PostgreSQL: Getting Started

Installation and Configuration

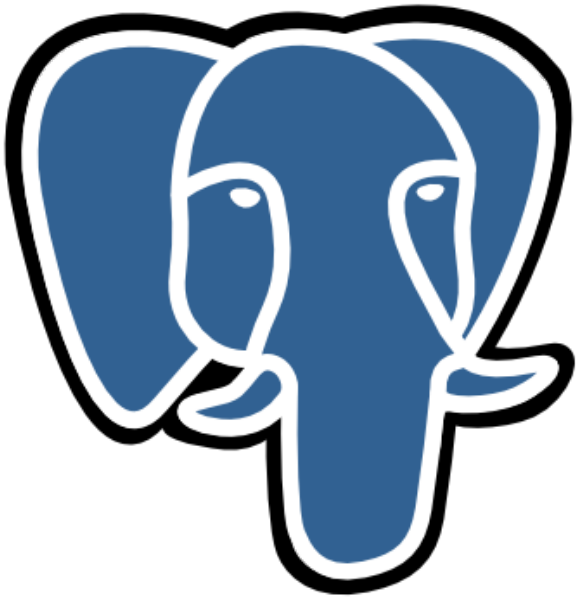
Pinal Dave
http://blog.sqlauthority.com
@pinaldave

pluralsight
hardcore developer training

William Shakespeare

“**What's in a name?** that which we call a rose By any other name would smell as sweet.”
~ Romeo and Juliet

PostgreSQL
or
Postgres



PostgreSQL

A Brief History of the Name

- Evolved from Ingres Project at University of California, Berkeley
- Ingres Team Leader – Michael Stonebraker
- 1982 – Michael left University
- 1985 – Michael returned to University
- Started to work on Post-Ingres project
- 1988 – The first prototype of product
- 1994 – Ingres based Quel Query Language interpreter was replaced with SQL Query language interpreter
- 1996 – PostgreSQL
- 1997 – First PostgreSQL release

What is PostgreSQL?

- **Object Relational Database Management System (ORDBMS)**
- **Free**
- **Open Source**
- **Cross Platform**
  - Linux
  - FreeBSD
  - Solaris
  - Microsoft Windows
  - Mac OS X (starting with OS X 10.7 Lion)

PostgreSQL Important Features

- **SQL: 2011 Standard**
- **ACID compliant**
- **Indexes**
- **Views, Triggers, Procedures, Functions**
- **Relationships**
- **MultiVersion Concurrency Control (MVCC)**
- **SQL:2008 Datatypes**
  - INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, and TIMESTAMP
- **Native Programming Interface**
  - C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC

PostgreSQL Limits

Limit	Value
Maximum Database Size	Unlimited
Maximum Table Size	32 TB
Maximum Row Size	1.6 TB
Maximum Field Size	1 GB
Maximum Rows per Table	Unlimited
Maximum Columns per Table	250 - 1600 depending on column types
Maximum Indexes per Table	Unlimited

Prominent Users of PostgreSQL

- Yahoo
- Skype
- Instagram
- Disqus
- OpenStreetMap
- Reddit

What is This Course About?

- Installation and Configuration
- Creating and Accessing Database and Table
- Updating and Deleting Data from Table
- Retrieving Data from Multiple Tables
- Resources

Series of 5 Courses

- Course 1: PostgreSQL: Getting Started
- Course 2: PostgreSQL: Introduction to SQL Queries
- Course 3: PostgreSQL: Advanced SQL Queries
- Course 4: PostgreSQL: Advanced Server Programming
- Course 5: PostgreSQL: Index Tuning and Performance Optimization

Important Downloads

- **PostgreSQL Download**
  - http://www.postgresql.org/download/
- **Windows Graphical Installer**
  - PostgreSQL Server
  - pgAdmin III – a graphical IDE
  - http://www.postgresql.org/download/windows/
- **Samples Database - *pagilia***
  - http://www.postgresqltutorial.com/postgresql-sample-database/
  - http://bit.ly/pagilia
- **Postgres for .NET Developers by Rob Conery**

Installation and Configuration

Summary

- **PostgreSQL is Free and Open Source Object Relational Database Management System (ORDBMS)**
- **Cross Platform, ACID compliant**
- **Native Language Support**
- **Windows Graphical Installer**
  - PostgreSQL Server
  - pgAdmin III – a graphical IDE

Let's START !

Pluralsight Discussion Forum

http://twitter.com/pinaldave

http://facebook.com/SQLAuth

PostgreSQL: Creating and Accessing Database Tables

Pinal Dave
http://blog.sqlauthority.com
@pinaldave

pluralsight
hardcore developer training

In Last Module

- Brief history of PostgreSQL
- How to Install and Configure PostgreSQL
- Wrote very first SELECT statement

In This Module

- How to create a database
- How to create a database table
- Populating data in a table
- Various options to restore a database

Scenario Setup

- Two Database Administrators
- Rahul – Sr. Database Administrator
- Mike – Jr. Database Administrator

Icon Courtesy: http://www.icons-land.com/

Scenario 1

Task:

- Rahul asks Mike to restore database to the server from compressed backup file.

Scenario 2

Task:

- Rahul asks Mike to restore sample database to the server over the old database and ask for only schema (no data).

Scenario 3

Task:

- Rahul asks Mike to restore a single **table** schema and data (along with index) from one database to another database.

Scenario 4

Task:

- Rahul asks Mike to prepare an HTML report for all the dependencies and dependents.

Summary

- pgAdmin III – a graphical IDE provides effective ways to create new objects with PostgreSQL
- COPY command can help generate CSV file from database tables
- You can generate Schema or Data of the table from pgAdmin III

PostgreSQL: Data Operations
SELECT, UPDATE, DELETE

Pinal Dave
http://blog.sqlauthority.com
@pinaldave

pluralsight
hardcore developer training

In Last Module

- Restoring sample database
- Create a database table schema with pgAdmin III
- Export data from database tables

In This Module

- Basic Structure of SELECT statement
- Updating/Inserting Database Tables
- Deleting Database Tables

Scenario Setup

- Two Database Administrators
- Rahul – Sr. Database Administrator
- Mike – Jr. Database Administrator

Icon Courtesy: http://www.icons-land.com/

Scenario 1

Task:

- Rahul asks Mike to retrieve following various data from actor table of DVDRental Database

Subtask 1: Retrieve all the actor with actor_id lesser than 11

Subtask 2: Order data by first_name

Subtask 3: Order data by last_name in decending order

Scenario 2

Task:

- Rahul asks Mike to list unique first name along with count of occurrence from actor table of DVDRental Database

Subtask 1: Count first name of actor where actor_id is between 100 and 200

Subtask 2: List all the first name which occurs more than 1 time

Scenario 3

Task:

- Rahul asks Mike to update first name of an actor and insert new data into table

Subtask 1: Change name of the actor from *Cuba* to *Jacob*

Subtask 2: Insert a new row into table with actor name *Robert Johnson*

Scenario 4

Task:

- Rahul asks Mike to delete rows with specific condition from table actor

Subtask 1: Delete all rows where first name of actor is *Robert*

Summary

- **Basic structure of SELECT statement**
  - SELECT columnname
  - FROM tablename
  - WHERE condition
  - GROUP BY grouping-columns
  - HAVING grouping-condition
  - ORDER BY columnname

- **We can use pgAdmin III to generate various SELECT, INSERT, UPDATE and DELETE script**

PostgreSQL: Database Joins
Retrieving Data from Multiple Tables

Pinal Dave
http://blog.sqlauthority.com
@pinaldave

pluralsight
hardcore developer training

In Last Module

- Basics of SELECT statement
- Updating data into a table
- Inserting data into a table
- Deleting data from a table

In This Module

- **Retrieving data from more multiple tables**
- **Basics of Join**
  - Inner Join
  - Outer Join
    - Left Outer Join
    - Right Outer Join
    - Full Outer Join
  - Cross Join

Scenario Setup

- Two Database Administrators
- Rahul – Sr. Database Administrator
- Mike – Jr. Database Administrator

Icon Courtesy: http://www.icons-land.com/

Scenario Setup

- Two Database Administrators and a Teacher
- Rahul – Sr. Database Administrator
- Mike – Jr. Database Administrator
- Troy – School Teacher

Icon Courtesy: http://www.icons-land.com/

Scenario Setup

- **We have three tables –**
  - Students
  - Classes
  - StudentClass
- **The student can sign up maximum of three classes**
- **In summer student can opt out and can sign up for no classes**

Icon Courtesy: http://www.icons-land.com/

Scenario 1

Task:

- Troy wants to retrieve all the students who have signed up for classes in the summer.

- **Rahul's hint to Mike:**
  - Learn Inner Join

Inner Join

- INNER join returns rows when there is at least one match in both the tables
- Avoid ambiguity by qualifying each column name with table name
- Join tables based on relationships as well ad-hoc
- Operators for Join
  - =
  - >
  - <
  - <=
  - >=

Inner Join

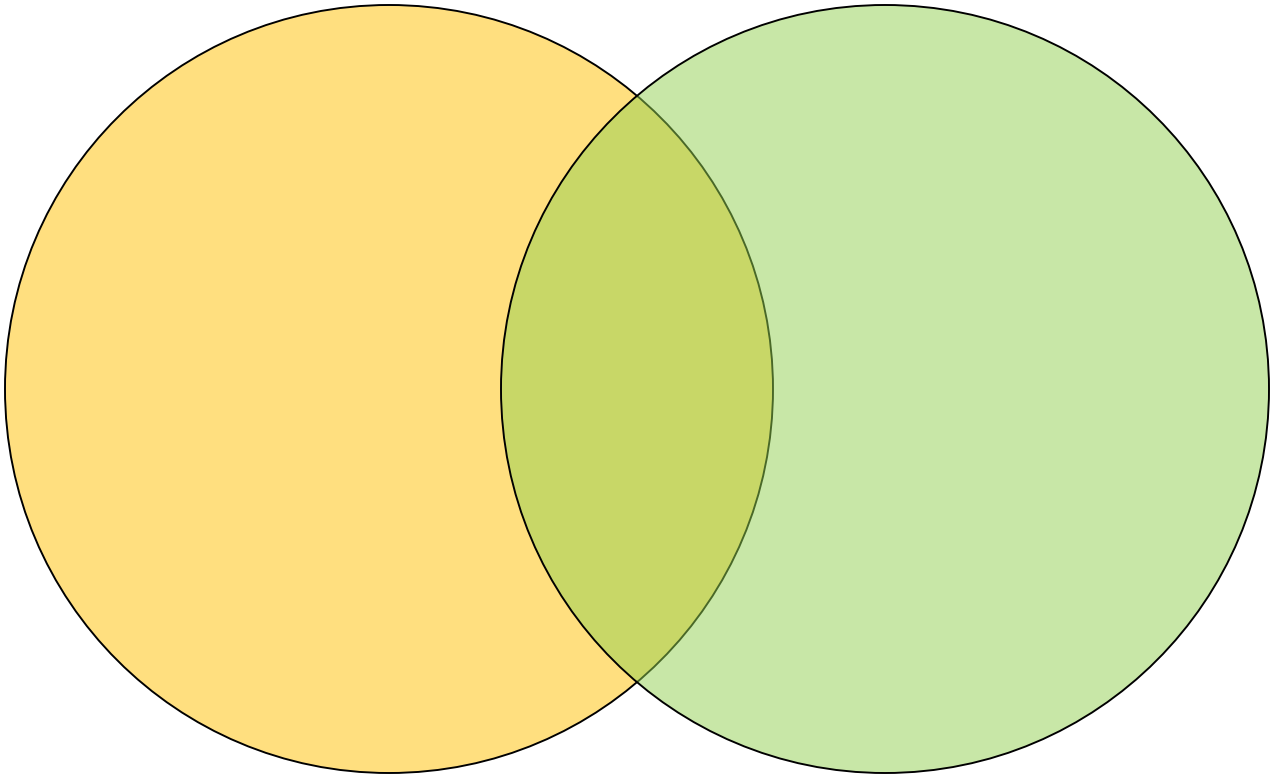


Table 1
Table 2

Inner Join

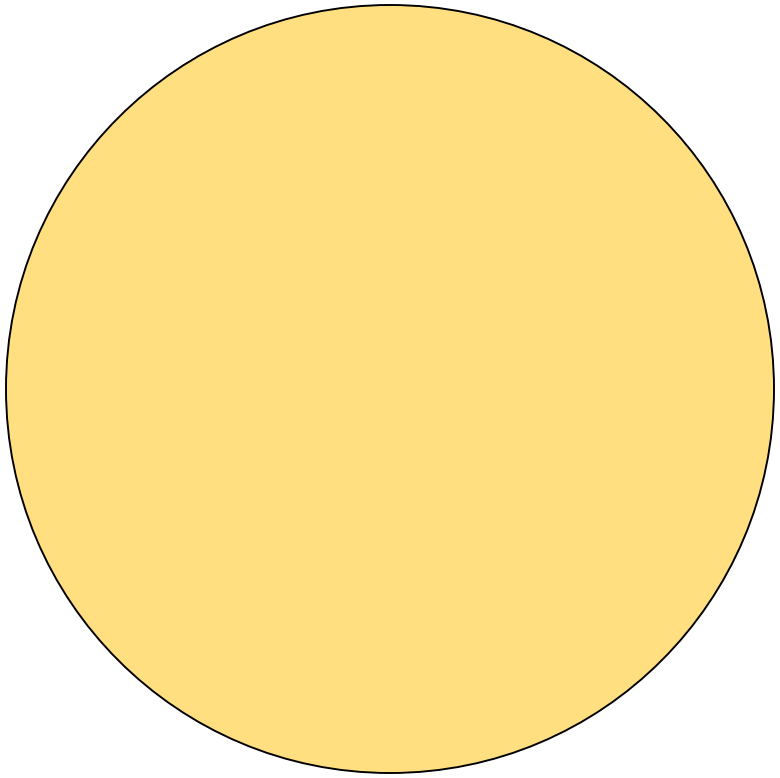


Table 1

Inner Join

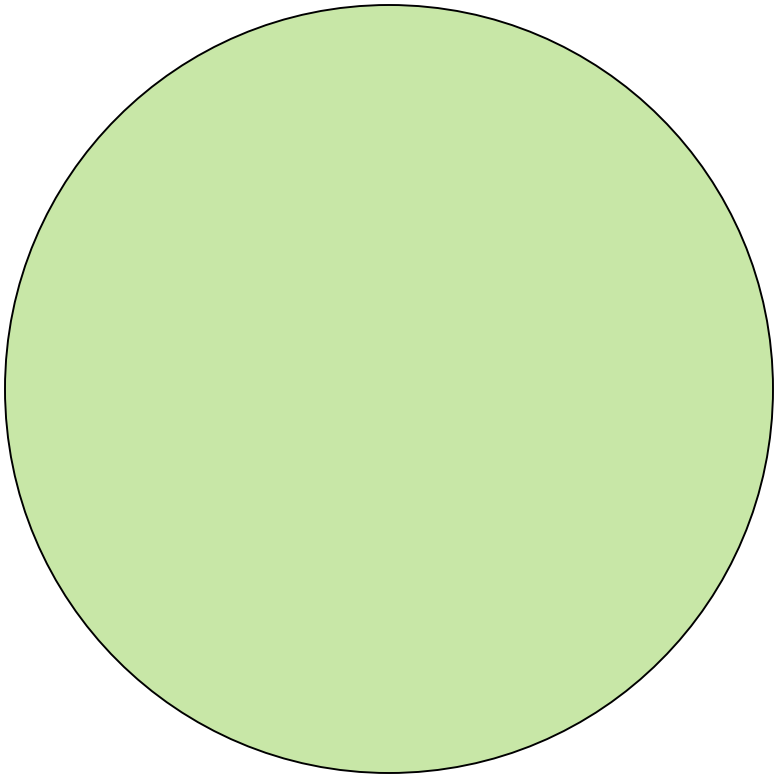


Table 2

Inner Join

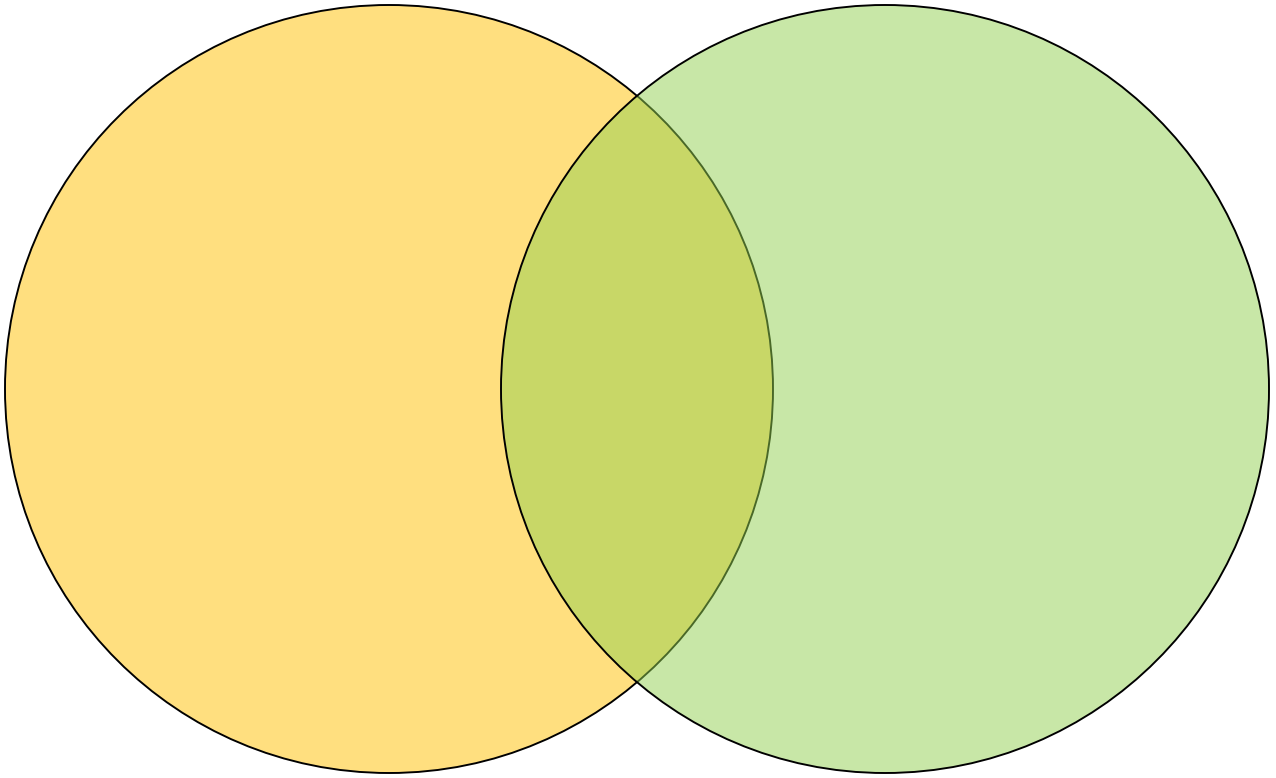


Table 1 Table 2

Inner Join

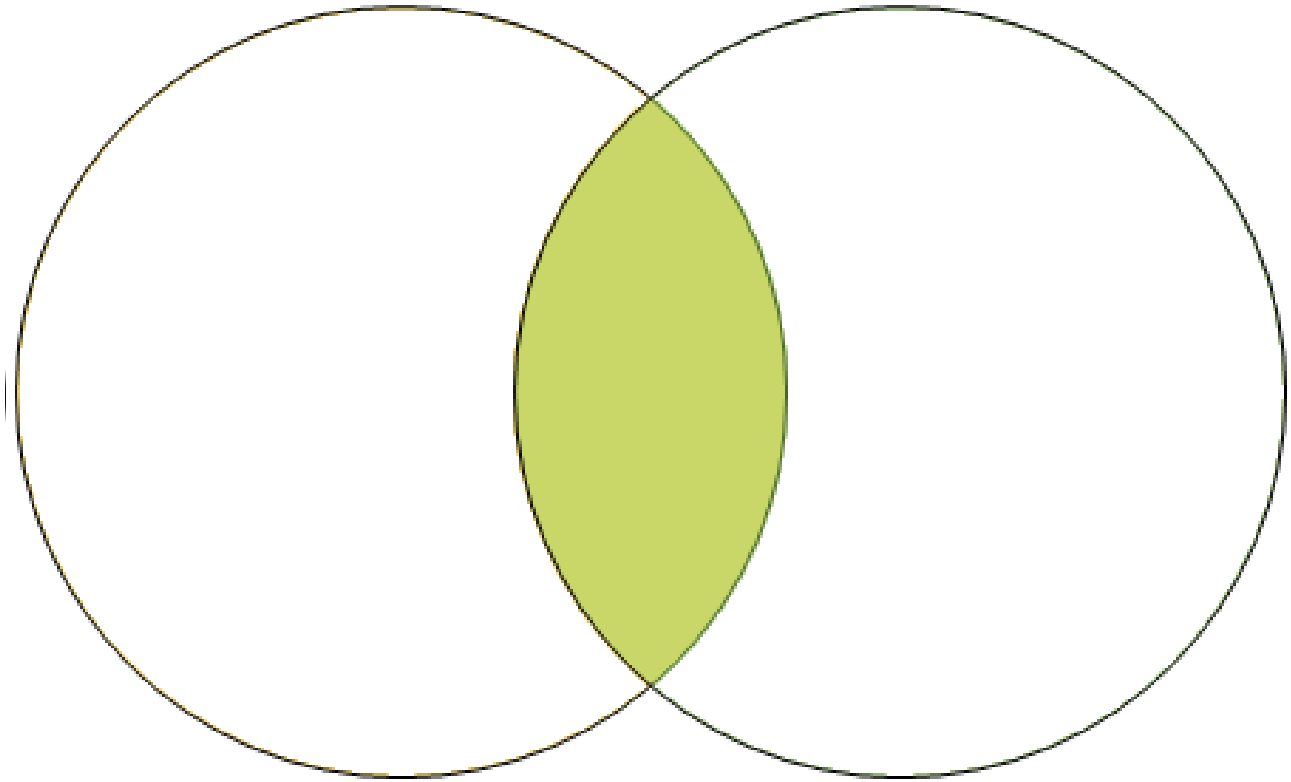


Table 1 Table 2

Scenario 1

Task:

- Troy wants to retrieve all the students who have signed up for classes in the summer.

- **Rahul's hint to Mike:**
  - Learn Inner Join

Scenario 2

Task:

- Troy wants to retrieve all the students who have signed up for no classes in the summer.

- **Rahul's hint to Mike:**
  - Learn Left Outer Join

Left Outer Join

- LEFT OUTER join returns all the rows from the left table with the matching rows from the right table
- If there are no columns matching in the right table, it returns NULL values

Left Outer Join

Left Outer Join

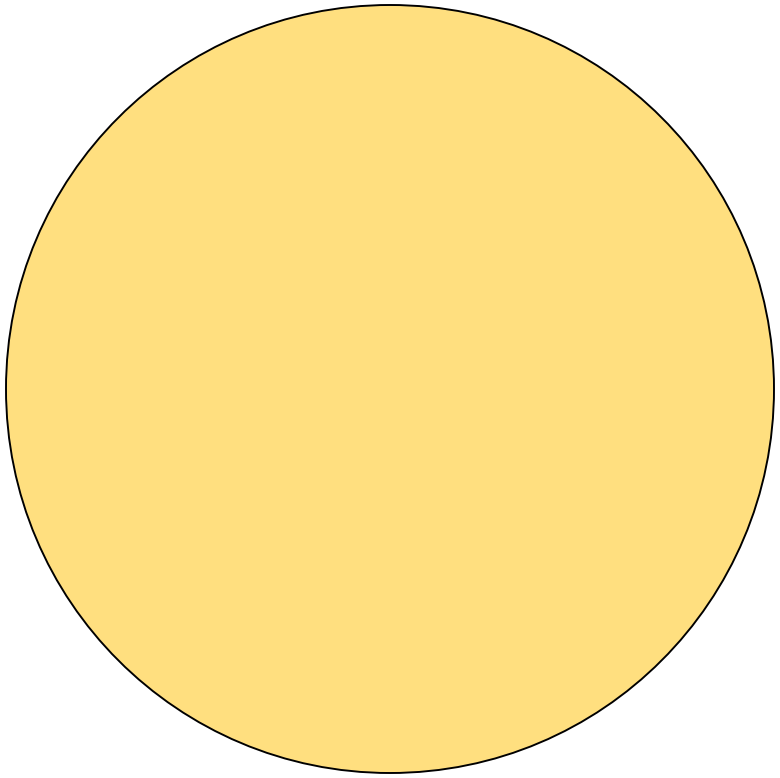


Table 1

Left Outer Join

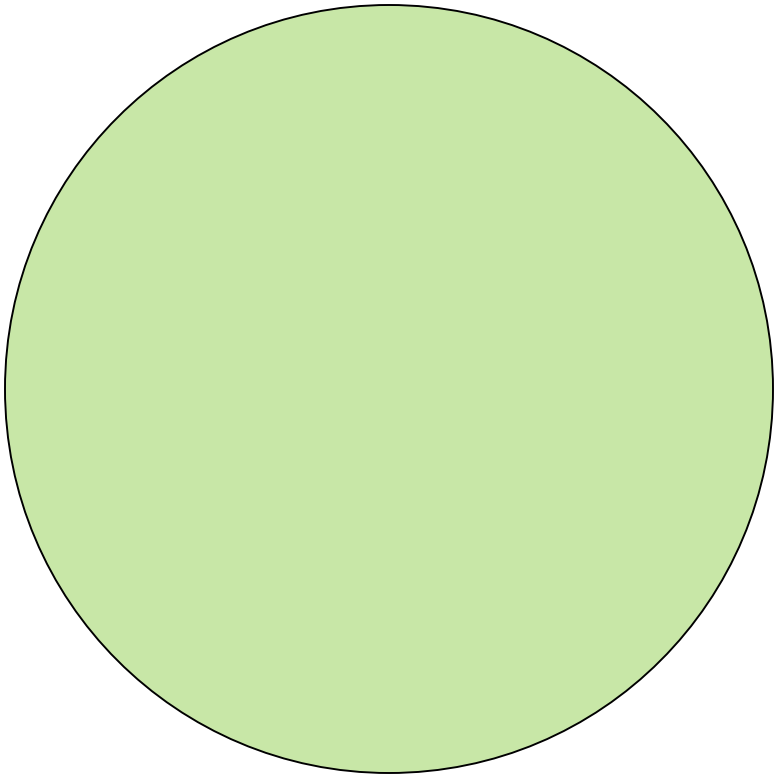


Table 2

Left Outer Join

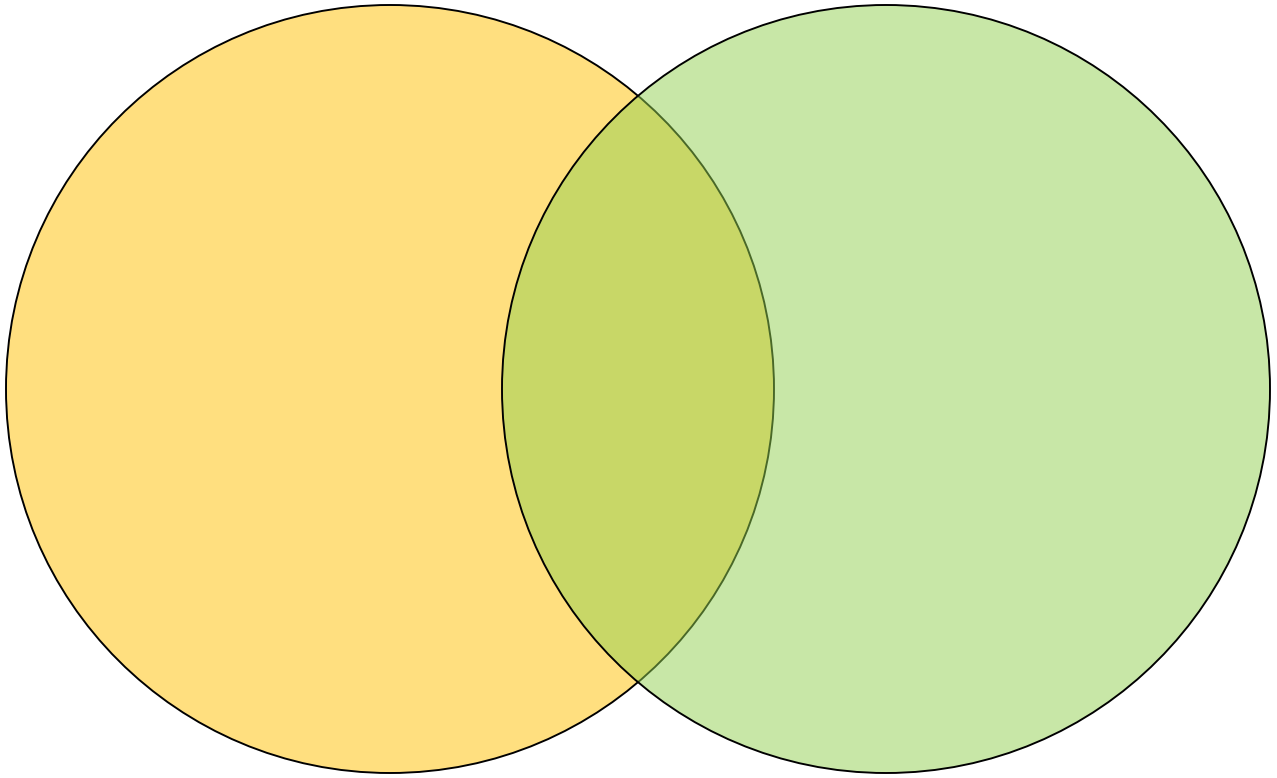
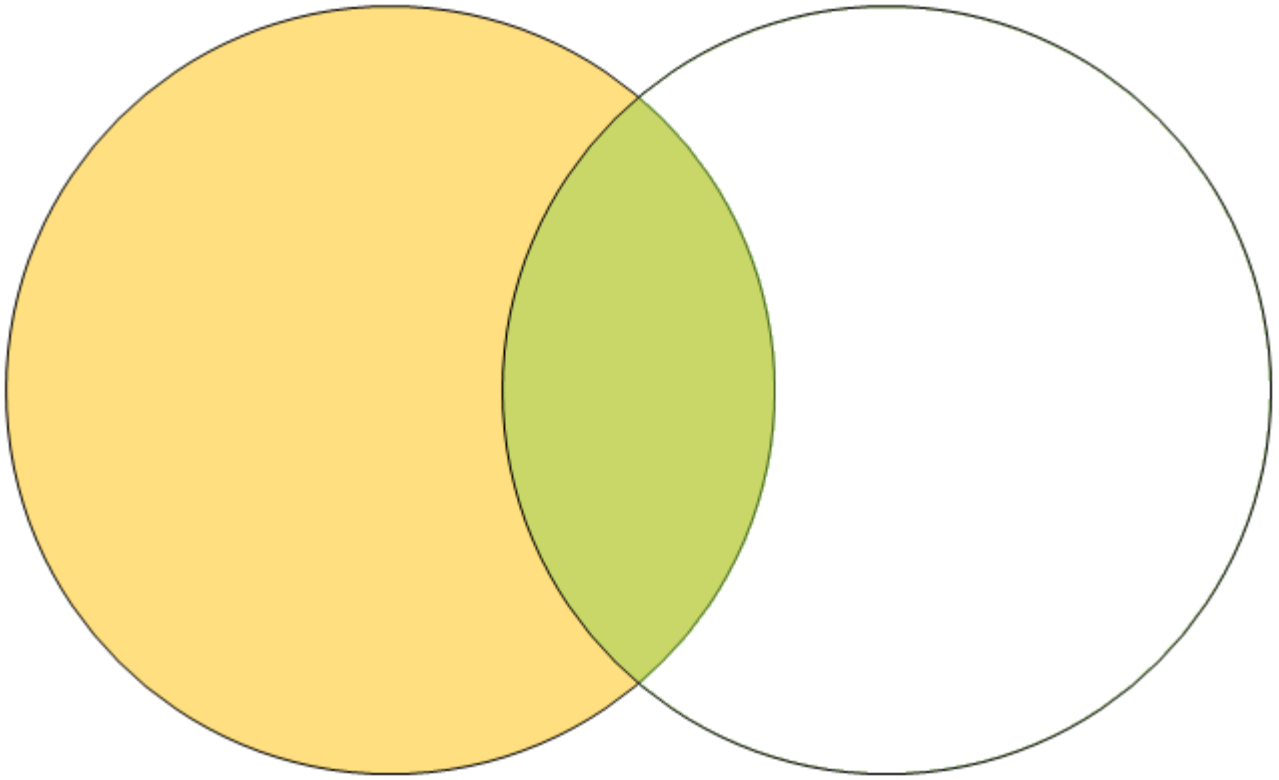


Table 1 Table 2

Left Outer Join



Scenario 2

Task:

- Troy wants to retrieve all the students who have signed up for no classes in the summer.

- **Rahul's hint to Mike:**
  - Learn Left Outer Join

Scenario 3

Task:

- Troy wants to retrieve all the classes not signed up by any student in the summer.

- **Rahul's hint to Mike:**
  - Learn Right Outer Join

Right Outer Join

- RIGHT OUTER join returns all the rows from the right table with the matching rows from the left table
- If there are no columns matching in the left table, it returns NULL values

Right Outer Join

Right Outer Join

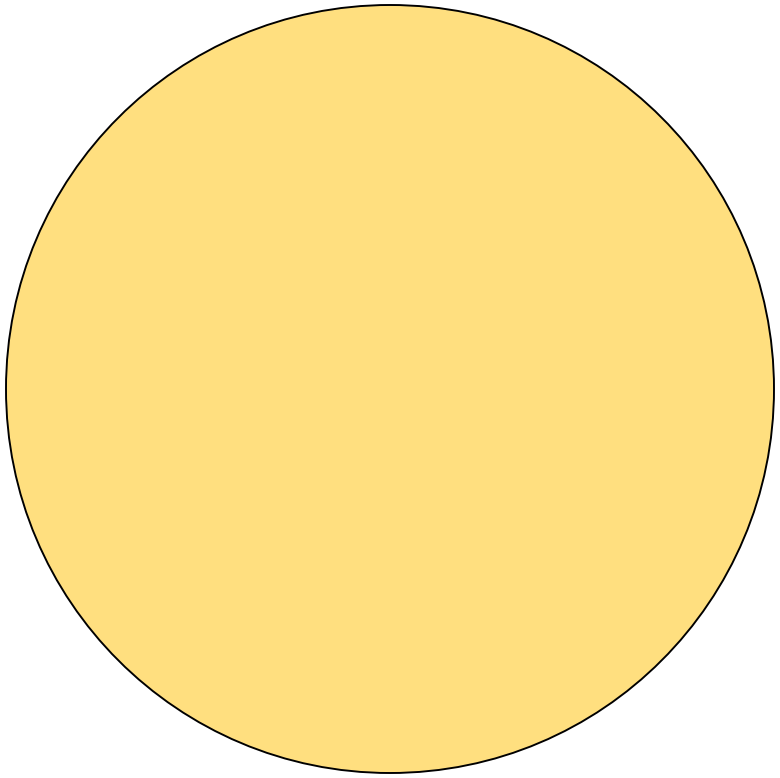


Table 1

Right Outer Join

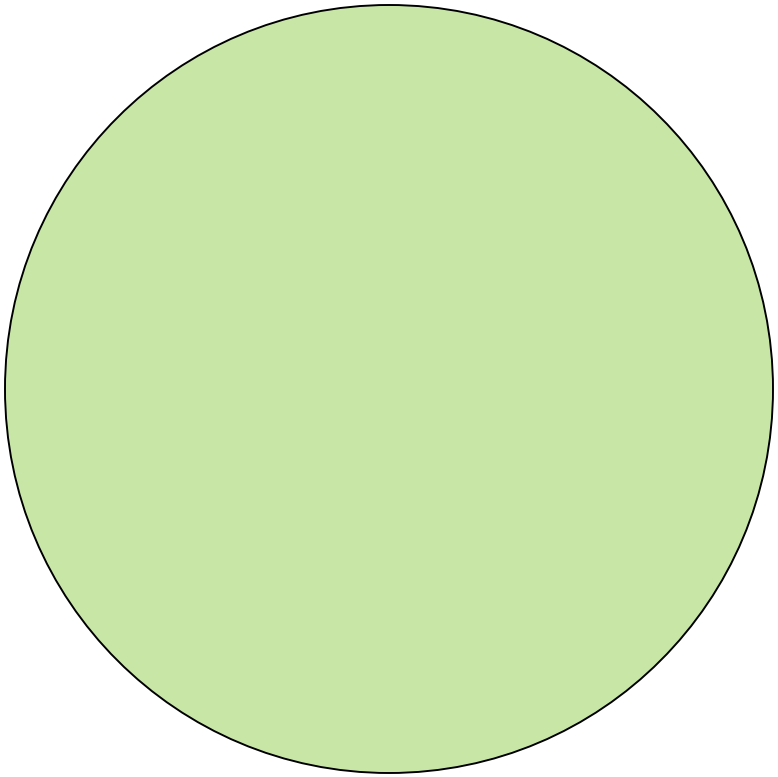


Table 2

Right Outer Join

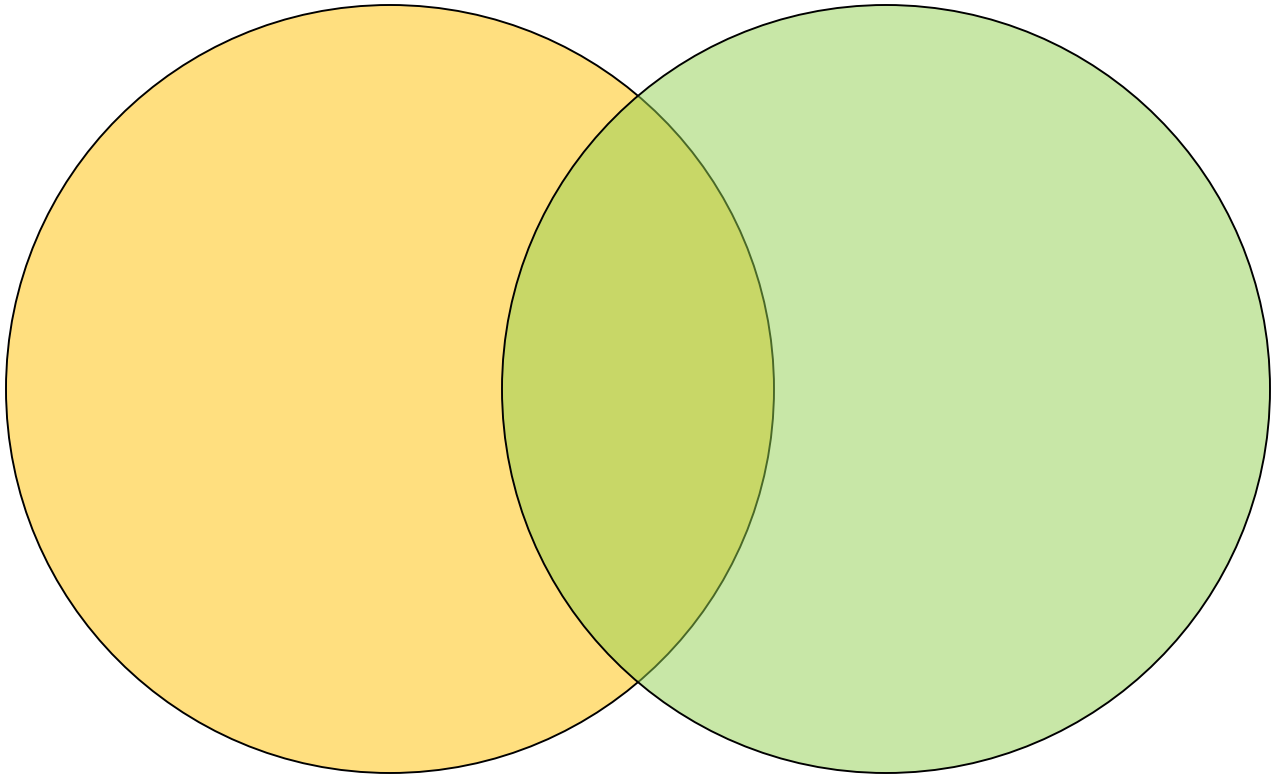


Table 1 Table 2

Right Outer Join

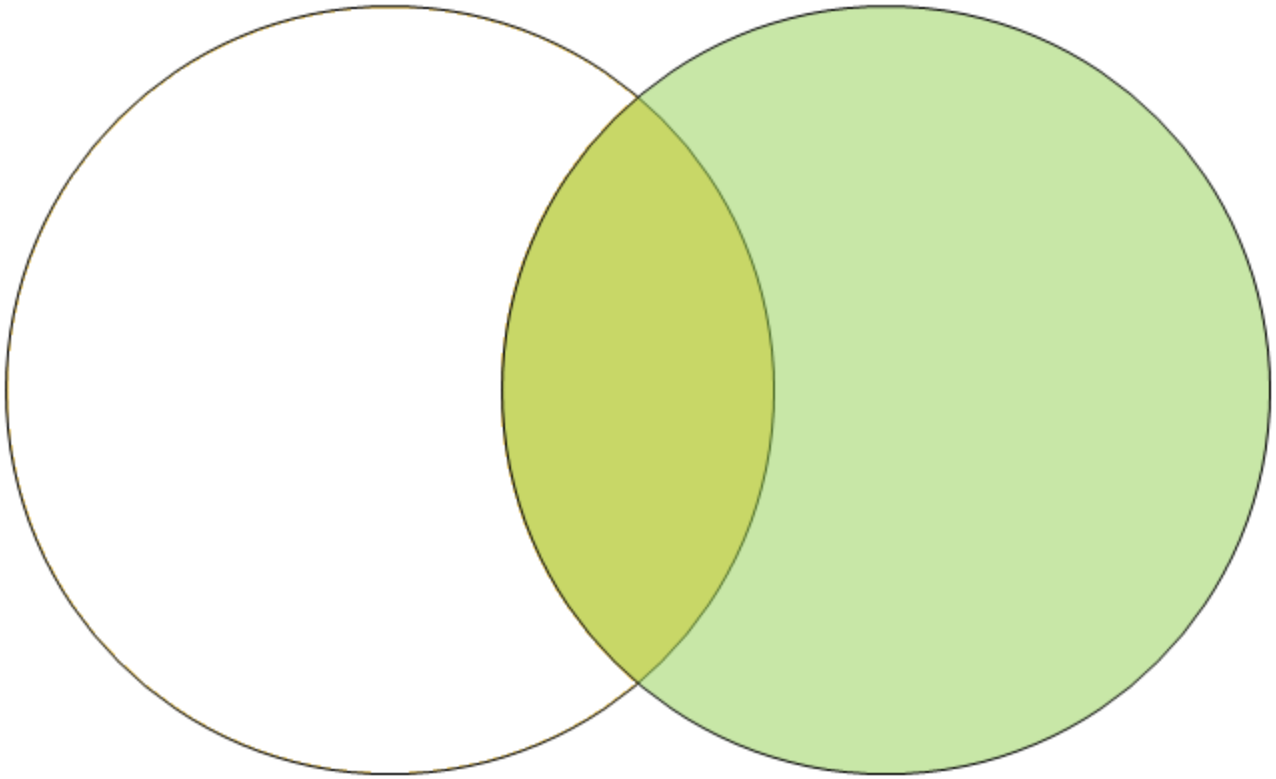


Table 1 Table 2

Scenario 3

Task:

- Troy wants to retrieve all the classes not signed up by any student in the summer.

- **Rahul's hint to Mike:**
  - Learn Right Outer Join

Scenario 4

Task:

- Troy wants to see how big the class can grow if all the students sign up for all the classes in the summer.

- **Rahul's hint to Mike:**
  - Learn Cross Join

Cross Join

- CROSS join is a Cartesian join that does not necessitate any condition to join
- The result set contains records that are multiples of the record number of both the tables

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join

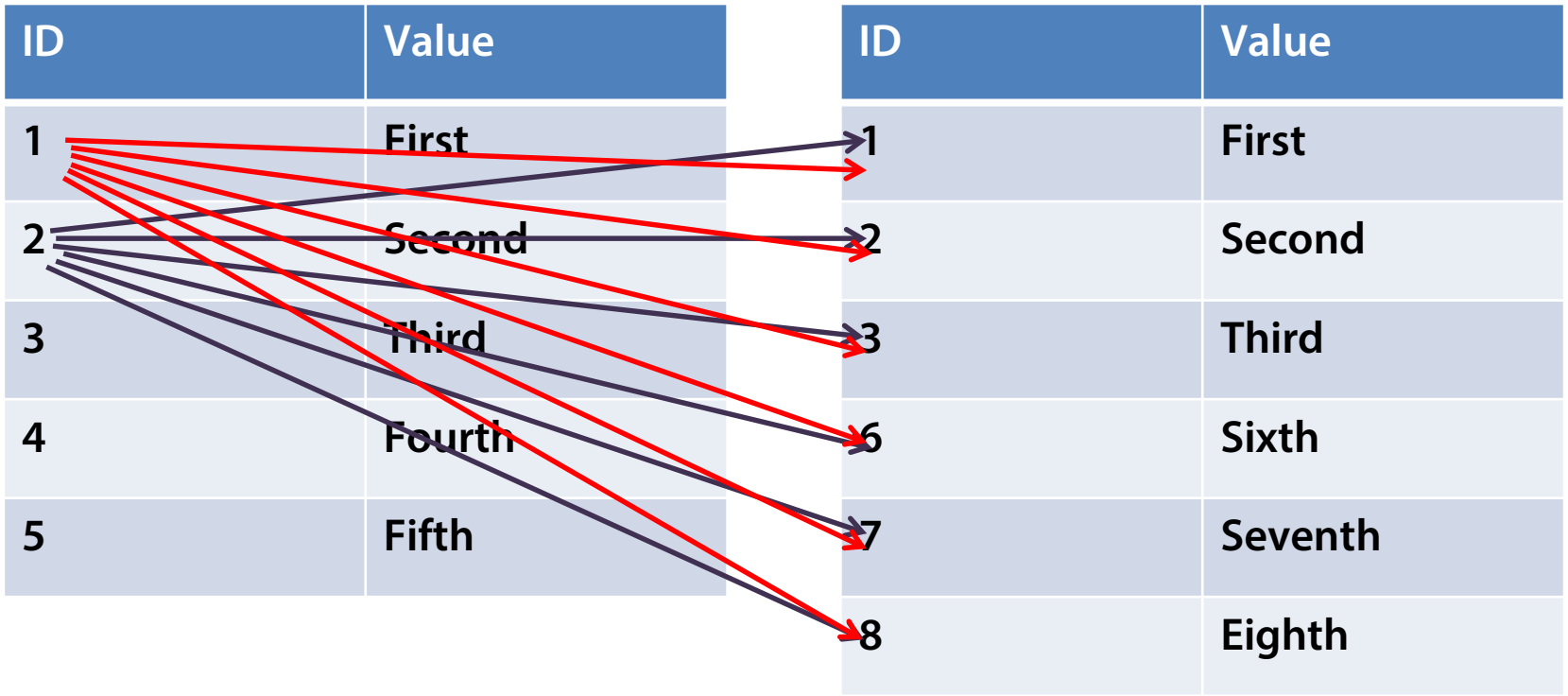
ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

Cross Join



ID	Value
1	First
2	Second
3	Third
4	Fourth
5	Fifth

Table 1

ID	Value
1	First
2	Second
3	Third
6	Sixth
7	Seventh
8	Eighth

Table 2

	ID	Value
1	1	First
2	2	Second
3	3	Third
4	4	Fourth
5	5	Fifth

	ID	Value
1	1	First
2	2	Second
3	3	Third
4	6	Sixth
5	7	Seventh
6	8	Eighth

	ID	Value	ID	Value
1	1	First	1	First
2	1	First	2	Second
3	1	First	3	Third
4	1	First	6	Sixth
5	1	First	7	Seventh
6	1	First	8	Eighth
7	2	Second	1	First
8	2	Second	2	Second
9	2	Second	3	Third
10	2	Second	6	Sixth
11	2	Second	7	Seventh
12	2	Second	8	Eighth

Scenario 4

Task:

- Troy wants to see how big the class can grow if all the students sign up for all the classes in the summer.

- **Rahul's hint to Mike:**
  - Learn Cross Join

Scenario 5

Task:

- Troy wants to see a list of enrolled students along with students who did not sign up for any classes as well as a classes not signed up by any students.

- **Rahul's hint to Mike:**
  - Learn Full Outer Join

Full Outer Join

- FULL OUTER join combines left outer join and right outer join
- This join returns rows from either table when the conditions are met and returns a null value when there is no match

Full Outer Join

Full Outer Join

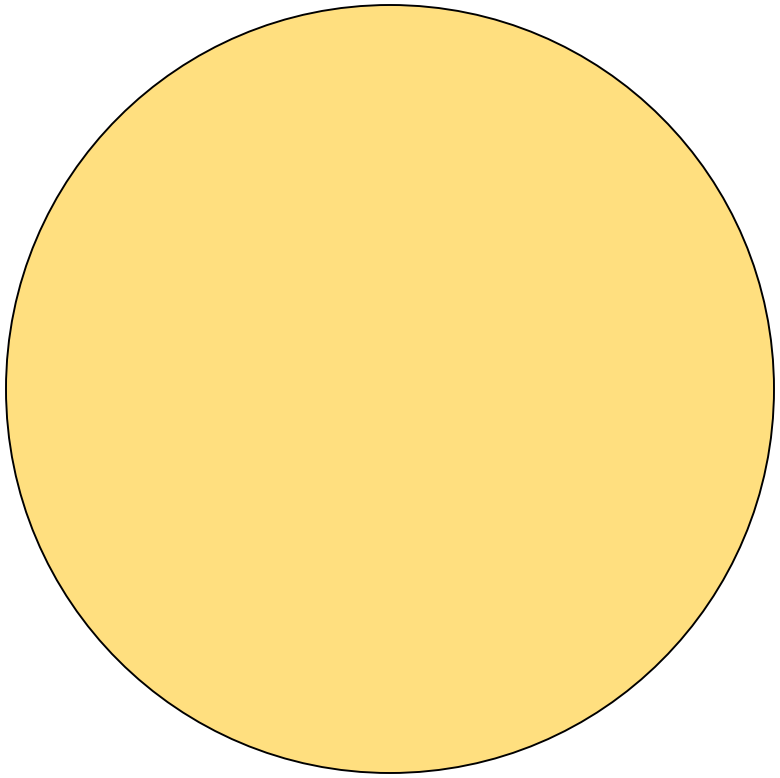


Table 1

Full Outer Join

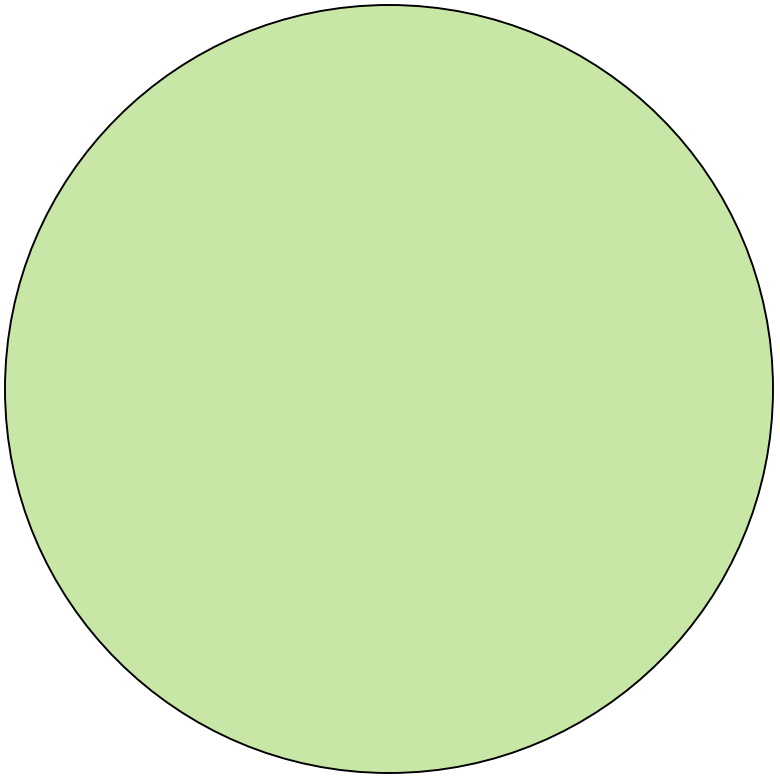


Table 2

Full Outer Join

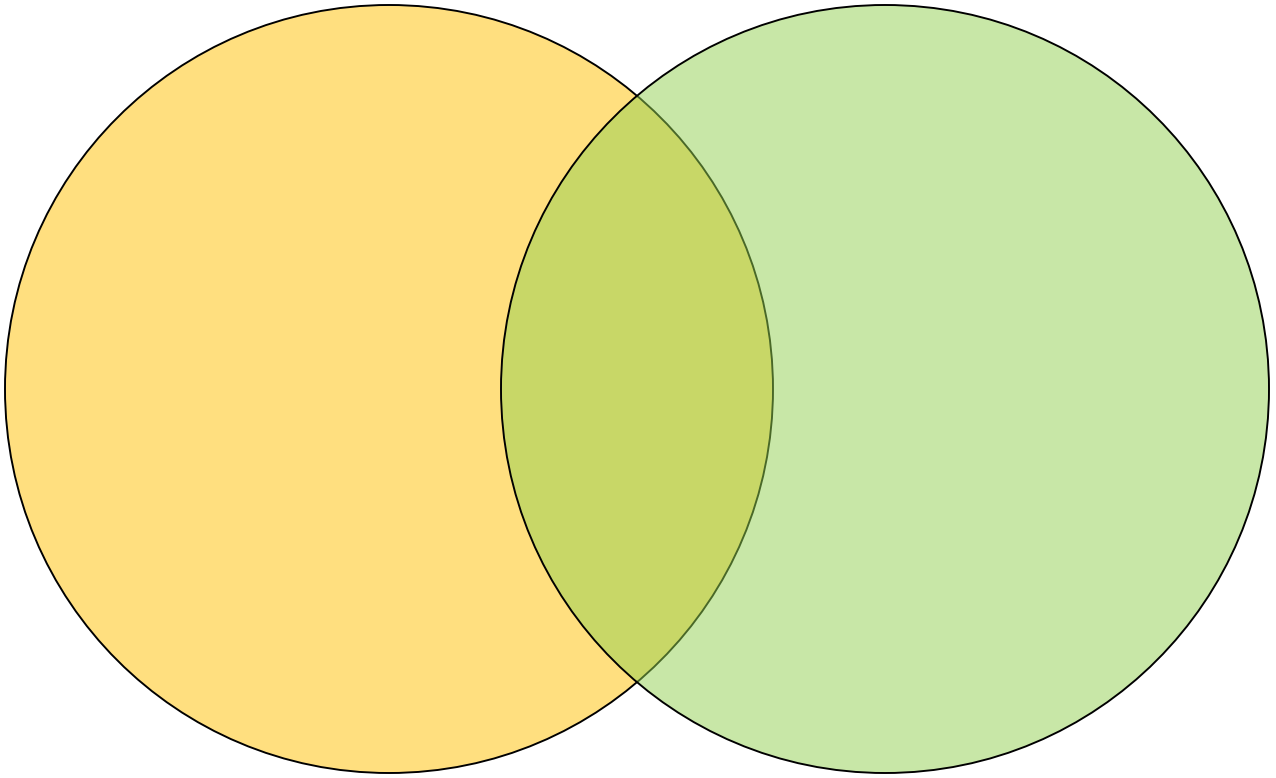
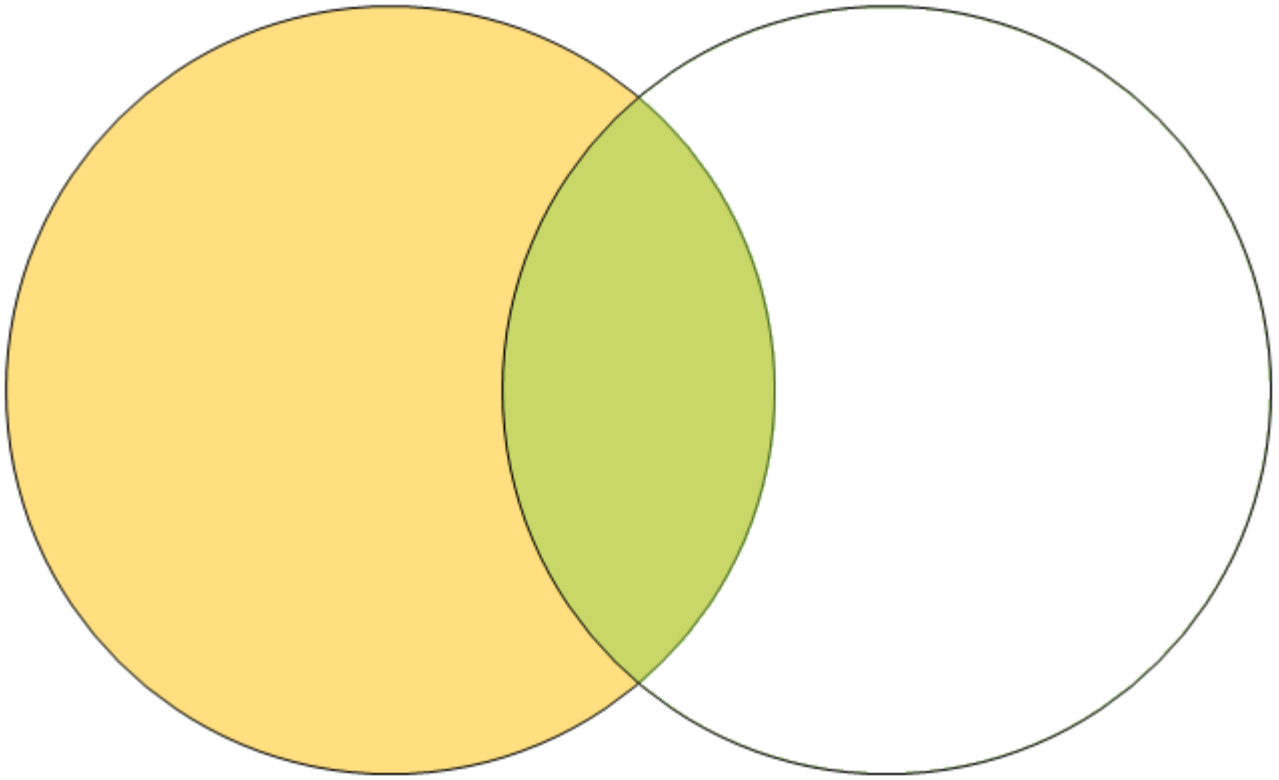


Table 1 Table 2

Left Outer Join



Right Outer Join

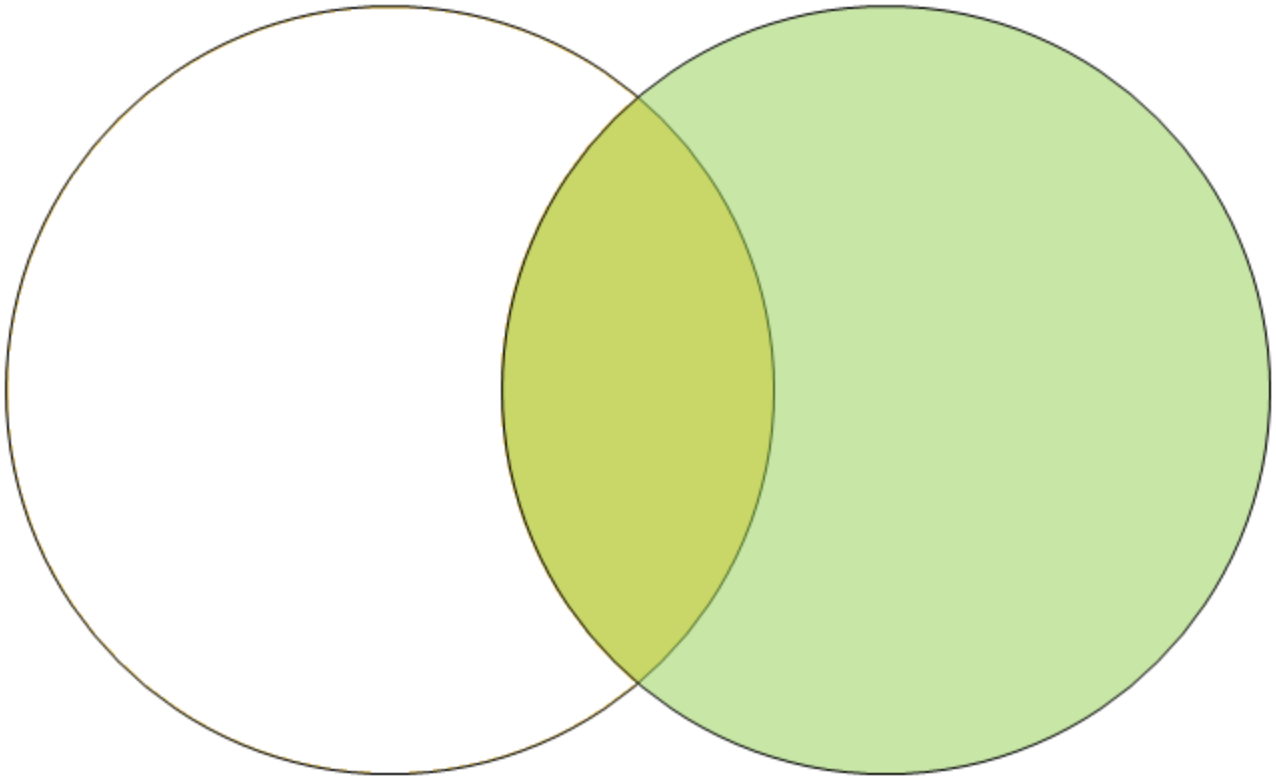


Table 1 Table 2

Full Outer Join

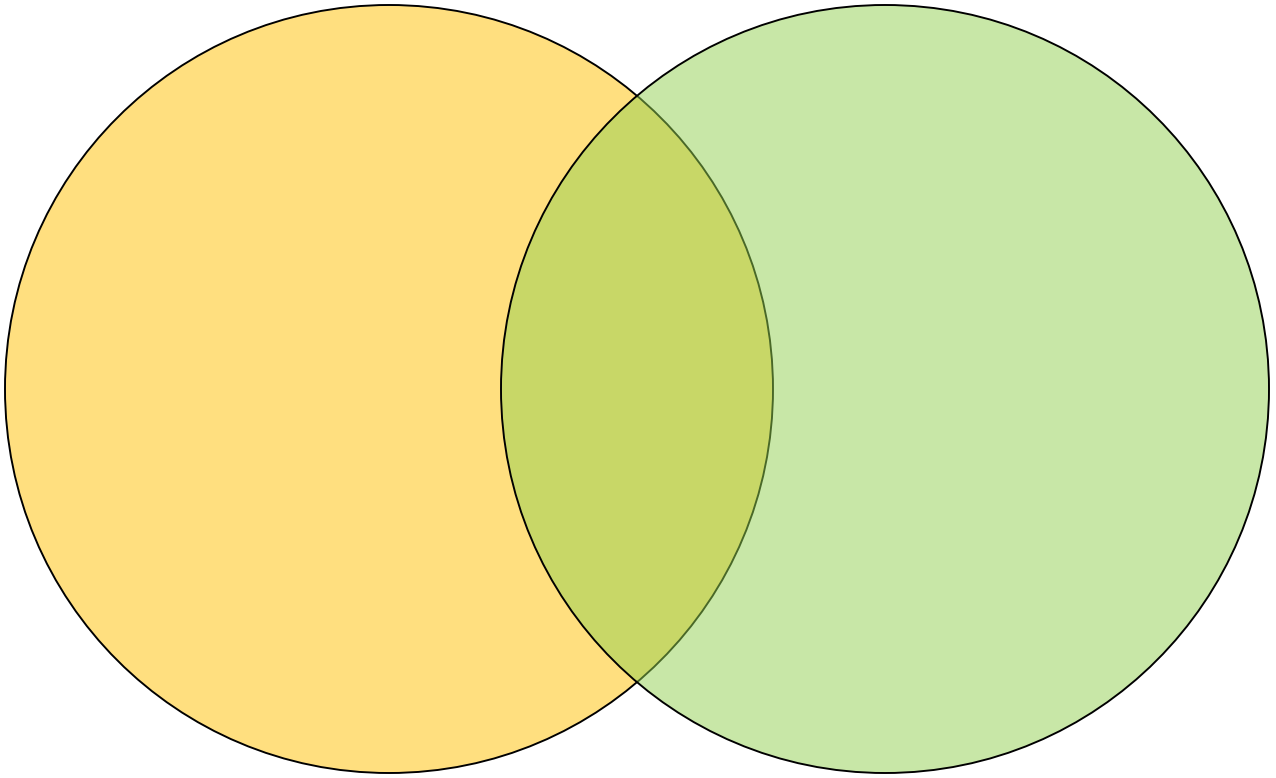


Table 1 Table 2

Scenario 5

Task:

- Troy wants to see a list of enrolled students along with students who did not sign up for any class as well as a class not signed up by any students.

- **Rahul's hint to Mike:**
  - Learn Full Outer Join

Summary

- **A SQL JOIN combines columns from two or more tables in a single result set**
- **Basics of Join**
  - Inner Join
  - Outer Join
    - Left Outer Join
    - Right Outer Join
    - Full Outer Join
  - Cross Join
- **Always alias your column with table to avoid ambiguity in the code**

PostgreSQL: Summary

Pinal Dave
http://blog.sqlauthority.com
@pinaldave

pluralsight
hardcore developer training

In Last Module

- **Retrieving data from more multiple tables**
- **Basics of Join**
  - Inner Join
  - Outer Join
    - Left Outer Join
    - Right Outer Join
    - Full Outer Join
  - Cross Join

In This Module

- Conclusion of scenario
- Important resources
- What Next?

Scenario Conclusion

- Two Database Administrators and a Teacher
- Rahul – Sr. Database Administrator
- Mike – Jr. Database Administrator
- Troy – School Teacher

Icon Courtesy: http://www.icons-land.com/

Scenario Conclusion

- Two Database Administrators and a Teacher
- Rahul – Sr. Database Administrator
- Mike – ~~Jr.~~ Database Administrator
- Troy – School Teacher

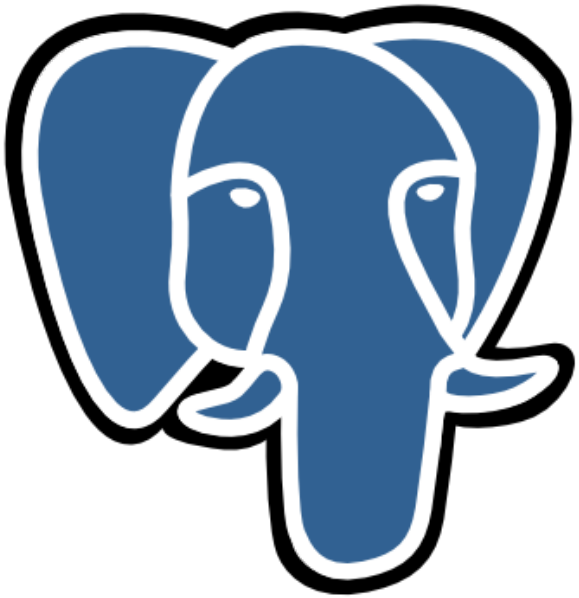
Icon Courtesy: http://www.icons-land.com/

Scenario Conclusion

- Two Database Administrators and a Teacher
- Rahul – Sr. Database Administrator
- Mike – Sr. Database Administrator
- Troy – School Teacher

Icon Courtesy: http://www.icons-land.com/

PostgreSQL
or
Postgres



PostgreSQL

PostgreSQL Limits

Limit	Value
Maximum Database Size	Unlimited
Maximum Table Size	32 TB
Maximum Row Size	1.6 TB
Maximum Field Size	1 GB
Maximum Rows per Table	Unlimited
Maximum Columns per Table	250 - 1600 depending on column types
Maximum Indexes per Table	Unlimited

Important Resources

- **PostgreSQL Download**
  - http://www.postgresql.org/download/
- **Windows Graphical Installer**
  - PostgreSQL Server
  - pgAdmin III – a graphical IDE
  - http://www.postgresql.org/download/windows/
- **Samples Database - *pagilia***
  - http://www.postgresqltutorial.com/postgresql-sample-database/
  - http://bit.ly/pagilia
- **Postgres for .NET Developers by Rob Conery**

Series of 5 Courses

- **Course 1: PostgreSQL: Getting Started**
- Course 2: PostgreSQL: Introduction to SQL Queries
- Course 3: PostgreSQL: Advanced SQL Queries
- Course 4: PostgreSQL: Advanced Server Programming
- Course 5: PostgreSQL: Index Tuning and Performance Optimization

Series of 5 Courses

- Course 1: PostgreSQL: Getting Started
- **Course 2: PostgreSQL: Introduction to SQL Queries**
- Course 3: PostgreSQL: Advanced SQL Queries
- Course 4: PostgreSQL: Advanced Server Programming
- Course 5: PostgreSQL: Index Tuning and Performance Optimization

Thank YOU!

Pluralsight Discussion Forum

http://twitter.com/pinaldave

http://facebook.com/SQLAuth