Berkeley Engineering | BerkeleyHaas

PROFESSIONAL CERTIFICATE IN MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Module 4

Fundamentals of Data Analysis

Office Hours with Viviana Márquez
September 21, 2023

AGENDA

- Slack
- Required activities for Module 4
- Content review Module 4: Fundamentals of Data Analytics
- Questions

Slack

#cohort-august-2023

Slack Workspace Invitation

Required Activities for Module 4

- 🤖 Codio Activity 4.1: BASIC JOINS ON DATASETS
- 🤖 Codio Activity 4.2: COMPLEX JOINS ON DATASETS
- 🤖 Codio Activity 4.3: CREATING SCATTERPLOTS, HISTOGRAMS, AND DISTRIBUTION PLOTS
- 🤖 Codio Activity 4.4: CREATING VIOLIN, BOX, AND JOINT PLOTS
- Try-It Activity 4.1: MORE SOPHISTICATED PLOTTING
- 🤖 Codio Activity 4.5: STRING OPERATIONS
- 🤖 Codoi Activity 4.6: DATA CLEANING
- Try-It Activity 4.2: ANALYZING A REAL-WORLD DATASET
- 🤖 Quiz 4.1: THE FUNDAMENTALS OF DATA ANALYSIS

Content review Module 4: Fundamentals Data Analysis

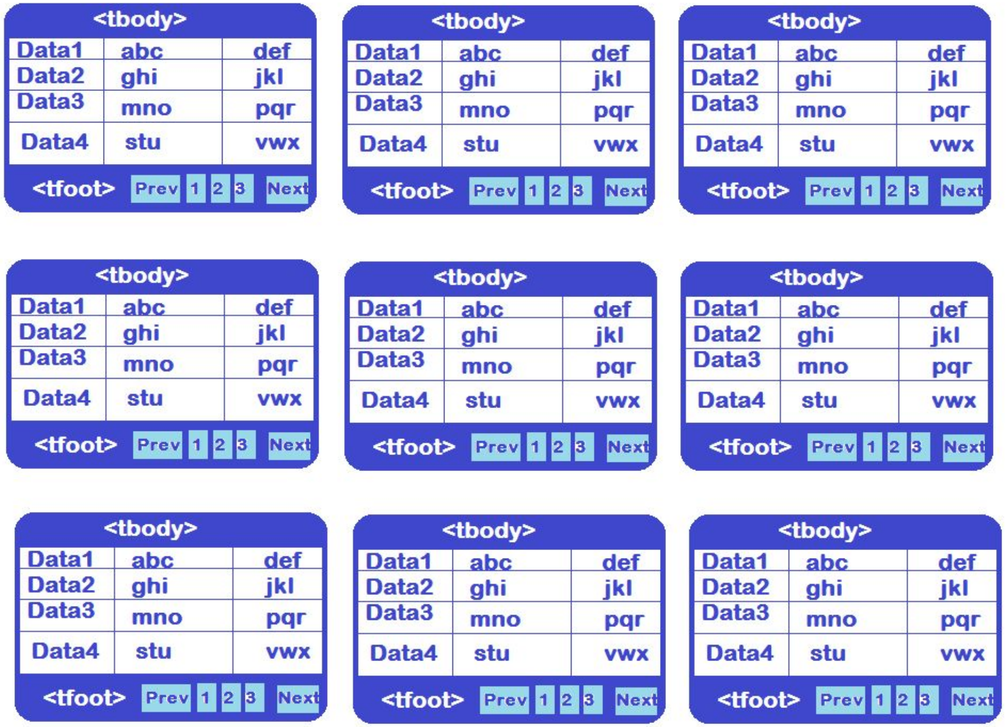
- Joining tables
- Data cleaning
- Code
    - Pandas merge
    - Data cleaning
        - String operations

Content review Module 4: Fundamentals Data Analysis

- Joining tables
- Data cleaning
- Code
  - Pandas merge
  - Data cleaning
    - String operations

Joining tables

- In the real-world, often data will be split across multiple tables
- You will have to combine records from those tables based on related columns



<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

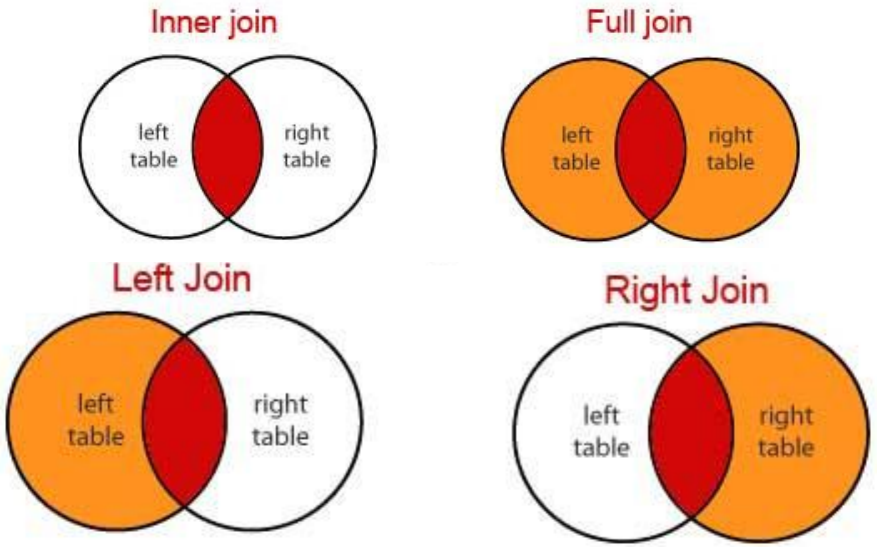
<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

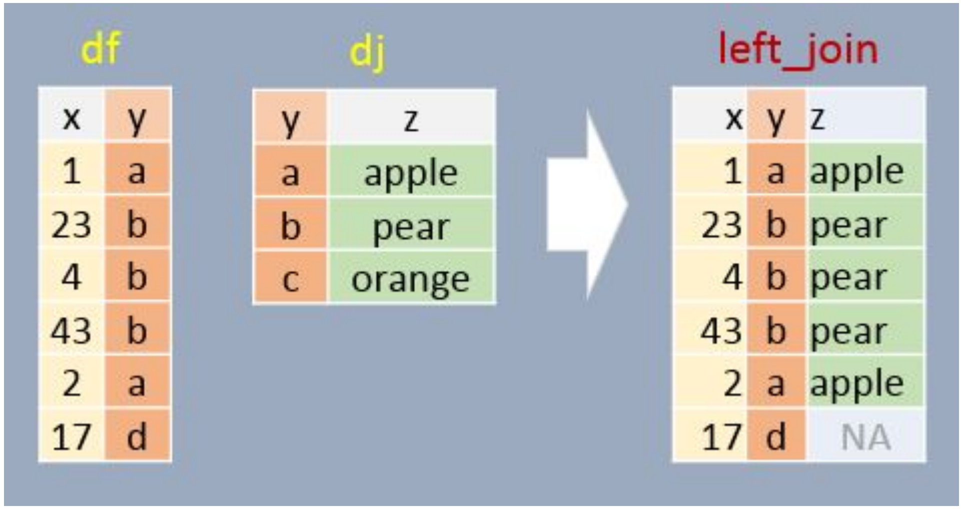
<tbody>		
Data1	abc	def
Data2	ghi	jkl
Data3	mno	pqr
Data4	stu	vwx
<tfoot>	Prev 1 2 3	Next

Types of joins

- **INNER JOIN**: Returns records that have matching values in both tables.
- **LEFT (OUTER) JOIN**: Returns all records from the left table, and the matched records from the right table.
- **RIGHT (OUTER) JOIN**: Returns all records from the right table, and the matched records from the left table.
- **FULL (OUTER) JOIN**: Returns all records when there is a match in either the left or the right table.
- **CROSS JOIN**: Returns the Cartesian product of the two tables.
- **SELF JOIN**: Joining a table with itself.



Left join



df

x	y
1	a
23	b
4	b
43	b
2	a
17	d

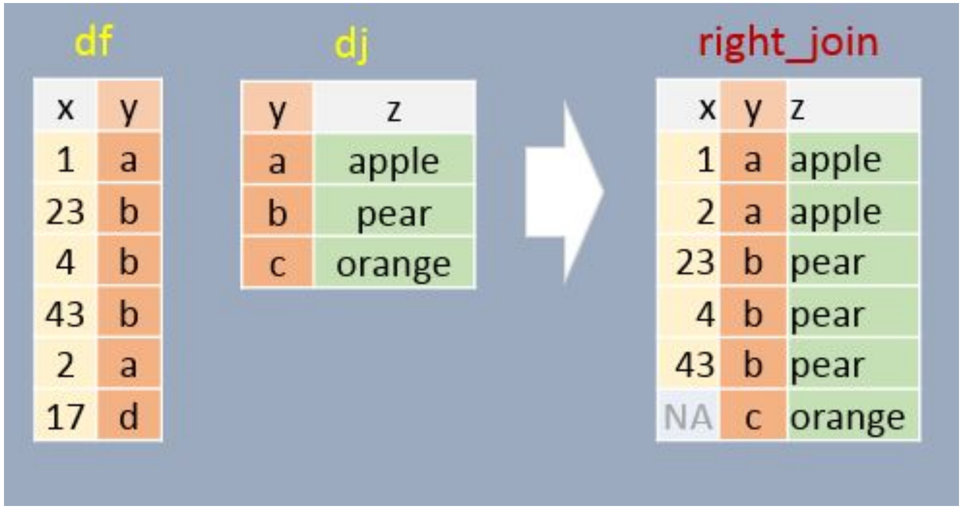
dj

y	z
a	apple
b	pear
c	orange

left_join

x	y	z
1	a	apple
23	b	pear
4	b	pear
43	b	pear
2	a	apple
17	d	NA

Right join



df

x	y
1	a
23	b
4	b
43	b
2	a
17	d

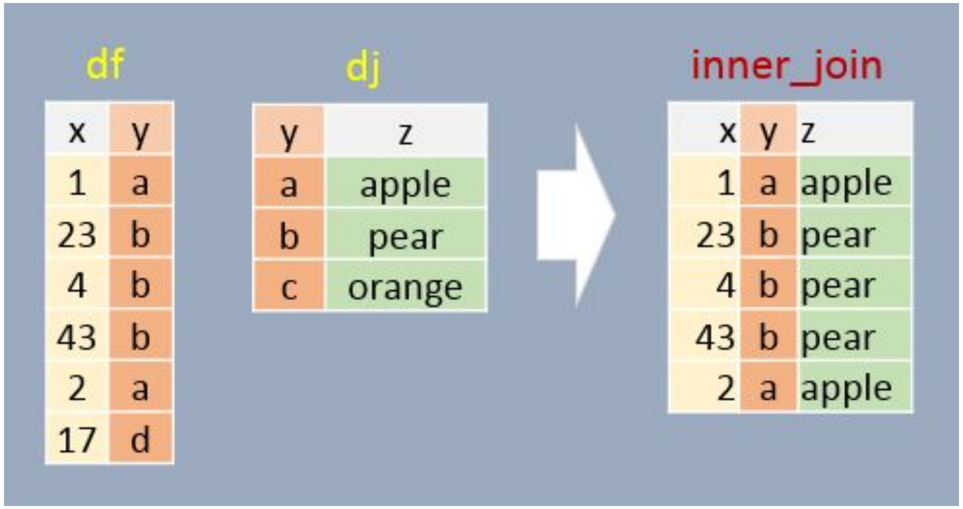
dj

y	z
a	apple
b	pear
c	orange

right_join

x	y	z
1	a	apple
2	a	apple
23	b	pear
4	b	pear
43	b	pear
NA	c	orange

Inner join



df

x	y
1	a
23	b
4	b
43	b
2	a
17	d

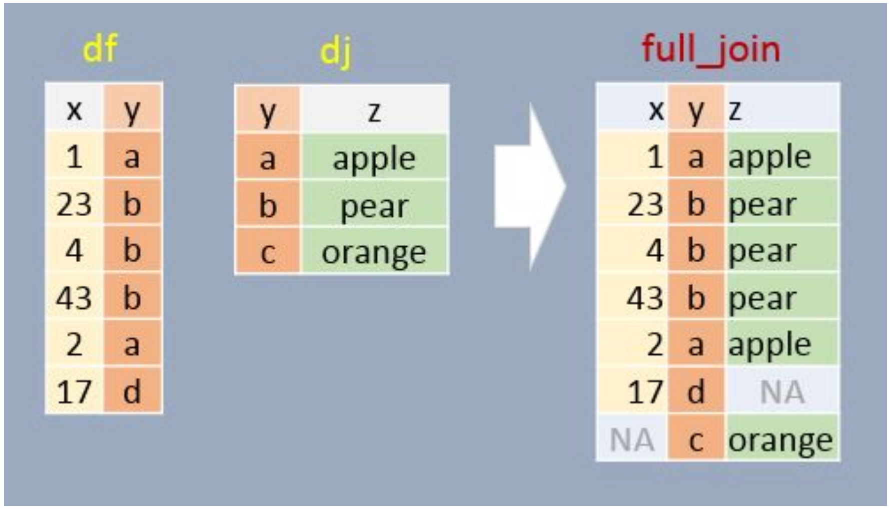
dj

y	z
a	apple
b	pear
c	orange

inner_join

x	y	z
1	a	apple
23	b	pear
4	b	pear
43	b	pear
2	a	apple

Full join



df

x	y
1	a
23	b
4	b
43	b
2	a
17	d

dj

y	z
a	apple
b	pear
c	orange

full_join

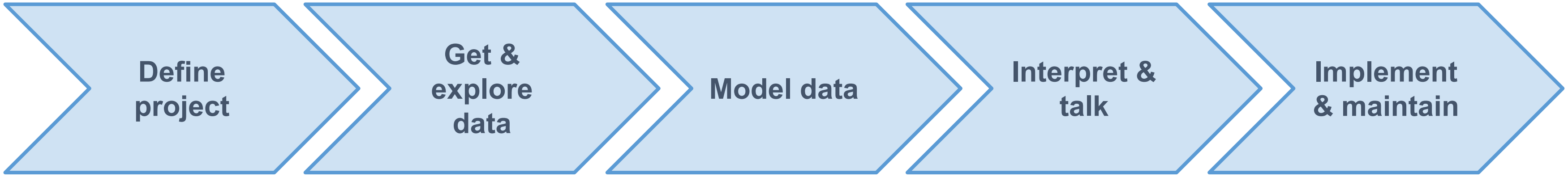
x	y	z
1	a	apple
23	b	pear
4	b	pear
43	b	pear
2	a	apple
17	d	NA
NA	c	orange

Let’s code joins!

Content review Module 4: Fundamentals Data Analysis

- ✅ Joining tables
- Data cleaning
- Code
    - Pandas merge
    - Data cleaning
        - String operations

The Data Science Lifecycle



Define project

- Specify business problem
- Acquire domain knowledge

Get and explore data

- Find appropriate data
- Exploratory Data Analysis
- Clean and pre-process data
- Feature engineering

Model data

- Determine ML task
- Build candidate models
- Select model based on performance metrics

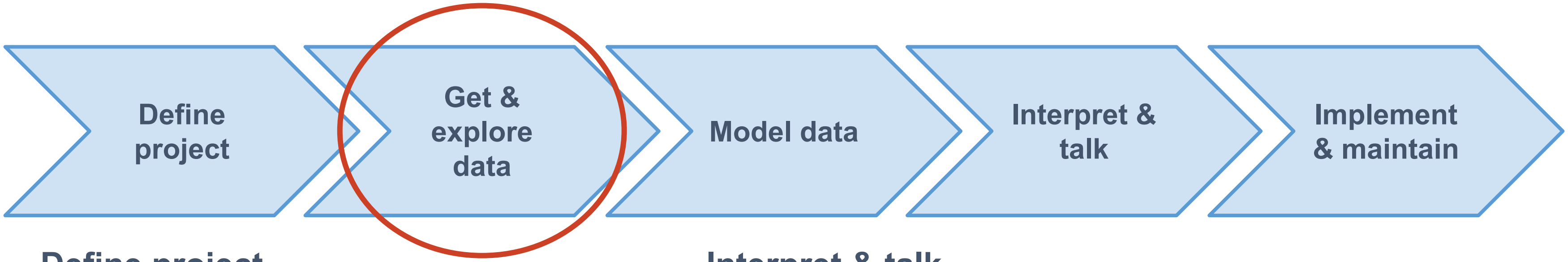
Interpret & talk

- Interpret model
- Communicate model insights

Implement & maintain

- Set up function to predict on new data
- Document process
- Monitor and maintain model

The Data Science Lifecycle



Define project

- Specify business problem
- Acquire domain knowledge

Get and explore data

- Find appropriate data
- Exploratory Data Analysis
- Clean and pre-process data
- Feature engineering

Model data

- Determine ML task
- Build candidate models
- Select model based on performance metrics

Interpret & talk

- Interpret model
- Communicate model insights

Implement & maintain

- Set up function to predict on new data
- Document process
- Monitor and maintain model

Data cleaning

- Identifying and correcting or removing errors, inaccuracies, and incomplete or irrelevant data
- Objective: Improve data quality, making it suitable for modeling

GARBAGE
DATA
GREAT
MODEL
GARBAGE
RESULTS

Data cleaning

- Included but not limited to:
  - Handling missing values: drop them or account for them
  - Handling outliers: drop them or account for them or keep them
  - Remove duplicates
  - Handling incorrect data types
  - Handling inconsistent data (example: age shouldn't be negative)

Content review Module 4: Fundamentals Data Analysis

- ✅ Joining tables
- ✅ Data cleaning
- Code
    - Pandas merge
    - Data cleaning
        - String operations

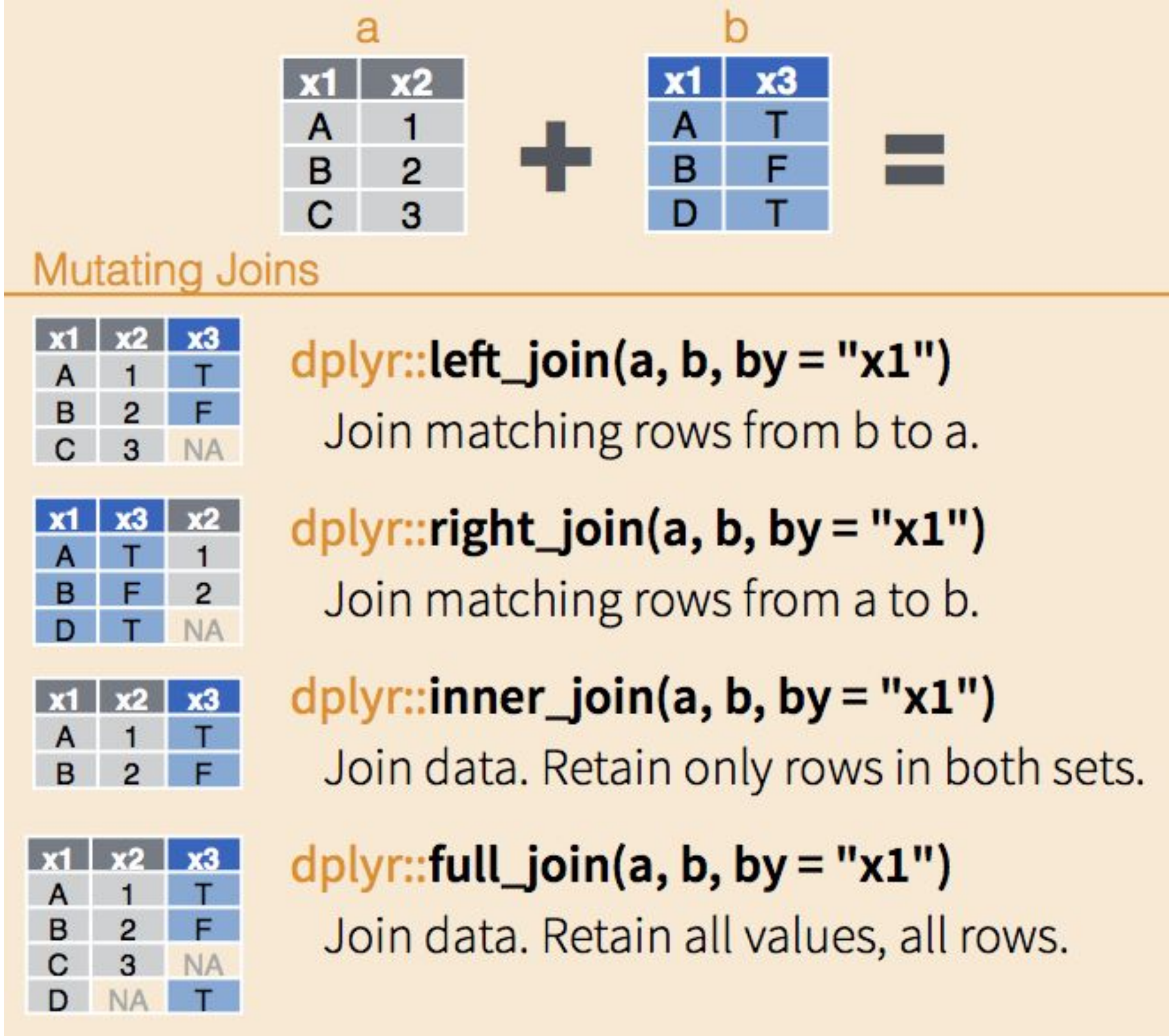
Content review Module 4: Fundamentals Data Analysis

- ✅ Joining tables
- ✅ Data cleaning
- ✅ Code
  - Pandas merge
  - Data cleaning
    - String operations

Let’s code data cleaning!

QUESTIONS?

Types of joins



a

x1	x2
A	1
B	2
C	3

+

b

x1	x3
A	T
B	F
D	T

=

Mutating Joins

x1	x2	x3
A	1	T
B	2	F
C	3	NA

```
dplyr::left_join(a, b, by = "x1")
```

Join matching rows from b to a.

x1	x3	x2
A	T	1
B	F	2
D	T	NA

```
dplyr::right_join(a, b, by = "x1")
```

Join matching rows from a to b.

x1	x2	x3
A	1	T
B	2	F

```
dplyr::inner_join(a, b, by = "x1")
```

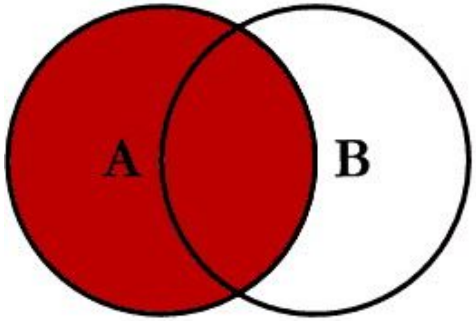
Join data. Retain only rows in both sets.

x1	x2	x3
A	1	T
B	2	F
C	3	NA
D	NA	T

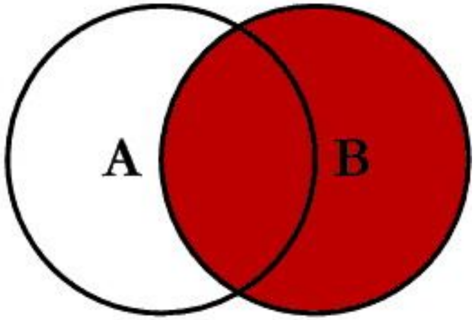
```
dplyr::full_join(a, b, by = "x1")
```

Join data. Retain all values, all rows.

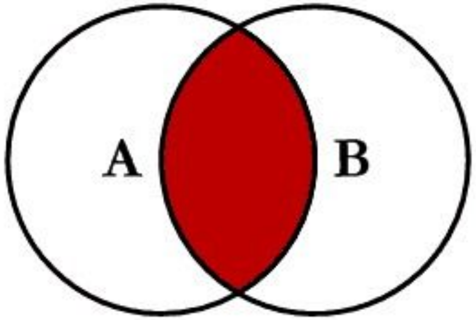
SQL JOINS



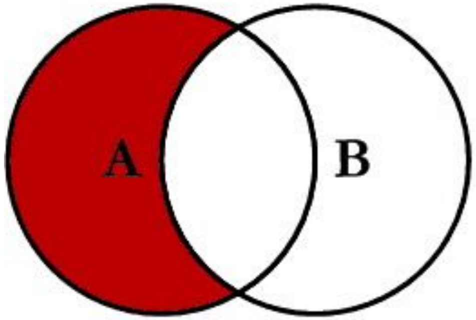
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



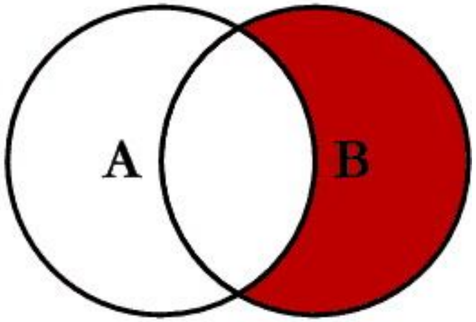
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



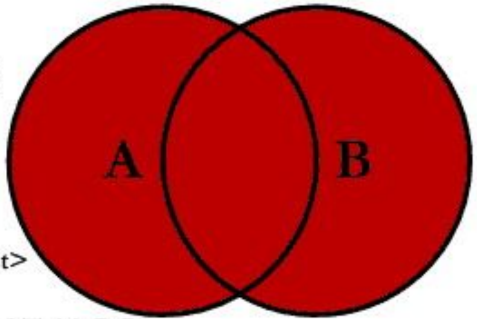
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



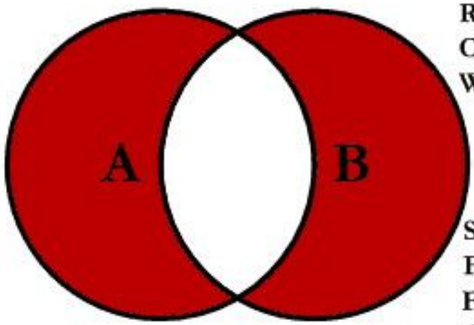
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```