Intro to SQL

SQL

Example DB

Students

ID	Name	Age	Gender	Address
1	Nick D.	20	M	2
2	Andy D.	28	M	2
3	Beth M.	23	F	1
4	Lisa N.	20	F	4

Addresses

ID	Street	Zip	City	State
1	423 Main St.	60647	Chicago	IL
2	13 Main St	60655	Barrington	IL
3	15 Main St	60651	Elsewhere	IL
4	14 Main St	60650	Chicago	IL

All 20 Year Old Students

Students

ID	Name	Age	Gender	Address
1	Nick D.	20	M	2
2	Andy D.	28	M	2
3	Beth M.	23	F	1
4	Lisa N.	20	F	4

20 Year Old Students

ID	Name	Age
1	Nick D.	20
4	Lisa N.	20

```
SELECT ID, Name, Age
FROM Students
WHERE Age = 20;
```

Students

ID	Name	Age	Gender	Address
1	Nick D.	20	M	2
2	Andy D.	28	M	2
3	Beth M.	23	F	1
4	Lisa N.	20	F	4

```
SELECT Students.ID, Name, Street, Zip, City
  FROM Students
JOIN Addresses
  ON Students.Address = Addresses.ID
```

Addresses

ID	Street	Zip	City	State
1	423 Main St.	60647	Chicago	IL
2	13 Main St.	60655	Barrington	IL
3	15 Main St.	60651	Elsewhere	IL
4	14 Main St.	60650	Chicago	IL

Students with Addresses

Student.ID	Name	Street	Zip	City
1	Nick D.	13 Main St.	60655	Barrington
2	Andy D.	13 Main St.	60655	Barrington
3	Beth M.	423 Main St.	60647	Chicago
4	Lisa N.	14 Main St.	60650	Chicago

Students

ID	Name	Age	Gender	Address
1	Nick D.	20	M	2
2	Andy D.	28	M	2
3	Beth M.	23	F	1
4	Lisa N.	20	F	4

```
SELECT Student.ID, Name, Street, Zip, City
  FROM Students
JOIN Addresses
  ON Students.Address = Addresses.ID
WHERE Adresses.City = 'chicago';
```

Addresses

ID	Street	Zip	City	State
1	423 Main St.	60647	Chicago	IL
2	13 Main St.	60655	Barrington	IL
3	15 Main St.	60651	Elsewhere	IL
4	14 Main St.	60650	Chicago	IL

Students with Addresses

Student.ID	Name	Street	Zip	City
3	Beth M.	423 Main St.	60647	Chicago
4	Lisa N.	14 Main St.	60650	Chicago

Some Common SQL Keywords

Keyword	Action
SELECT	Which COLUMNS to include in output table (shrinks the result horizontally!)
FROM	Which TABLE to pull data from
JOIN	Another TABLE to glue / concatenate to the output
ON	What COLUMNS must match when joining two tables
WHERE	Which ROWS to include in the output table (shrinks the result vertically!)

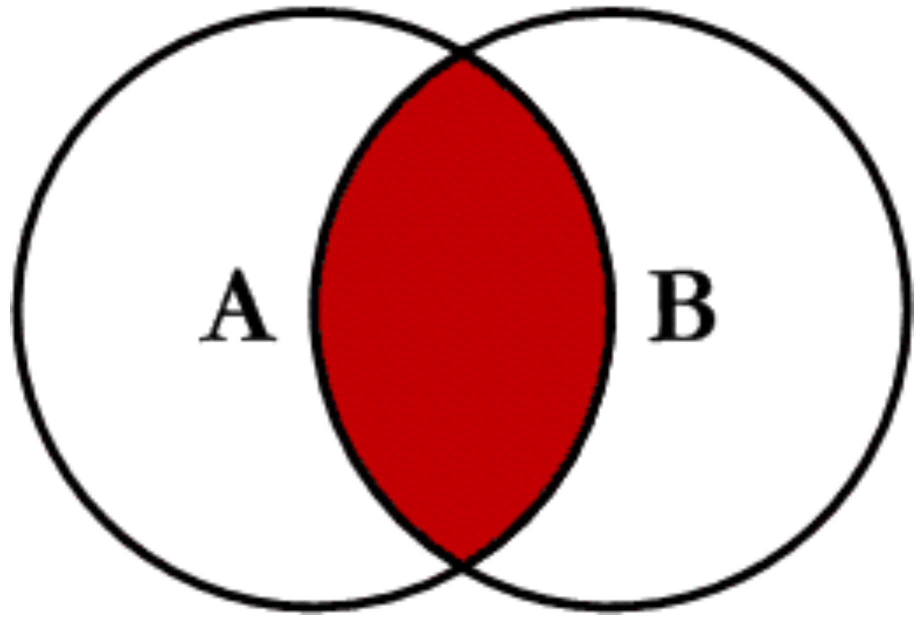
CRUD Operations

SQL is used to create/read/update/delete (CRUD) data from a database

- INSERT: Insert new rows into a table
- SELECT: Get data from a database
- UPDATE: Update existing rows in a table
- DELETE: Delete rows from a table

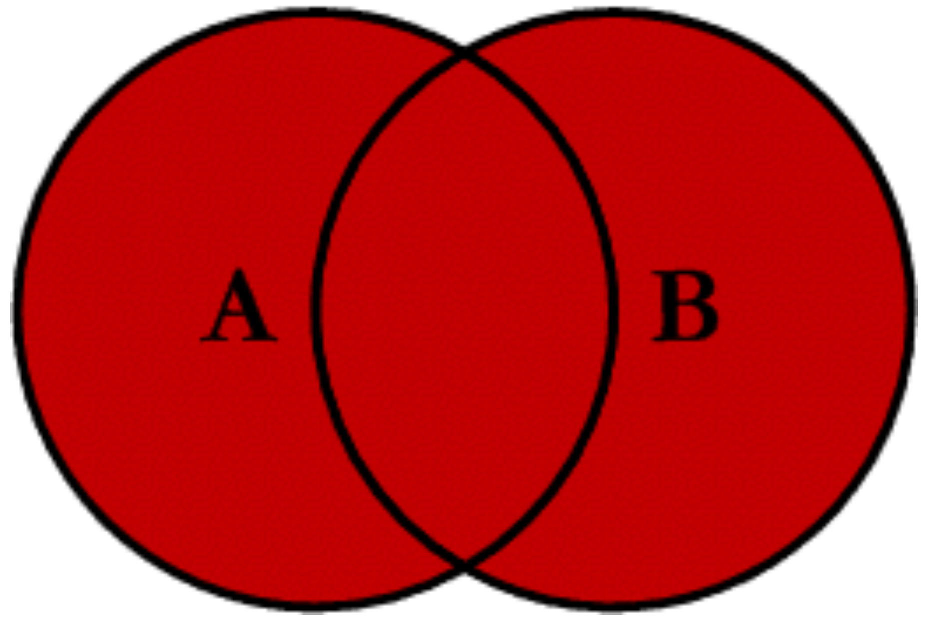
- CREATE / DROP: Make / delete new dbs/tables/views/indexes

Inner Join



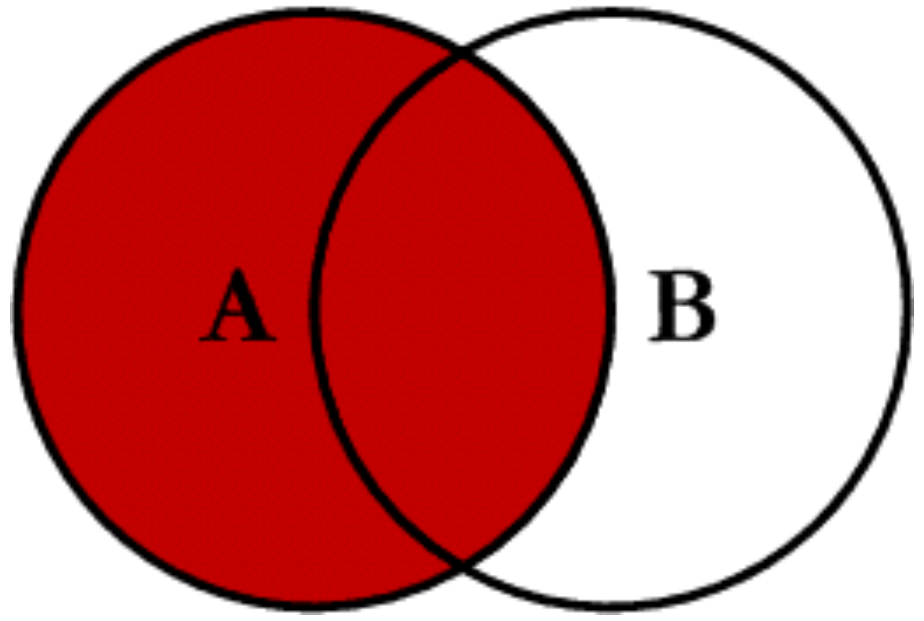
```
SELECT *
FROM A
INNER JOIN B
ON A.Key = B.Key
```

Outer Join



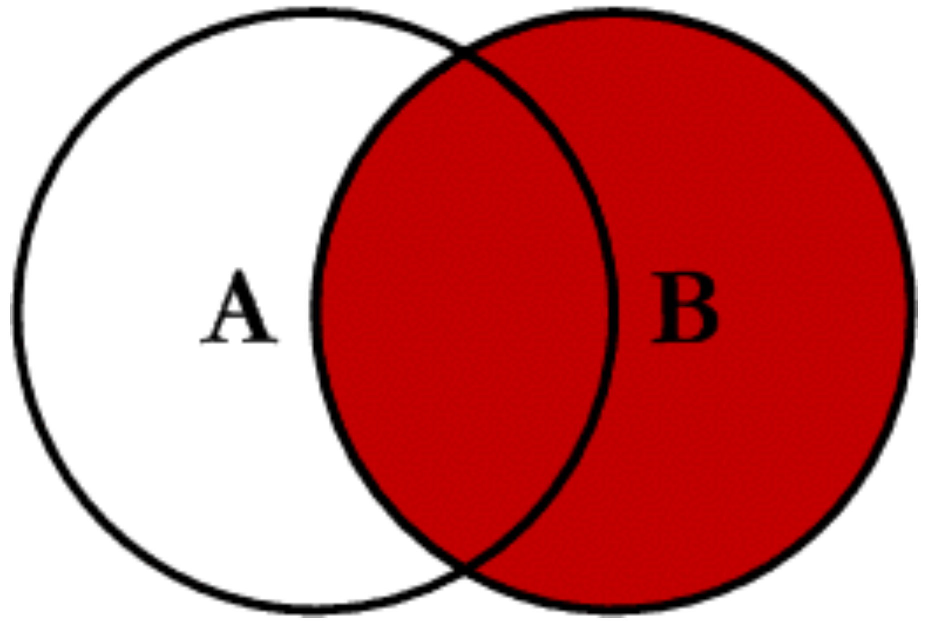
```
SELECT *
FROM A
FULL OUTER JOIN B
ON A.Key = B.Key
```

Left Join



```
SELECT *
FROM A
LEFT JOIN B
ON A.Key = B.Key
```

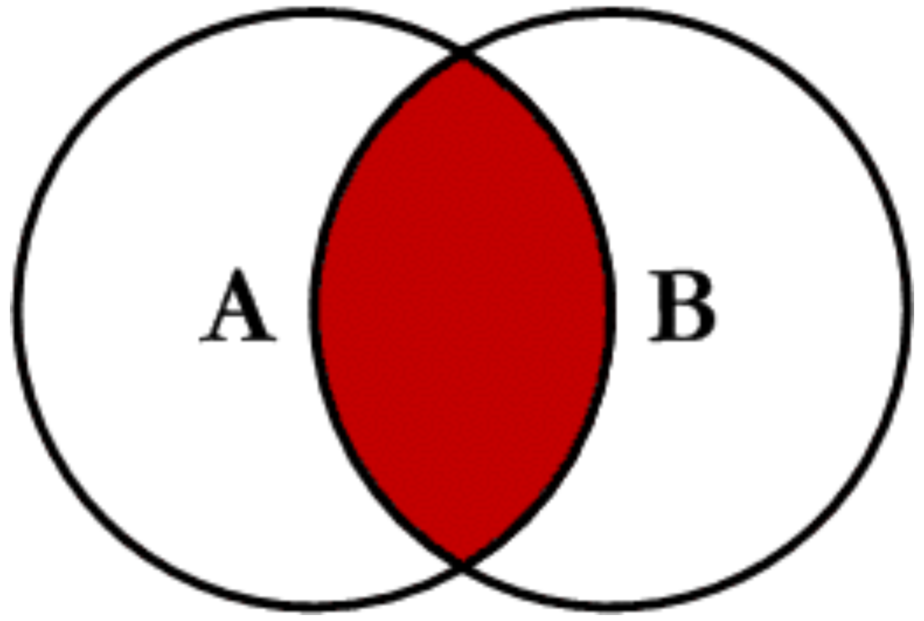
Right Join



```
SELECT *
FROM A
RIGHT JOIN B
ON A.Key = B.Key
```

http://www.codeproject.com/Articles/33052/Visual-Representation-of-SQL-Joins

Inner Join



```
SELECT pets.name, owners.name
FROM owners
INNER JOIN pets
ON pets.ownerID = owners.ID
```

OWNERS

ID	name
1	Geordi
2	Janeway
3	Data
4	Spock

PETS

ID	ownerID	type	name
1	4	Monkey	Mittens
2	null	Lizard	Carol
3	1	Dog	Rufus
4	2	Cat	Fireball

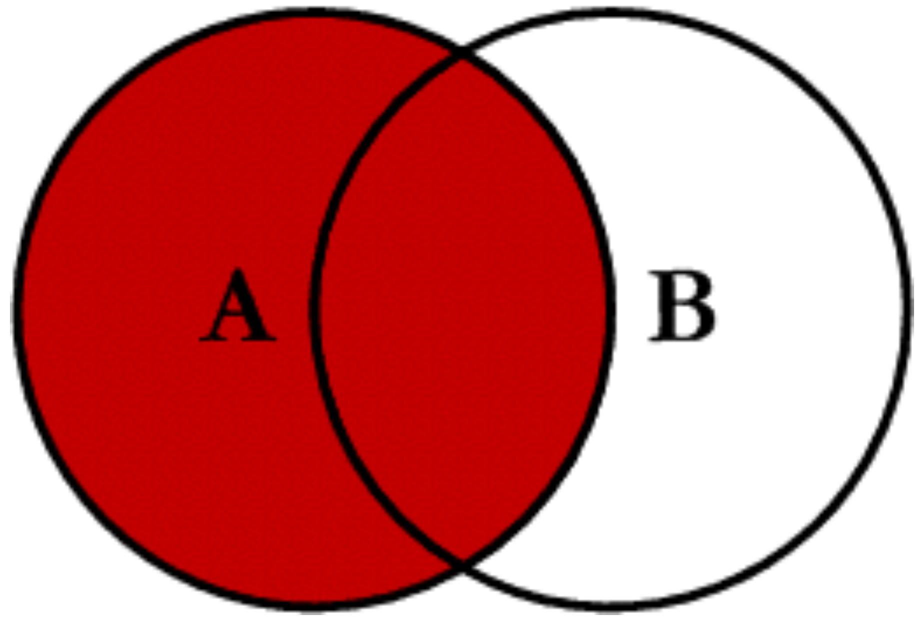
pets.name	owners.name
Mittens	Spock
Rufus	Geordi
Fireball	Janeway

PETS

ID	ownerID	type	name
1	4	Monkey	Mittens
2	null	Lizard	Carol
3	1	Dog	Rufus
4	2	Cat	Fireball

pets.name	owners.name
Mittens	Spock
Rufus	Geordi
Fireball	Janeway
null	Data

Left Join



```
SELECT pets.name, owners.name
FROM owners
LEFT JOIN pets
ON pets.ownerID = owners.ID
```

OWNERS

ID	name
1	Geordi
2	Janeway
3	Data
4	Spock

pets.name	owners.name
Mittens	Spock
Carol	null
Rufus	Geordi
Fireball	Janeway

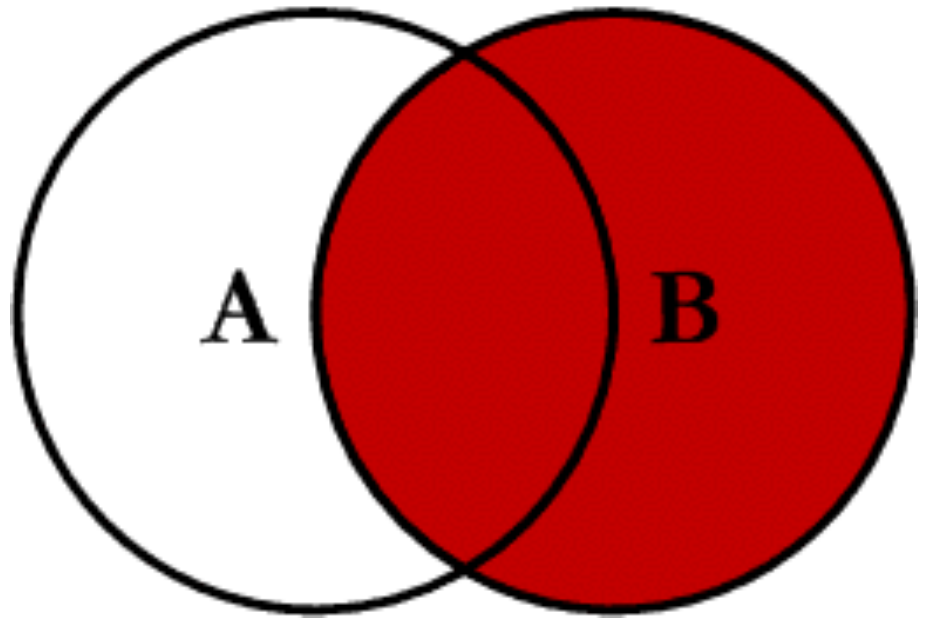
PETS

ID	ownerID	type	name
1	4	Monkey	Mittens
2	null	Lizard	Carol
3	1	Dog	Rufus
4	2	Cat	Fireball

OWNERS

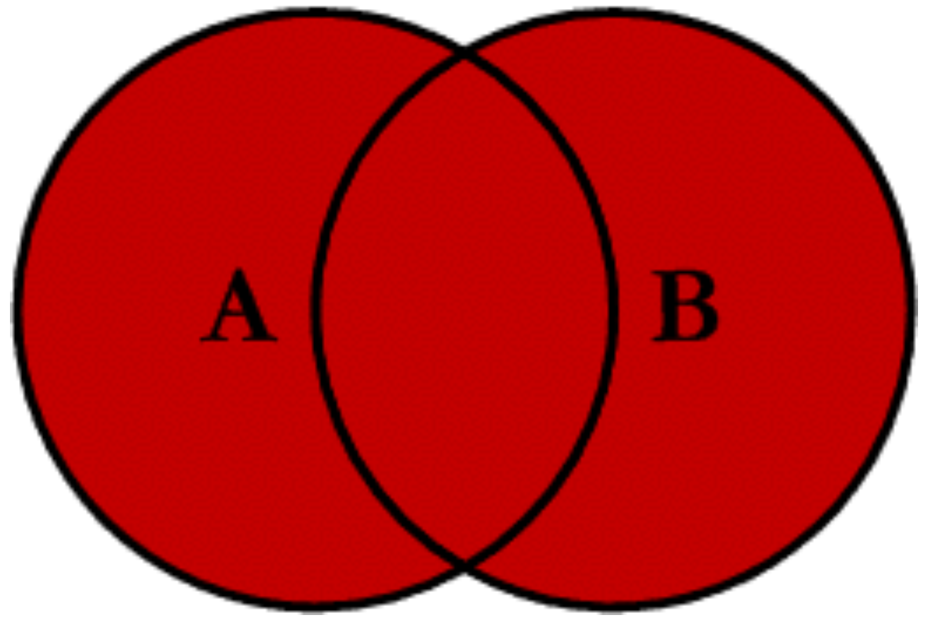
ID	name
1	Geordi
2	Janeway
3	Data
4	Spock

Right Join



```
SELECT pets.name, owners.name
FROM owners
RIGHT JOIN pets
ON pets.ownerID = owners.ID
```

Outer Join



```
SELECT pets.name, owners.name
FROM owners
FULL OUTER JOIN pets
ON pets.ownerID = owners.ID
```

OWNERS

ID	name
1	Geordi
2	Janeway
3	Data
4	Spock

PETS

ID	ownerID	type	name
1	4	Monkey	Mittens
2	null	Lizard	Carol
3	1	Dog	Rufus
4	2	Cat	Fireball

pets.name	owners.name
Mittens	Spock
Carol	null
Rufus	Geordi
Fireball	Janeway
null	Data

AS

ID	Name	Age
1	Bart S.	10
2	Lisa S.	8
3	Jim F.	13
4	Joan B.	15

StudentID	SchoolID
1	1
2	1
3	2
4	3

ID	Name	Level
1	Springfield Elementary	E
2	Brook Middle	M
3	Springbrook High	H
4	Springfield University	U

```
SELECT *
  FROM Student AS st
  INNER JOIN Enrollment AS e
    ON st.ID = e.StudentID
  INNER JOIN School as sc
    ON e.SchoolID = sc.ID;
```

st.ID	st.Name	Age	StudentID	SchoolID	sc.ID	sc.Name	Level
1	Bart S.	10	1	1	1	Springfield Elementary	E
2	Lisa S.	8	2	1	1	Springfield Elementary	E
3	Jim F.	13	3	2	2	Brook Middle	M
4	Joan B.	15	4	3	3	Springbrook High	H

AS

(without AS)

ID	Name	Age
1	Bart S.	10
2	Lisa S.	8
3	Jim F.	13
4	Joan B.	15

StudentID	SchoolID
1	1
2	1
3	2
4	3

ID	Name	Level
1	Springfield Elementary	E
2	Brook Middle	M
3	Springbrook High	H
4	Springfield University	U

```
SELECT *
  FROM Student st
  INNER JOIN Enrollment e
    ON st.ID = e.StudentID
  INNER JOIN School sc
    ON e.SchoolID = sc.ID;
```

st.ID	st.Name	Age	StudentID	SchoolID	sc.ID	sc.Name	Level
1	Bart S.	10	1	1	1	Springfield Elementary	E
2	Lisa S.	8	2	1	1	Springfield Elementary	E
3	Jim F.	13	3	2	2	Brook Middle	M
4	Joan B.	15	4	3	3	Springbrook High	H

GROUP BY
+
COUNT

ID	Name	Age
1	Bart S.	10
2	Lisa S.	8
3	Jim F.	13
4	Joan B.	15

StudentID	SchoolID
1	1
2	1
3	2
4	3

ID	Name	Level
1	Springfield Elementary	E
2	Brook Middle	M
3	Springbrook High	H
4	Springfield University	U

```
SELECT Name, COUNT(*)
  FROM School
  INNER JOIN Enrollment
    ON School.ID = Enrollment.StudentID
  GROUP BY Name;
```

Name	COUNT(*)
Springfield Elementary	2
Brook Middle	1
Springbrook High	1
Springfield University	0

ORDER BY

```
SELECT *
  FROM Student
  ORDER BY Age DESC;
```

ID	Name	Age
1	Bart S.	10
2	Lisa S.	8
3	Jim F.	13
4	Joan B.	15

StudentID	SchoolID
1	1
2	1
3	2
4	3

ID	Name	Level
1	Springfield Elementary	E
2	Brook Middle	M
3	Springbrook High	H
4	Springfield University	U

ID	Name	Age
4	Joan B.	15
3	Jim F.	13
1	Bart S.	10
2	Lisa S.	8

SUB-QUERIES

ID	Name	Age
1	Bart S.	10
2	Lisa S.	8
3	Jim F.	13
4	Joan B.	15

StudentID	SchoolID
1	1
2	1
3	2
4	3

ID	Name	Level
1	Springfield Elementary	E
2	Brook Middle	M
3	Springbrook High	H
4	Springfield University	U

```sql
SELECT ID, Name, Age
  FROM Student
  INNER JOIN Enrollment
    ON Student.ID = Enrollment.StudentID
  INNER JOIN (
    SELECT SchoolID
      FROM Student
      WHERE Student.Name = 'Lisa S.'
      INNER JOIN Enrollment
        ON Student.ID = Enrollment.StudentID
  ) AS LisaSchools
    ON LisaSchools.SchoolID = Enrollment.SchoolID
  WHERE Name != 'Lisa S.';
```

ID	Name	Age
1	Bart S.	10

WORKSHOP