records.csv".

## Instructions

- 1-Import the pandas module under the alias pd.
- 2-Load the CSV "credit\_records.csv" into a DataFrame called credit\_records.
- 3-Display the first five rows of credit\_records using the .head() method.

In [ ]:

```
# Import pandas under the alias pd
____

# Load the CSV "credit_records.csv"
credit_records = ____

# Display the first five rows of credit_records using the .head() method
print(____)

#_____#
#Solutions

# Import pandas under the alias pd
import pandas as pd

# Load the CSV "credit_records.csv"
credit_records = pd.read_csv("credit_records.csv")

# Display the first five rows of credit_records using the .head() method
print(credit_records.head())
```

# Inspecting a DataFrame

## Exercise

We've loaded the credit card records of our four suspects into a DataFrame called `credit_records`. Let's learn more about the structure of this DataFrame.

The pandas module has been imported under the alias `pd`. The DataFrame `credit_records` has already been imported.

How many rows are in `credit_records`?

## Instructions

Use the `.info()` method to inspect the DataFrame `credit_records`

In [ ]:

```
#Use .info() to inspect the DataFrame credit_records
print(...)
```

---

```
#_____#
#Solutions
```

---

```
#Use .info() to inspect the DataFrame credit_records
print(credit_records.info())
```

# Two methods for selecting columns

## Exercise

Once again, we've loaded the credit card records of our four suspects into a DataFrame called `credit_records`. Let's examine the items that they've purchased.

The pandas module has been imported under the alias `pd`. The DataFrame `credit_records` has already been imported.

## Instructions

1-Select the column `item` from `credit_records` using brackets and string notation.

2-Select the column `item` from `credit_records` using dot notation.

In [ ]:

```
# Select the column item from credit_records
# Use brackets and string notation
items = ____

# Display the results
print(items)

# Select the column item from credit_records
# Use dot notation
items = ____

# Display the results
print(items)

#_____#
#Solutions

# Select the column item from credit_records
# Use brackets and string notation
items = credit_records['item']

# Display the results
print(items)

# Select the column item from credit_records
# Use dot notation
items = credit_records.item

# Display the results
print(items)
```

## Correcting column selection errors

### Exercise

A junior detective tried to access the location columns of `credit_records`, but he made some mistakes. Help correct his code so that we can search for suspicious purchases.

In all exercises going forward, pandas will be imported as `pd`. The DataFrame `credit_records` has already been imported.

### Instructions

Correct the code so that it runs without errors.

In [ ]:

```
# One or more lines of code contain errors.
# Fix the errors so that the code runs.

# Select the location column in credit_records
location = credit_records[location]

# Select the item column in credit_records
items = credit_records."item"

# Display results
print(location)

#_____#
#Solutions

# One or more lines of code contain errors.
# Fix the errors so that the code runs.

# Select the location column in credit_records
location = credit_records['location']

# Select the item column in credit_records
items = credit_records.item

# Display results
print(location)
```

## More column selection mistakes

### Exercise

Another junior detective is examining a DataFrame of Missing Puppy Reports. He's made some mistakes that cause the code to fail.

The pandas module has been loaded under the alias `pd`, and the DataFrame is called `mpr`.

### Instructions

- 1-Inspect the DataFrame `mpr` using `info()`.
- 2-Correct the mistakes in the code so that it runs without errors.

In [ ]:

```
# Use info() to inspect mpr
print(mpr.____)

# The following code contains one or more errors
# Correct the mistakes in the code so that it runs without errors

# Select column "Dog Name" from mpr
name = mpr.Dog Name

# Select column "Missing?" from mpr
is_missing = mpr.Missing?

# Display the columns
print(name)
print(is_missing)

#_____#
#Solutions

# Use info() to inspect mpr
print(mpr.info())

# The following code contains one or more errors
# Correct the mistakes in the code so that it runs without errors

# Select column "Dog Name" from mpr
name = mpr['Dog Name']

# Select column "Missing?" from mpr
is_missing = mpr['Missing?']

# Display the columns
print(name)
print(is_missing)
```

# Logical testing

## Exercise

Let's practice writing logical statements and displaying the output.

Recall that we use the following operators:

`==` tests that two values are equal.

`!=` tests that two values are not equal.

**(>) and (<) test that greater than or less than, respectively.**

**(>=) and (<=) test greater than or equal to or less than or equal to, respectively.**

## Instructions

1-The variable `height_inches` represents the height of a suspect. Is `height_inches` greater than 70 inches?

2-The variable `plate1` represents a license plate number of a suspect. Is it equal to `FRQ123`?

3 The variable `fur_color` represents the color of Bayes' fur. Check that `fur_color` is not equal to "brown"

In [ ]:

```
# Is height_inches greater than 70 inches?
print(____ ____ 70)

# Is plate1 equal to "FRQ123"?
print(____ ____ "FRQ123")

# Is fur_color not equal to "brown"?
print(____ ____ "brown")

#_____#
#Solutions

# Is height_inches greater than 70 inches?
print(height_inches > 70)

# Is plate1 equal to "FRQ123"?
print(plate1 == "FRQ123")

# Is fur_color not equal to "brown"?
print(fur_color != "brown")
```

# Selecting missing puppies

## Exercise

Let's return to our DataFrame of missing puppies, which is loaded as mpr. Let's select a few different rows to learn more about the other missing dogs.

## Instructions

- 1-Select the dogs where Age is greater than 2.
- 2-Select the dogs whose Status is equal to Still Missing.
- 3-Select all dogs whose Dog Breed is not equal to Poodle.

In [ ]:

```
# Select the dogs where Age is greater than 2
greater_than_2 = mpr[mpr.Age ____ 2]
print(greater_than_2)

# Select the dogs whose Status is equal to Still Missing
still_missing = mpr[____ ____ ____]
print(still_missing)

# Select all dogs whose Dog Breed is not equal to Poodle
not_poodle = ____
print(not_poodle)

#_____#
#Solutions

# Select the dogs where Age is greater than 2
greater_than_2 = mpr[mpr.Age > 2]
print(greater_than_2)

# Select the dogs whose Status is equal to Still Missing
still_missing = mpr[mpr['Status'] == 'Still Missing']
print(still_missing)

# Select all dogs whose Dog Breed is not equal to Poodle
not_poodle = mpr[mpr['Dog Breed'] != 'Poodle']
print(not_poodle)
```

# Narrowing the list of suspects

## Exercise

In Chapter 1, we found a list of people whose cars matched the description of the one that kidnapped Bayes:

Fred Frequentist

Ronald Aylmer Fisher

Gertrude Cox

Kirstine Smith

We'd like to narrow this list down, so we obtained credit card records for each suspect. We'd like to know if any of them recently purchased dog treats to use in the kidnapping. If they did, they would have visited 'Pet Paradise'.

The credit records have been loaded into a DataFrame called `credit_records`

## Instructions

Select rows of `credit_records` such that the column `location` is equal to 'Pet Paradise'.

In [ ]:

```
# Select purchases from 'Pet Paradise'
purchase = ____[____.____ ____ 'Pet Paradise']

# Display
print(purchase)

#_____#
#Solutions

# Select purchases from 'Pet Paradise'
purchase = credit_records[credit_records.location == 'Pet Paradise']

# Display
print(purchase)
```