



QuantSprout

Intermediate SQL

by Craig Sakuma

Introductions

Craig Sakuma

- Founder of QuantSprout
- General Assembly Instructor for Data Science
- MBA from Wharton
- B.Eng from Northwestern University

Fun Fact

Developed a novelty
BBQ product that was
featured in USA Today



Class Introductions

- Name
- What's your job?
- How do you plan to apply skills from today's workshop?
- Fun Fact

Objectives for Class

- Use SQL Workbench software to connect to database
- Import and export data
- Apply advanced querying techniques
- Basic database administration commands
- Build your own databases in the cloud

Course Structure

- Lectures on topics
 - Interaction is good
 - Feel free to ask questions
 - If there's not enough time to cover questions, we'll put it in a parking lot for after class
- Hands on exercises
 - Pair programming
 - Mix up partners

Set Up for Advanced SQL

- Download and install SQLWorkbench/J
www.sql-workbench.net/downloads.html
- SQLWorkbench/J might require you to install or update your Java software
java.com/en/download/
- Download the PostgreSQL JDBC driver
(note: you won't be able to open this file)
jdbc.postgresql.org/download.html

SQL Workbench

- Open Source software for communicating SQL commands to databases
- Compatible with a variety of SQL versions
- Connects to individual databases using credentials and server information
- Read and Save SQL queries as text files
- Easy to Import / Export data to and from CSV files

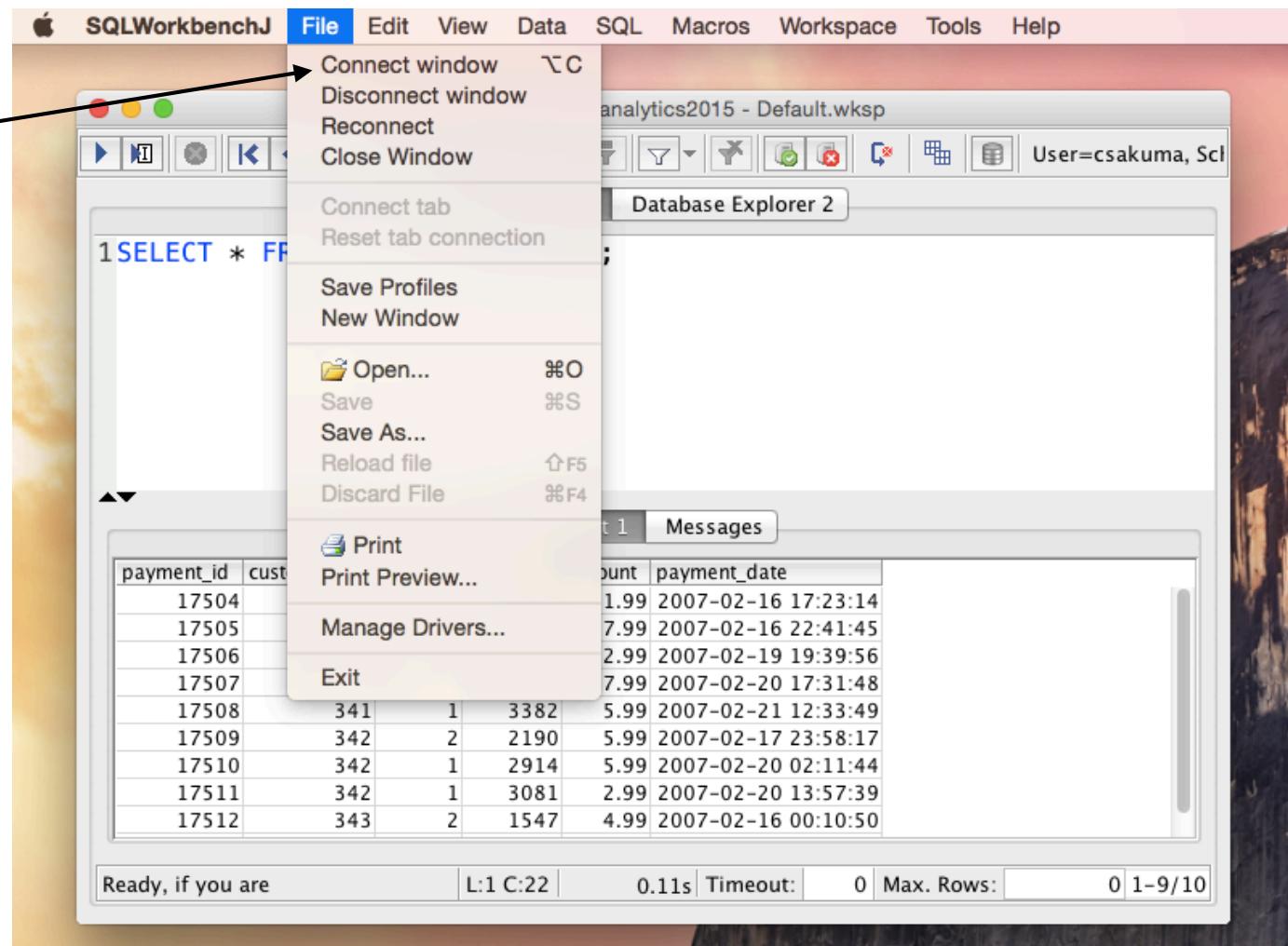
Download Driver

Driver files are used to communicate between SQL Workbench and each specific version of SQL (e.g., Postgres, MySQL, Oracle, etc...)

- Create folder called drivers in your home directory
- Download PostgreSQL driver
 - <https://jdbc.postgresql.org/download.html>
- Move file into your drivers folder

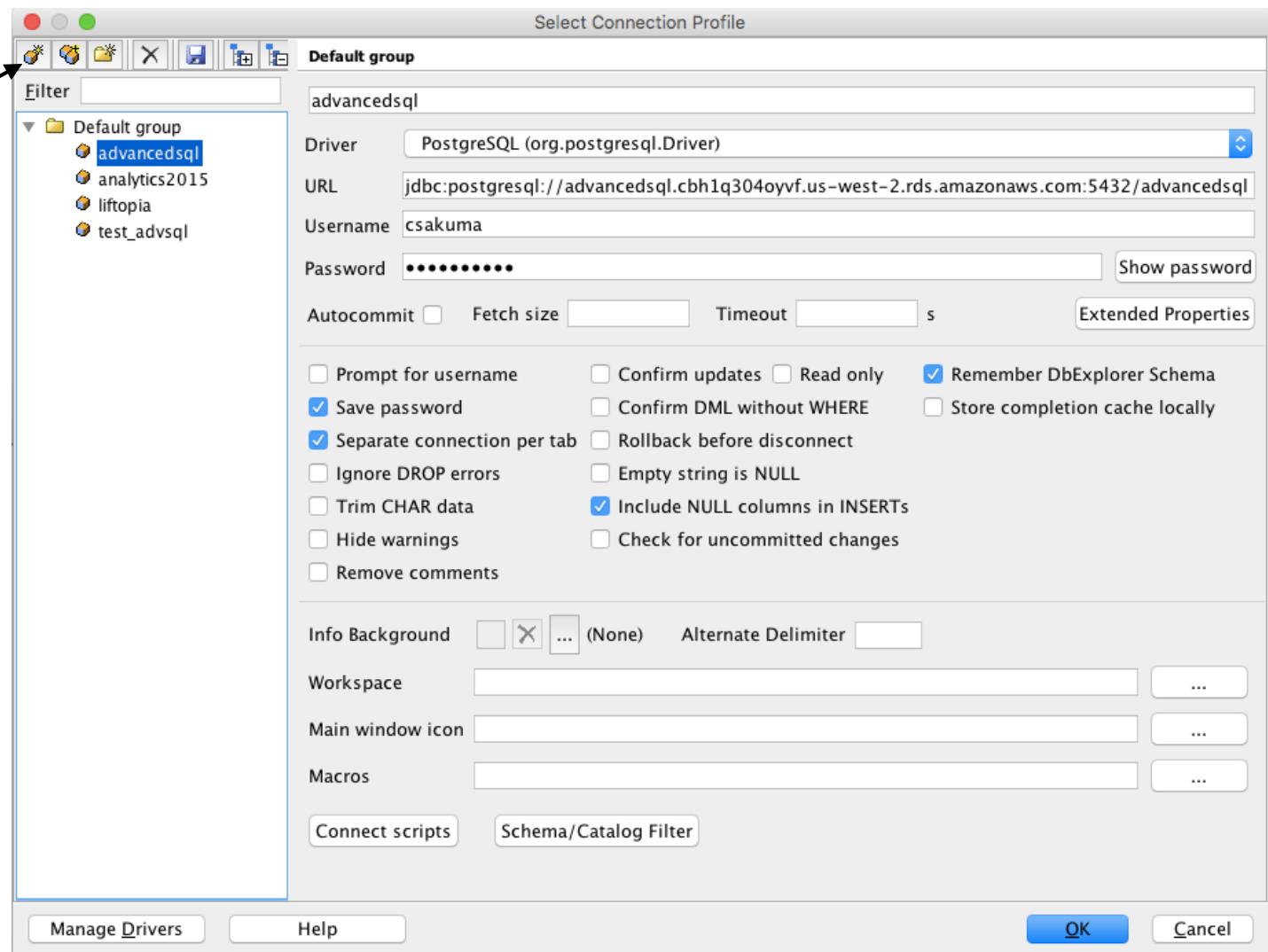
Open Connect Window

Select from
File Menu



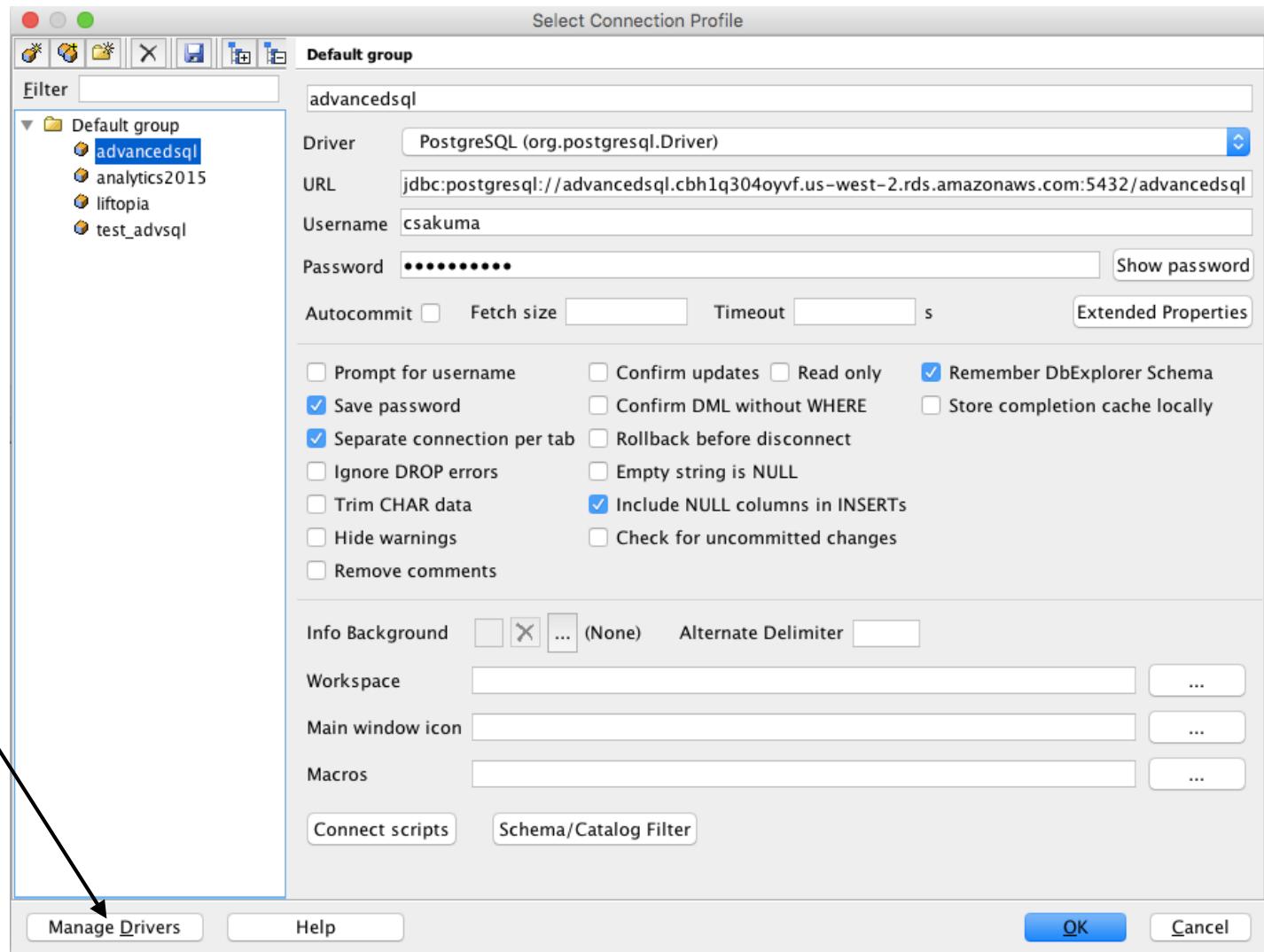
Create New Profile

Click to Create
New Connection
Profile



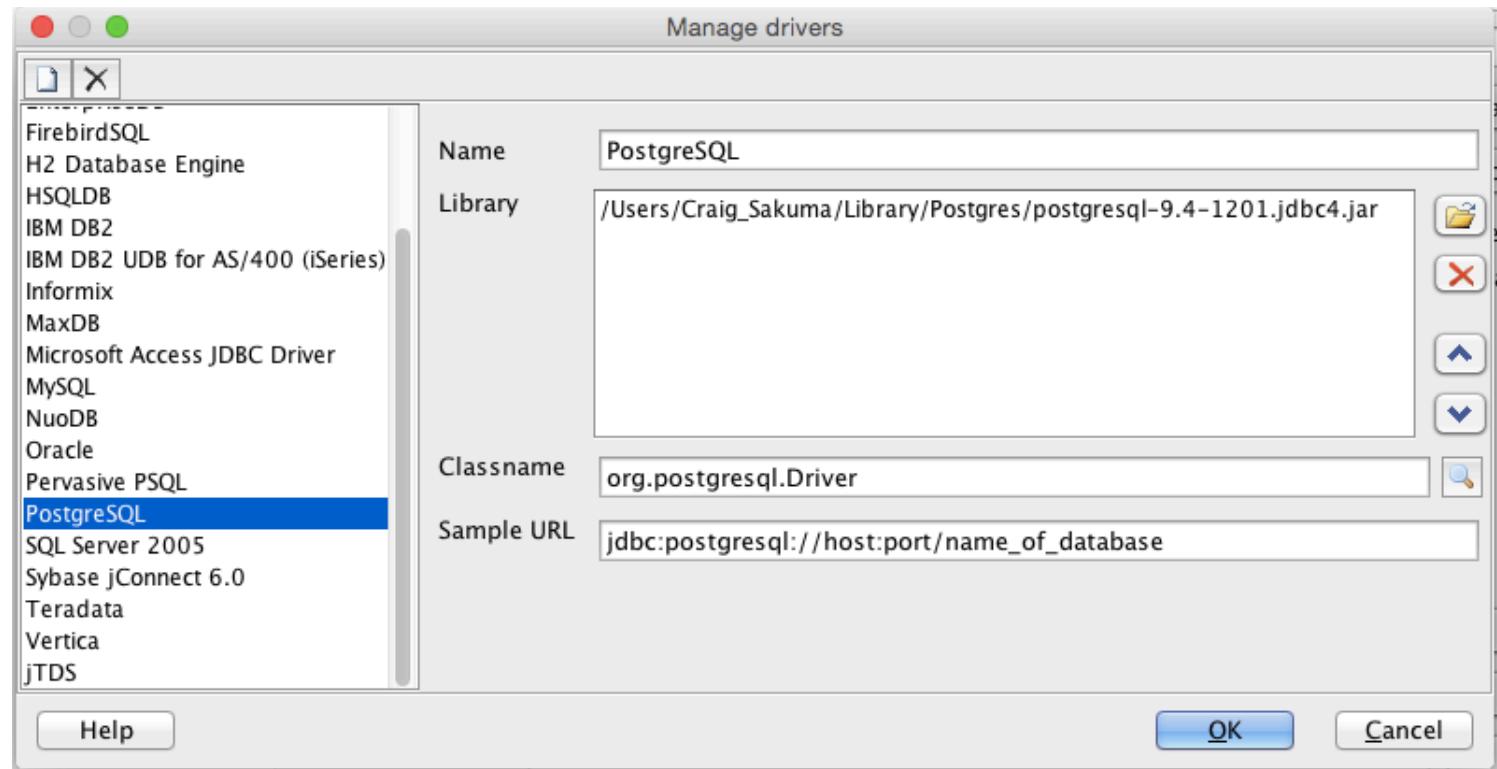
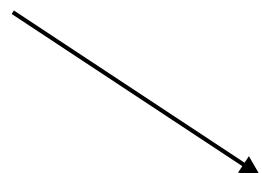
Update Driver

Click Manage Drivers

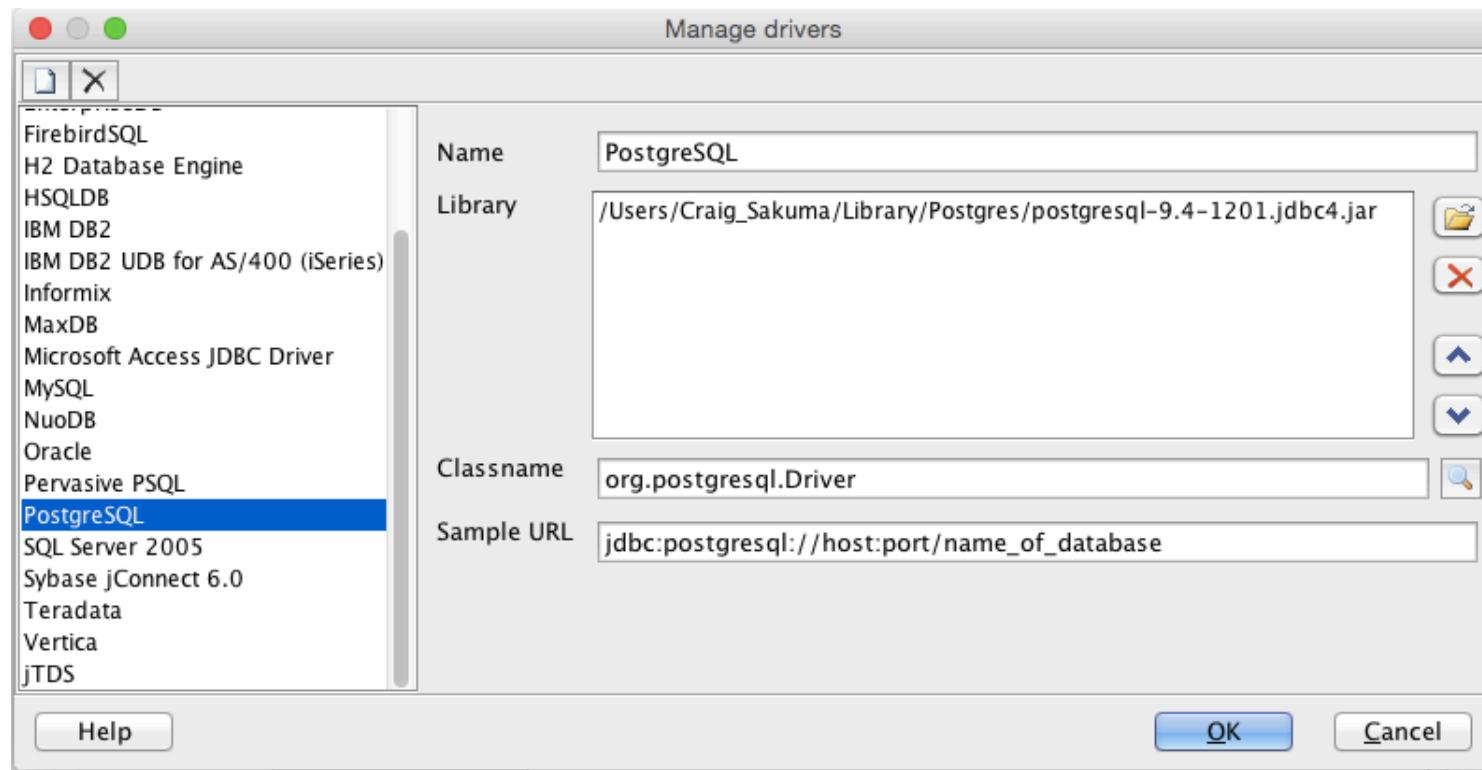


Update Driver

Select
PostgreSQL



Update Driver



Click on
folder icon
and select
Postgres
Driver from
your drivers
folder

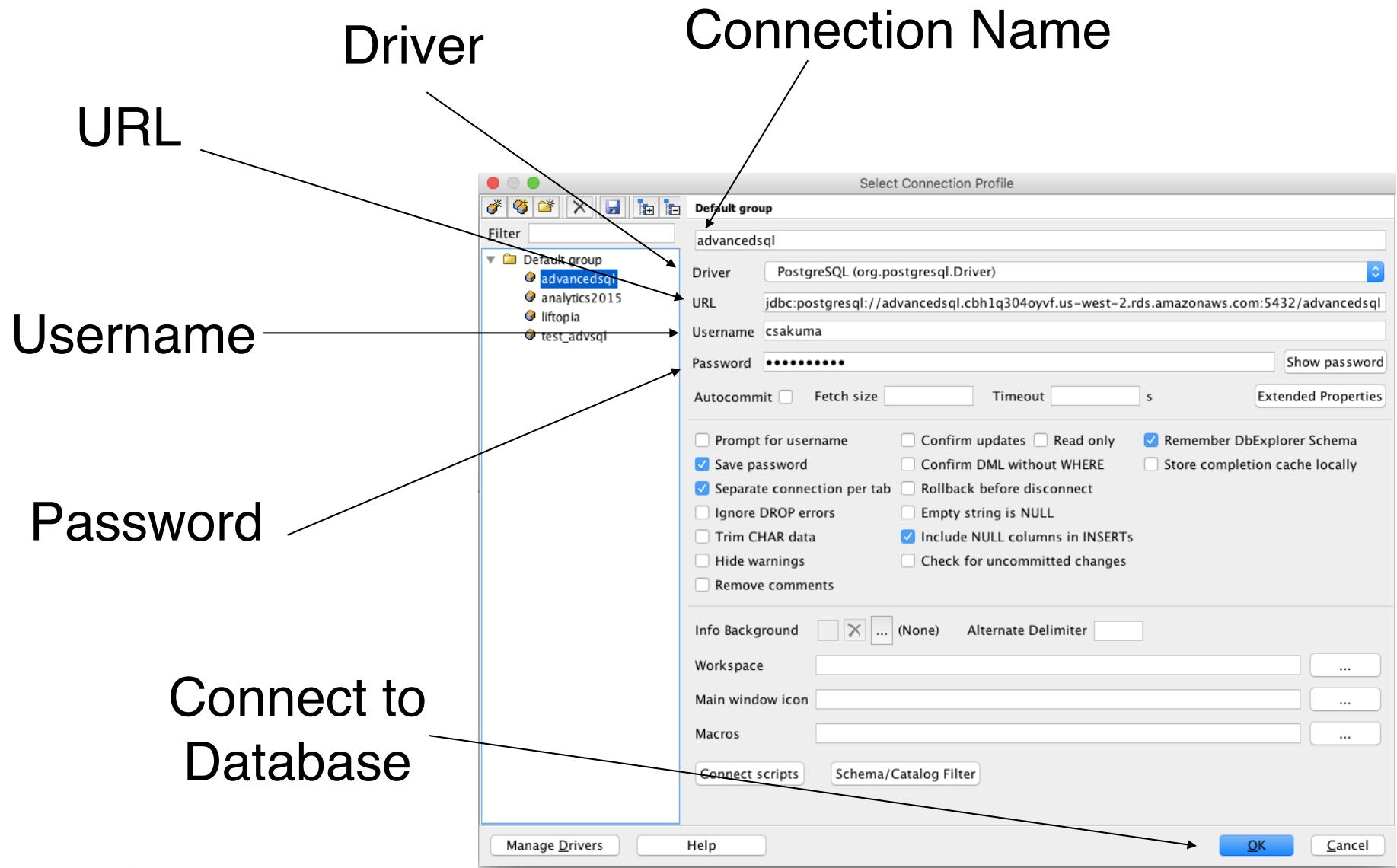
Connect to Database

- Set Up Connection (requires credentials)
 - Host
 - Port
 - Database Name
 - Username
 - Password
- URL for SQLWorkbench uses following format:
 - `jdbc:postgresql://<Host>:<Port>/<DB Name>`

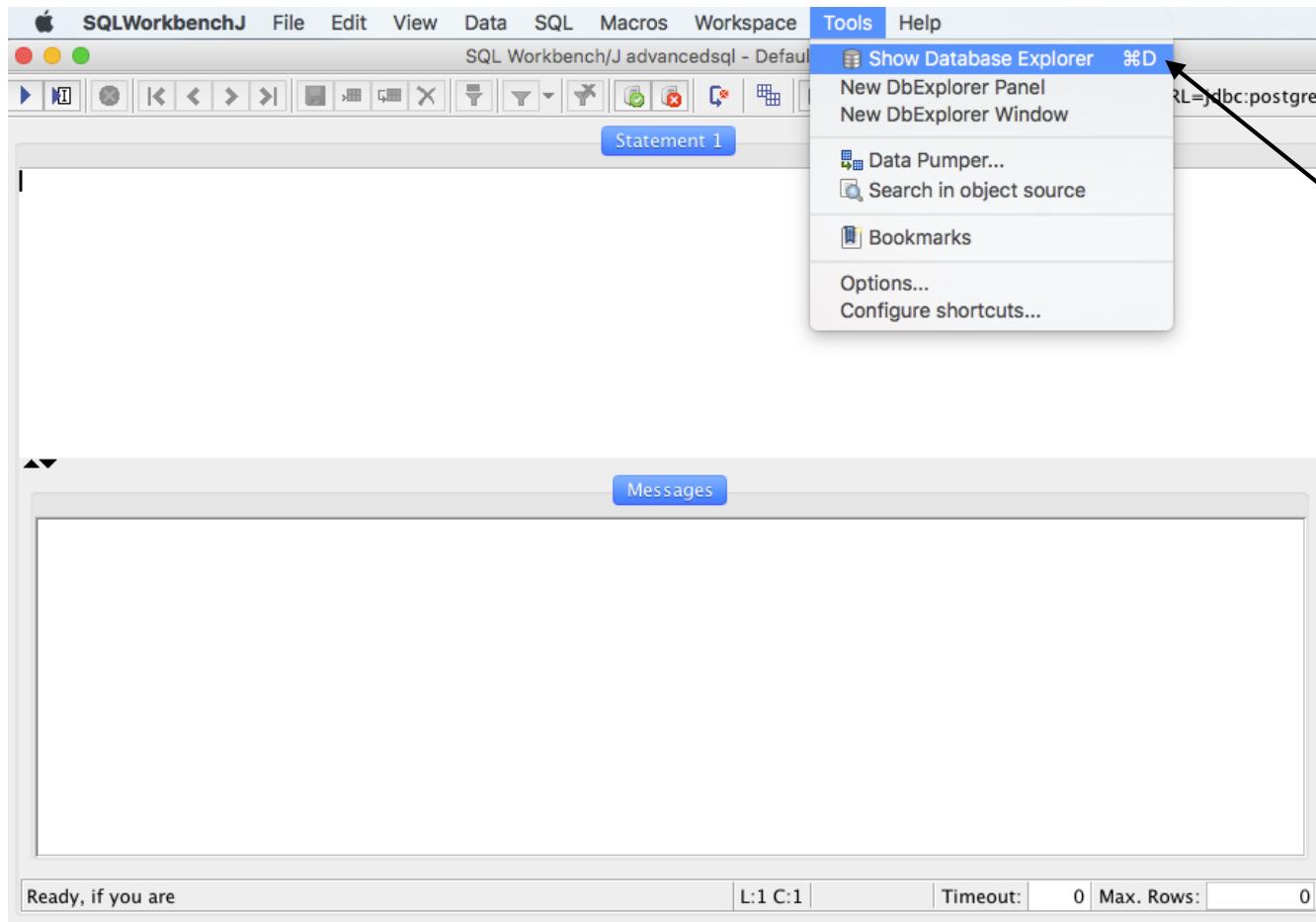
Credentials for Database

- URL:
jdbc:postgresql://quantsprout.cbh1q304oyvf.us-west-2.rds.amazonaws.com:5432/QS
- User Name: your registration name
(lower case with spaces replaced with underscore)
- Password: quantsprout

Add Credentials



View Database Explorer



Select
Database
Explorer
From Tools

Explore Database

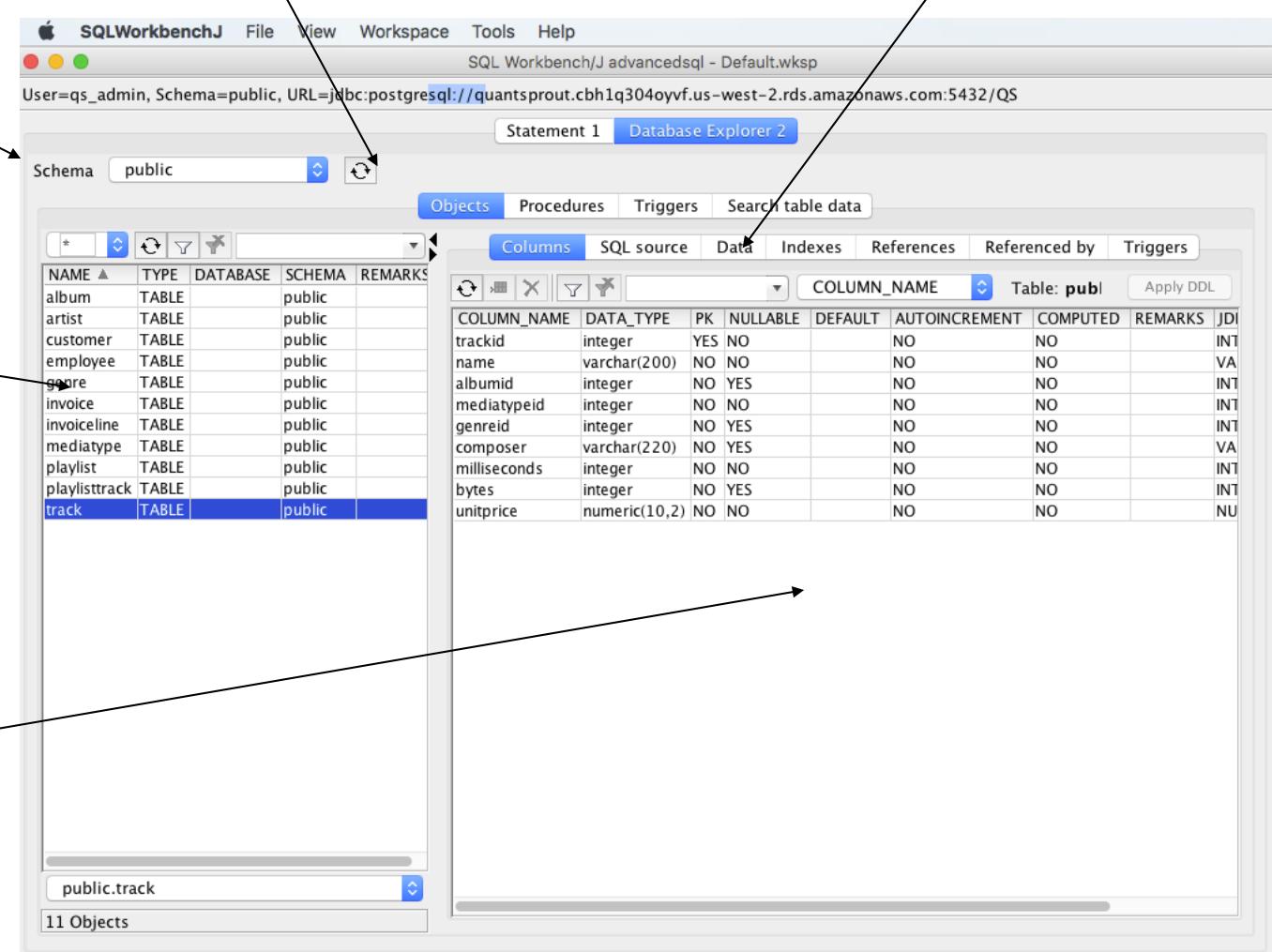
Select Schema

Table List

Details for Selected Table

Refresh View

View Data Sample



Write Query

Text Editor
for SQL code

The screenshot shows the SQLWorkbenchJ application interface. At the top is a menu bar with File, Edit, View, Data, SQL, Macros, Workspace, Tools, and Help. Below the menu is a toolbar with various icons. The main window has two tabs: Statement 1 (selected) and Database Explorer 2. In the Statement 1 tab, the following SQL code is written:

```
SELECT * FROM track LIMIT 10;
```

Below the code is a large table titled "Result 1" showing the query results. The table has columns: trackid, name, albumid, mediatypeid, genreid, composer, and millisec. The data consists of 10 rows of song information. An arrow points from the text "Window for Results" to the Result 1 table.

trackid	name	albumid	mediatypeid	genreid	composer	millisec
1	For Those About To Rock (We Salute You)	1	1	1	Angus Young, Malcolm Young, Brian Johnson	343
2	Balls to the Wall	2	2	1		342
3	Fast As a Shark	3	2	1	F. Baltes, S. Kaufman, U. Dirksneider & W. Hoffman	230
4	Restless and Wild	3	2	1	F. Baltes, R.A. Smith-Diesel, S. Kaufman, U. Dirksneider & W. Hoffman	252
5	Princess of the Dawn	3	2	1	Deaffy & R.A. Smith-Diesel	375
6	Put The Finger On You	1	1	1	Angus Young, Malcolm Young, Brian Johnson	205
7	Let's Get It Up	1	1	1	Angus Young, Malcolm Young, Brian Johnson	233
8	Inject The Venom	1	1	1	Angus Young, Malcolm Young, Brian Johnson	210
9	Snowballed	1	1	1	Angus Young, Malcolm Young, Brian Johnson	203
10	Evil Walks	1	1	1	Angus Young, Malcolm Young, Brian Johnson	263

Ready, if you are L:1 C:30 | 0.15s Timeout: Max. Rows: 0 1-10/10

SQL – Comment

Comment out code that you don't want executed

```
-- <single line of text>  
/* <multiple lines of text>  
<multiple lines of text> */
```

Shortcut keys:

SQL workbench = **command + c + shift**

Review Query Commands

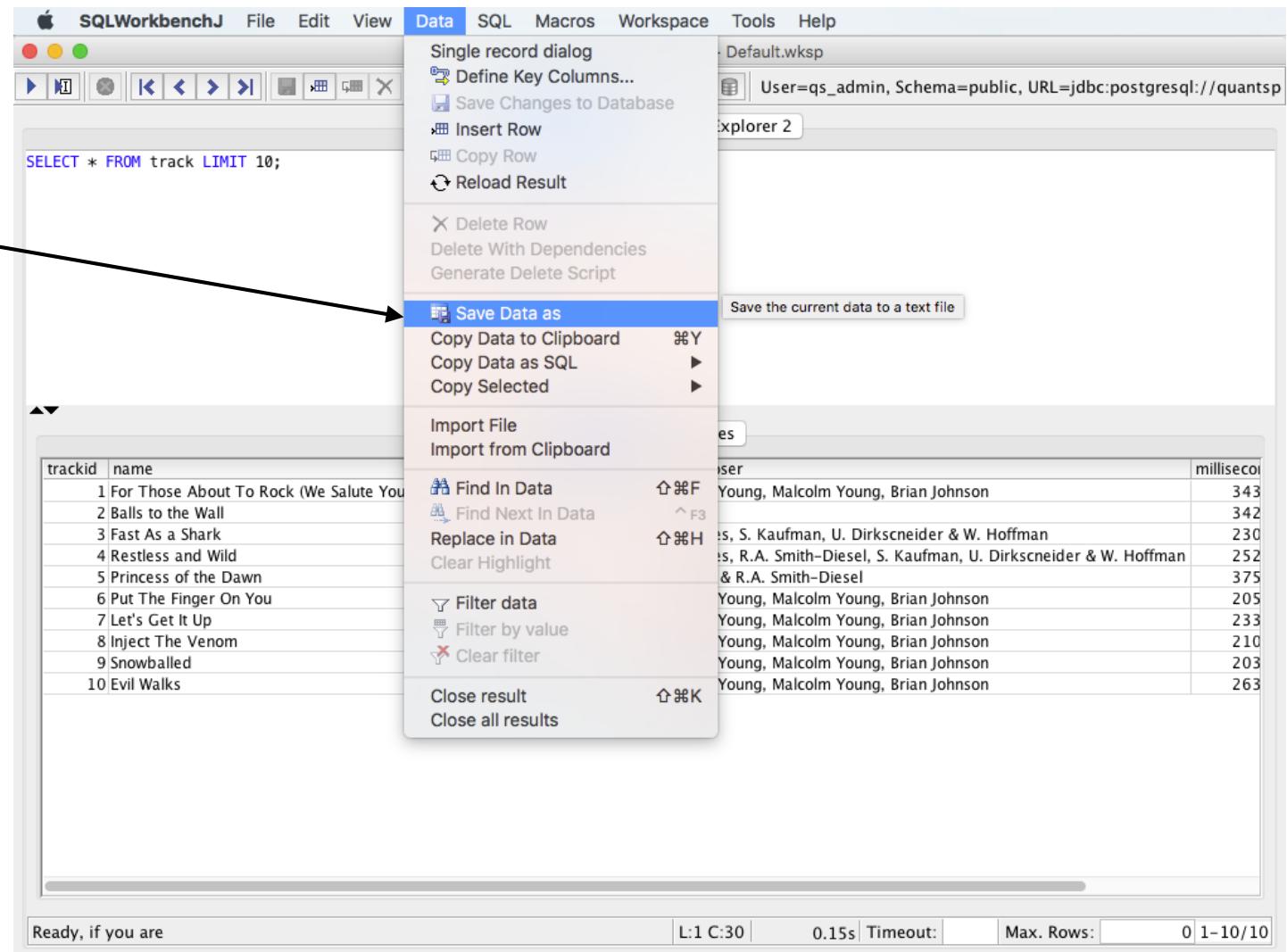
- **SELECT**: choose the fields for query
- **FROM**: pick table(s) for data source
- **WHERE**: filter data based upon conditions
- **GROUP BY**: segment data into groups
- **ORDER BY**: sort results
- **LIMIT**: limit the number of records returned
- **JOIN Types**:
 - INNER vs. OUTER
 - LEFT vs. RIGHT

Review Exercise

- How many days of content are there in the library?
- What are the longest songs (excluding video)?
- What is the average length of a song grouped by genre (convert time to minutes)?

Export Data

Select Save Data

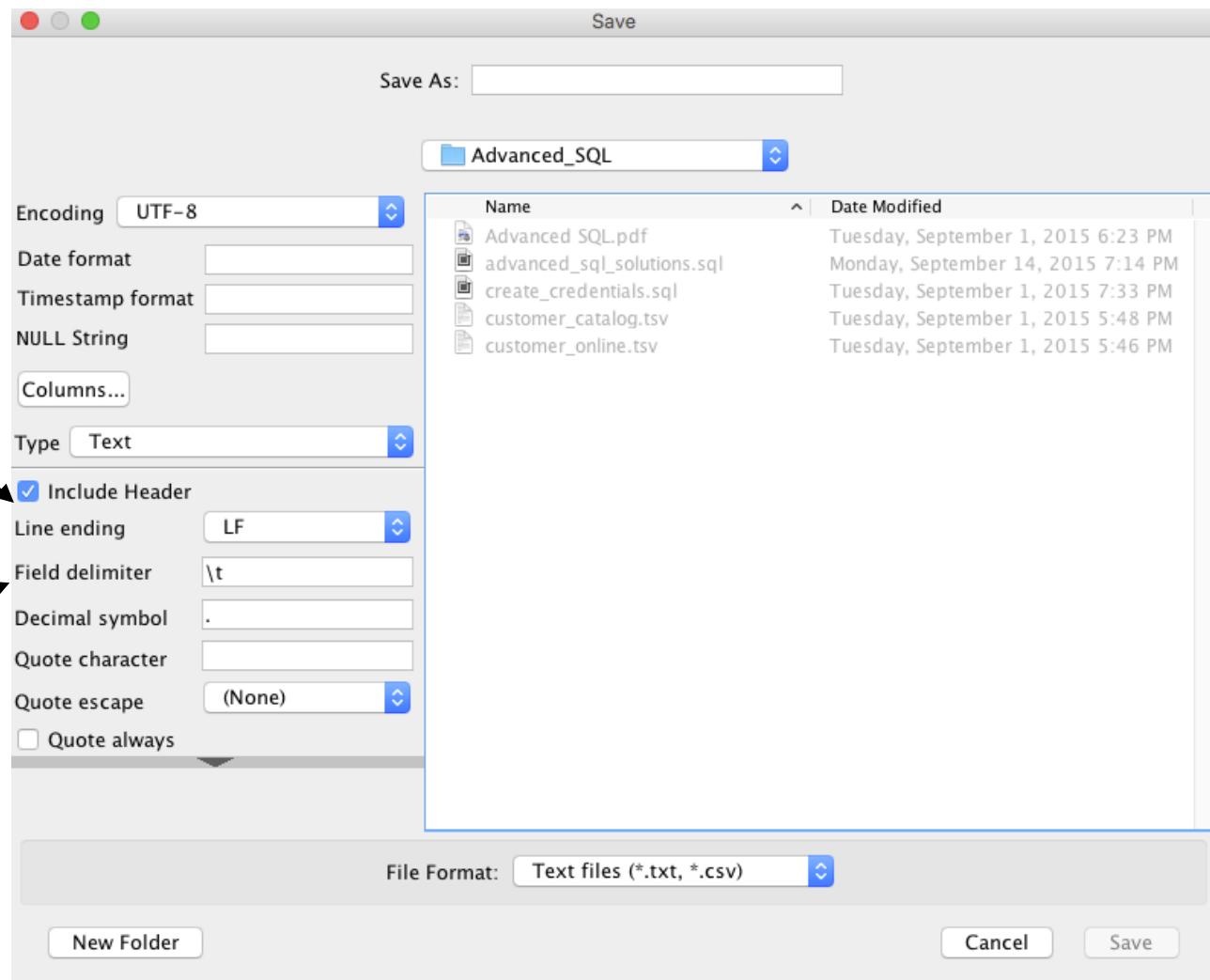


Export Data

Option for
Headers

Pick Delimiter

- tab: \t
- pipe: |
- comma: ,



Aliases

Rename fields in your queries:

- `SELECT <field> AS <alias>`
- `SELECT milliseconds/1000 AS seconds`

Reference tables as abbreviations:

- `FROM <table> <alias>`
- `FROM track t
JOIN genre g
ON t.genreid = g.genreid`

Aliases

Reference fields by position:

- ```
SELECT <field1>, <field2>
 FROM <table>
 ORDER BY 1, 2
```
- ```
SELECT name, milliseconds
      FROM track
      ORDER BY 2 DESC
```

Extract

- Syntax:
SELECT EXTRACT (<date_component> FROM <field>)
- Example:
SELECT EXTRACT(month FROM invoicedate)
FROM invoice;
- Date/Time Components

Day	Hour	DOW
Week	Minute	DOY
Month	Second	Quarter
Year		Timezone

Having

- Syntax:
SELECT <field1>, <agg function>(<field2>)
FROM <table>
GROUP BY <field1>
HAVING <agg function>(<field2>) <operator> <value>;
- Example:
SELECT genreid, COUNT(trackid)
FROM track
GROUP BY genreid
HAVING COUNT(*) > 50;

Exercise #1

- Which customers have spent more than \$40 (Use Group By and Having for the **customer** and **invoice** tables)?
- What are the total sales by month (Use Extract and Group By and the **invoice** table)?
- Create a roster of employees with their bosses (Join the **employee** table to itself by using table aliases)

Case Statements

- Apply conditional logic

```
SELECT CASE
```

```
    WHEN <condition1> THEN <result1>
```

```
    WHEN <condition 2> THEN <result 2>
```

```
    ELSE <result 3>
```

```
END
```

- Example:

```
SELECT CASE
```

```
    WHEN age < 18 THEN 'child'
```

```
    WHEN age >= 60 THEN 'senior'
```

```
    ELSE 'adult'
```

```
END AS age_market_segment
```

Union

- Merges data from two queries by stacking results on top of each other
- Must have same number of columns and corresponding data types
- Duplicate results are removed by default
- UNION ALL will include duplicates
- Syntax:
SELECT * FROM <table1>
UNION
SELECT * FROM <table2>

Coalesce

Picks first non-null value

- Syntax:
COALESCE ([field1], [field2], [field3])
- Example:
COALESCE(online.firstname, catalog.firstname) as
firstname

Exercise #2

- Using the **iowa liquor products** table, create an alcohol type label for whisky, vodka, tequila, rum, brandy, schnapps and any other liquor types (hint: use CASE STATEMENT and LIKE)
- Using the **catalog** and **online** tables, create a customer list that combines the names from the catalog and online tables using UNION without creating duplicates.
- FULL JOIN the **catalog** and **online** tables and inspect the results. Try adding the catalog sales and online sales totals together. Why do you get errors?

Window Functions

- Perform calculation across a set of table rows that are somehow related to the current row
- Applications:
 - Cumulative sales
 - Percentile rank
 - Group level results

Cumulative Sales

- Syntax:

```
SELECT sum(<field1>) OVER  
    (ORDER BY <field2>)  
FROM <table>;
```

- Example:

```
SELECT invoicedate, SUM(total)  
OVER (ORDER BY invoicedate)  
FROM invoice;
```

Percentiles

- Syntax:

```
SELECT NTILE(<# of groups>) OVER  
    (ORDER BY <field>)  
FROM <table>;
```

- Example:

```
SELECT name, milliseconds, NTILE(100)  
    OVER (ORDER BY milliseconds DESC) AS  
        percentile  
FROM track;
```

Group Level Results

- Syntax:

```
SELECT AVG(<field1>) OVER  
    (PARTITION BY <field2>)  
FROM <table>;
```

- Example:

```
SELECT name, genreid, milliseconds,  
    AVG(milliseconds)  
    OVER (PARTITION BY genreid)  
FROM track;
```

Subqueries

- Syntax:

```
WITH <subquery_name> AS (
    <code_for_subquery>
<code for parent query>;
```

- Example:

```
WITH top_customers AS (
    SELECT customerid, SUM(total) as sales
    FROM invoice
    GROUP BY customerid)
SELECT COUNT(*)
FROM top_customers
WHERE sales > 40;
```

Views

- Syntax:,
CREATE VIEW <view_name> AS <query_code>;
- Example:
CREATE VIEW top_customers AS
 SELECT customerid, SUM(total) as sales
 FROM invoice
 GROUP BY customerid
 ORDER BY sales DESC;

NOTE: You can't have duplicate field names in a view

Temporary Tables

- Syntax:

```
CREATE TEMP TABLE <table_name> AS  
<query_code>;
```

- Example:

```
CREATE TEMP TABLE total_sales AS  
    SELECT customerid, SUM(total) as sales  
    FROM invoice  
    GROUP BY customerid  
    ORDER BY sales DESC;
```

Exercise #3

- How many iowa liquor vendors have more than \$1 million in 2014 sales (hint: use subquery to group sales by vendor)?
- Group sales by month with a subquery and then calculate cumulative sales by month for 2014 (using iowa sales table)
- Create a View that adds liquor type to the iowa product data. Don't forget to commit your changes.

Regular Expressions

- Programming language for pattern matching text
- Similar in functionality to LIKE but much more powerful
- Syntax:
SELECT * FROM [table_name]
WHERE [field1] ~ [pattern];

Regular Expressions

- Example (all artists with names ending with n):
SELECT * FROM artist
WHERE name ~ 'n\$';
- Example (all artists that start with z):
SELECT * FROM artist
WHERE name ~ '^z';

Amazon Web Services

- Cloud services that can be rented
 - Pay based on usage (e.g., hours of use, volume of data)
 - Simple to set up and shutdown
 - Services are very scalable
 - Excellent security options
- Wide range of products:
 - Virtual Servers in the Cloud – EC2 (Elastic Compute Cloud)
 - Cloud file storage – S3 (Simple Storage Service)
 - Databases as a service – RDS (Relational Database Service)
 - Data Warehouses – Redshift

AWS Relational Database Service

- Set Up
 - Provision server in the cloud (e.g., EC2)
 - Download and install software
- Database Administration
 - Allocate storage space for database
 - Backup of data
 - Mirror data across multiple locations
- Database Maintenance
 - Upgrade software when new releases and patches
 - Scale database as data grows

Outsource Database Operations to Amazon

Create AWS Account

Follow registration process at aws.amazon.com:

- Create User ID and Password
- Add contact info
- Add credit card info (we'll be using free service so you won't be charged)
- Add phone number and verify identity via call
- Select Basic(Free) support plan
- Launch Management Console

AWS Console

The screenshot shows the AWS Console homepage. At the top is a dark header bar with the AWS logo, a "Services" dropdown, a "Resource Groups" dropdown, and a search icon. Below the header is a white content area titled "AWS services". A search bar at the top of this area contains the placeholder text "Find a service by name or feature (for example, EC2, S3 or VM, storage)." Below the search bar are two expandable sections: "Recently visited services" and "All services". The "All services" section is expanded, displaying services categorized into groups: Compute, Management Tools, Mobile Services, AR & VR, Application Integration, Media Services, and Customer Engagement. An arrow points from the text "View RDS Dashboard" on the left to the "RDS" link under the Database category.

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Recently visited services

All services

Compute

- EC2
- Lightsail
- Elastic Container Service
- Lambda
- Batch
- Elastic Beanstalk

Storage

- S3
- EFS
- Glacier
- Storage Gateway

Database

- RDS
- DynamoDB
- ElastiCache
- Amazon Redshift

Management Tools

- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks
- Service Catalog
- Systems Manager
- Trusted Advisor
- Managed Services

Mobile Services

- Mobile Hub
- AWS AppSync
- Device Farm
- Mobile Analytics

AR & VR

- Amazon Sumerian

Application Integration

- Step Functions
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

Media Services

- Elastic Transcoder
- Kinesis Video Streams
- MediaConvert
- MediaLive
- MediaPackage
- MediaStore
- MediaTailor

Customer Engagement

- Amazon Connect
- Pinpoint

View RDS
Dashboard

RDS Dashboard

The screenshot shows the AWS RDS Dashboard landing page. At the top, there's a navigation bar with the AWS logo, a search bar, and user information (Craig M Sakuma, Oregon, Support). On the left, a sidebar menu lists options like Instances, Reserved Purchases, Snapshots, Security Groups, Parameter Groups, Option Groups, Subnet Groups, Events, Event Subscriptions, and Notifications. The main content area features a large blue circular icon and the text "Amazon Relational Database Service". Below it is a descriptive paragraph about Amazon RDS and a "Get Started Now" button. An arrow points from the "Get Started Now" button to the "Launch" section. The "Launch" section includes an icon of three database instances with a plus sign, the text "Create DB Instances with just a few clicks.", and a "Learn more" link. The "Connect" section shows an icon of a computer monitor with a database connection symbol and the text "Once your DB instance is provisioned, you can use any standard SQL client application or utility to connect to your instance.", also with a "Learn more" link. The "Manage and Monitor" section shows an icon of a person with a lock and gear, and the text "You can easily add resources, modify configuration and monitor your DB Instances to meet your applications requirements.", also with a "Learn more" link.

Get Started Now

Amazon Relational Database Service

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale relational databases in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks, freeing you up to focus on your applications and business.

[Get Started Now](#)

[Getting Started Guide](#)

Launch

Create DB Instances with just a few clicks. You can customize database engine, instance size, storage, security, maintenance, and more.

[Learn more](#)

Connect

Once your DB instance is provisioned, you can use any standard SQL client application or utility to connect to your instance.

[Learn more](#)

Manage and Monitor

You can easily add resources, modify configuration and monitor your DB Instances to meet your applications requirements.

[Learn more](#)

Select Version of SQL

The screenshot shows the AWS RDS 'Launch DB instance' wizard. The top navigation bar includes the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and a bell icon. The left sidebar lists four steps: Step 1 'Select engine', Step 2 'Choose use case', Step 3 'Specify DB details', and Step 4 'Configure advanced settings'. The current step, 'Step 1 Select engine', is highlighted. The main content area is titled 'Select engine' and contains a sub-section 'Engine options' with six options arranged in a grid:

- Amazon Aurora
Amazon Aurora
- MySQL
- MariaDB
- PostgreSQL
- Oracle
ORACLE
- Microsoft SQL Server

A large callout text 'Select PostgreSQL' with an arrow points to the PostgreSQL option.

Multi-AZ Deployment

The screenshot shows the AWS RDS 'Launch DB instance' wizard. The top navigation bar includes the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and a bell icon. The left sidebar lists steps: 'Step 1 Select engine', 'Step 2 Choose use case' (which is currently selected), 'Step 3 Specify DB details', and 'Step 4 Configure advanced settings'. The main content area is titled 'Choose use case' and contains a question: 'Do you plan to use this database for production purposes?'. It offers two options: 'Production' (radio button is empty) and 'Dev/Test' (radio button is filled). Below the radio buttons, it says: 'Use Multi-AZ Deployment and Provisioned IOPS Storage as defaults for high availability and fast, consistent performance.' and 'This instance is intended for use outside of production or under the RDS Free Usage Tier.' At the bottom are 'Cancel', 'Previous', and 'Next' buttons. A large black arrow points from the text 'Choose Dev/Test' at the bottom left towards the 'Dev/Test' radio button.

Step 1
Select engine

Step 2
Choose use case

Step 3
Specify DB details

Step 4
Configure advanced settings

RDS > Launch DB instance

Choose use case

Use case

Do you plan to use this database for production purposes?

Use case

Production
Use Multi-AZ Deployment and Provisioned IOPS Storage as defaults for high availability and fast, consistent performance.

Dev/Test
This instance is intended for use outside of production or under the RDS Free Usage Tier.

Billing is based on RDS pricing.

Cancel Previous Next

Choose Dev/Test

Specify DB Details

Specify DB details

Instance specifications

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

DB engine

PostgreSQL

License model [Info](#)

postgresql-license



DB engine version [Info](#)

PostgreSQL 9.6.6-R1



Free tier

The Amazon RDS Free Tier provides a single db.t2.micro instance as well as up to 20 GB of storage, allowing new AWS customers to gain hands-on experience with Amazon RDS. Learn more about the RDS Free Tier and the instance restrictions [here](#).

Only enable options eligible for RDS Free Usage Tier [Info](#)

DB instance class [Info](#)

db.t2.micro — 1 vCPU, 1 GiB RAM



Check
Free tier

Select
db.t2.micro

Specify DB Details

No Multi-AZ

20GB
Storage

Add Identifier,
Username and
Password

The screenshot shows the 'Specify DB Details' step of the AWS RDS wizard. It has two main sections: 'Storage' and 'Settings'.

Storage Section:

- Multi-AZ deployment:** Info link, Create replica in different zone (radio button), No (selected).
- Storage type:** Info link, General Purpose (SSD) dropdown.
- Allocated storage:** Input field set to 20 GB.
- (Minimum: 20 GB, Maximum: 20 GB) Higher allocated storage may improve IOPS performance.

Settings Section:

- DB instance identifier:** Info link, mydbinstance input field. Specify a name that is unique for all DB instances owned by your AWS account in the current region.
- Master username:** Info link, input field. Specify an alphanumeric string that defines the login ID for the master user. Master Username must start with a letter. Must contain 1 to 63 alphanumeric characters.
- Master password:** Info link, input field. Master Password must be at least eight characters long, as in "mypassword". Can be any printable ASCII character except "/", "", or "@".
- Confirm password:** Info link, input field.

At the bottom right are 'Cancel', 'Previous', and 'Next' buttons.

Advanced Settings

Configure advanced settings

Network & Security

Virtual Private Cloud (VPC) [Info](#)

VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-f3b73d96)



Only VPCs with a corresponding DB subnet group are listed.

Subnet group [Info](#)

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default



Public accessibility [Info](#)

Yes

EC2 instances and devices outside of the VPC hosting the DB instance will connect to the DB instances. You must also select one or more VPC security groups that specify which EC2 instances and devices can connect to the DB instance.

No

DB instance will not have a public IP address assigned. No EC2 instance or devices outside of the VPC will be able to connect.

Availability zone [Info](#)

No preference



VPC security groups

Security groups have rules authorizing connections from all the EC2 instances and devices that need to access the DB instance.

Create new VPC security group

Choose existing VPC security groups

Check for
Public
accessibility

Advanced Settings

Add
Database
Name

→

Database options

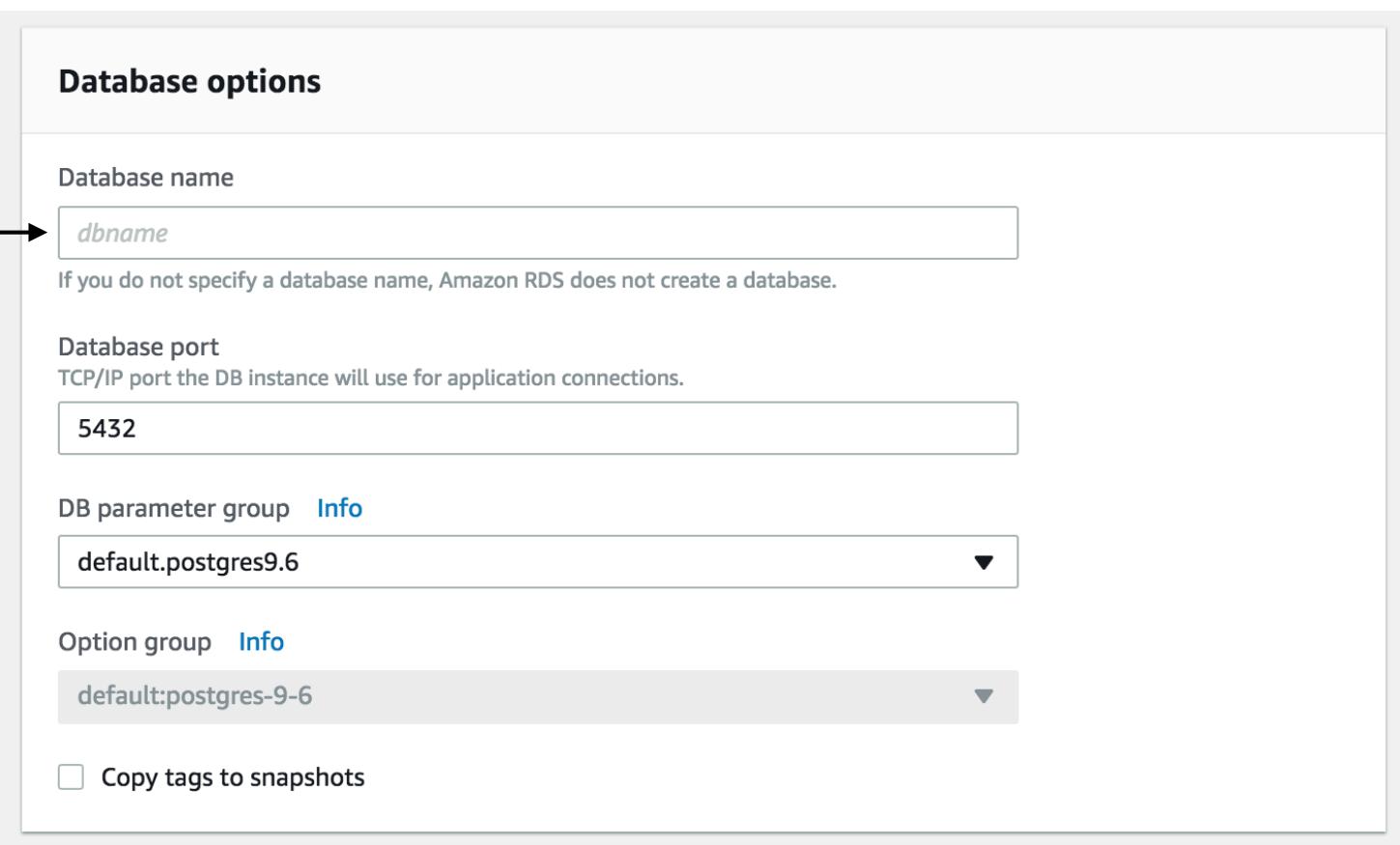
Database name If you do not specify a database name, Amazon RDS does not create a database.

Database port TCP/IP port the DB instance will use for application connections.

DB parameter group [Info](#)

Option group [Info](#)

Copy tags to snapshots



View Database Details

RDS > Instances > quantsprout

quantsprout

Instance Actions

Instance actions ▾

- Create read replica
- Promote read replica
- Take snapshot
- Restore to point in time
- Backtrack DB cluster
- Migrate latest snapshot
- Modify
- Stop
- Reboot
- Delete

Summary

Engine PostgreSQL 9.4.15	DB instance class Info db.t2.micro	DB instance status available	Pending maintenace none
-----------------------------	---	---------------------------------	----------------------------

CloudWatch (54)

Legend: [quantsprout](#)

[G](#) [Add instance to compare](#) [Monitoring ▾](#)

1 2 3 4 5 6

CPU Utilization (Percent)

0 0.5 1 1.5 2 2.5

04/14 22:30 04/14 23:00

DB Connections (Count)

0 25 50 75

04/14 22:30 04/14 23:00

Free Storage Space (MB)

0 5,000 10,000 15,000 20,000

04/14 22:30 04/14 23:00

View Database Details

Endpoint for Host

Connect		
Endpoint	Port	Publicly accessible
quantspout.cbh1q304oyvf.us-west-2.rds.amazonaws.com		
5432	Yes	
Security group rules (2)		
<input type="text"/> Filter security group rules		
Security group	Type	Rule
default (sg-96b950f2)	CIDR/IP - Inbound	0.0.0.0/0
default (sg-96b950f2)	CIDR/IP - Outbound	0.0.0.0/0

Security Group

Confirm Publicly Accessible

Configure Security

View VPC

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Recently visited services

RDS

All services

Compute

- EC2
- Lightsail
- Elastic Container Service
- Lambda
- Batch
- Elastic Beanstalk

Storage

- S3
- EFS
- Glacier
- Storage Gateway

Database

- RDS
- DynamoDB
- ElastiCache
- Amazon Redshift

Migration

- AWS Migration Hub
- Application Discovery Service
- Database Migration Service
- Server Migration Service
- Snowball

Networking & Content Delivery

- VPC
- CloudFront
- Route 53
- API Gateway
- Direct Connect

Management Tools

- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks
- Service Catalog
- Systems Manager
- Trusted Advisor
- Managed Services

Mobile Services

- Mobile Hub
- AWS AppSync
- Device Farm
- Mobile Analytics

AR & VR

- Amazon Sumerian

Application Integration

- Step Functions
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

Media Services

- Elastic Transcoder
- Kinesis Video Streams
- MediaConvert
- MediaLive
- MediaPackage
- MediaStore
- MediaTailor

Customer Engagement

- Amazon Connect
- Pinpoint
- Simple Email Service

Machine Learning

- Amazon SageMaker
- Amazon Comprehend
- AWS DeepLens
- Amazon Lex
- Machine Learning
- Amazon Polly
- Rekognition
- Amazon Transcribe
- Amazon Translate

Business Productivity

- Alexa for Business
- Amazon Chime
- WorkDocs
- WorkMail

Desktop & App Streaming

- WorkSpaces
- AppStream 2.0

Analytics

- Athena
- EMR
- CloudSearch

Internet of Things

- IoT Core

Helpful tips



Manage your costs

Get real-time billing alerts based on your cost and usage budgets. [Start now](#)



Create an organization

Use AWS Organizations for policy-based management of multiple AWS accounts. [Start now](#)

Explore AWS

Amazon Relational Database Service (RDS)

RDS manages and scales your database for you. RDS supports Aurora, MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server. [Learn more](#).

Real-Time Analytics with Amazon Kinesis

Stream and analyze real-time data, so you can get timely insights and react quickly. [Learn more](#).

Get Started with Containers on AWS

Amazon ECS helps you build and scale containers for any size application. [Learn more](#).

AWS Marketplace

Discover, procure, and deploy popular software products that run on AWS. [Learn more](#).

Have feedback?

[Submit feedback](#) to tell us about your experience with the AWS Management Console.

Update Security Groups

View
Security
Groups

The screenshot shows the AWS VPC Dashboard. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and a search icon. Below the navigation is a sidebar titled 'VPC Dashboard' with a 'Filter by VPC:' dropdown containing 'Select a VPC'. The main content area has a title 'Resources' with a back arrow and a refresh icon. It features two buttons: 'Start VPC Wizard' (blue) and 'Launch EC2 Instances' (gray). A note below says 'Note: Your Instances will launch in the US West (Oregon) region.' To the left of the main content is a 'Virtual Private Cloud' sidebar with the following items: Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, Endpoint Services, NAT Gateways, Peering Connections, Security, Network ACLs, and Security Groups. A large black arrow points from the 'View Security Groups' text down to the 'Security Groups' item in the sidebar. The main content area also displays a summary of resources: 1 VPC, 0 Egress-only Internet Gateways, 1 Route Table, 0 Elastic IPs, 0 Endpoints, 3 Security Groups, 0 VPN Connections, 0 Customer Gateways, 1 Internet Gateway, 3 Subnets, 1 Network ACL, 0 VPC Peering Connections, 0 Nat Gateways, 0 Running Instances, 0 Virtual Private Gateways, and 1 DHCP Options Set. Below this is a section titled 'VPN Connections' with a note about using Amazon VPC for isolated resources and connecting to a datacenter via IPsec VPN, followed by a 'Create VPN Connection' button.

VPC Dashboard

Filter by VPC:

Select a VPC

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

Endpoint Services

NAT Gateways

Peering Connections

Security

Network ACLs

Security Groups

Resources

Start VPC Wizard

Launch EC2 Instances

Note: Your Instances will launch in the US West (Oregon) region.

You are using the following Amazon VPC resources in the US West (Oregon) region:

1 VPC	1 Internet Gateway
0 Egress-only Internet Gateways	3 Subnets
1 Route Table	1 Network ACL
0 Elastic IPs	0 VPC Peering Connections
0 Endpoints	0 Nat Gateways
3 Security Groups	0 Running Instances
0 VPN Connections	0 Virtual Private Gateways
0 Customer Gateways	1 DHCP Options Set

VPN Connections

Amazon VPC enables you to use your own isolated resources within the AWS cloud, and then connect those resources directly to your own datacenter using industry-standard encrypted IPsec VPN connections.

Create VPN Connection

Update Inbound & Outbound Rules

The screenshot shows the AWS VPC Dashboard with the 'Services' and 'Resource Groups' navigation bar at the top. On the left, there's a sidebar with various VPC-related options like 'Virtual Private Cloud', 'Your VPCs', 'Subnets', etc. The main area is titled 'Create Security Group' and 'Security Group Actions'. It shows a list of security groups with columns for 'Name tag', 'Group ID', 'Group Name', and 'VPC'. Two groups are listed: 'sg-3c55765b' and 'sg-8fba53eb'. The second group is selected. Below the list, there are tabs for 'Summary', 'Inbound Rules' (which is highlighted in blue), 'Outbound Rules', and 'Tags'. Under the 'Inbound Rules' tab, there's a table with columns 'Type', 'Protocol', 'Port Range', 'Source', and 'Description'. One rule is shown: 'ALL Traffic' for Protocol, 'ALL' for Port Range, and '0.0.0.0/0' for Source. The 'Description' column is empty.

Edit Inbound & Outbound Rules

All Traffic with
Source 0.0.0.0/0

Database Administration

- Create User Credentials
- Grant and Revoke privileges to tables
- Organize database into schemas
- Update database parameters on AWS
 - Increase/decrease instance size
 - Increase memory size (can't decrease)
 - Delete database and save image of data

Create User

- Create new users for your database:
- Syntax:
CREATE USER <username> WITH PASSWORD
<password>;
- Example:
CREATE USER craig WITH PASSWORD 'test';

Grant / Revoke Privileges

Users have no privileges by default

- Syntax:

GRANT ALL ON <table> TO <user>;

GRANT SELECT on <table> TO <user>;

REVOKE SELECT on <table> FROM <user>;

- Example:

GRANT ALL ON artist TO craig;

GRANT SELECT ON track TO public;

REVOKE INSERT ON track FROM public;

Create and Grant Schema

- Syntax:

```
CREATE SCHEMA [schema_name];
```

```
GRANT USAGE ON SCHEMA TO [user];
```

- Example:

```
CREATE SCHEMA craig_schema;
```

```
GRANT USAGE ON craig_schema to dylan_moore;
```

Search Path

- Privileges can be set for each schema
- Reference tables in a schema:
`<schema>.<table>`
public.track
- Set default schema:
`set search_path = <schema1>, <schema2>;`
`set search_path = craig, public;`
- Everyone has a schema based on the name you used to register (all lower case with spaces replaced with underscore)

Set Your Search Path to Your Schema

Table Options

- Unique – prevents duplicate values
- Not Null – prevents insertion of records with null values for field
- Primary Key – adds unique and non-null constraint
- Foreign Key – adds constraint for field to ensure values exist in corresponding reference table
- Check – applies logical statement as constraint for field
- Serial data type – automatically increments integers for new records

Create Table

- Syntax:

```
CREATE TABLE IF NOT EXISTS <table name> (  
    <field name1> <data type1>,  
    <field name2> <data type2>);
```

- Example:

```
CREATE TABLE IF NOT EXISTS web_log(  
    trans_id SERIAL PRIMARY KEY,  
    user_name VARCHAR(255),  
    transaction VARCHAR(255),  
    ts TIMESTAMP)
```

Import Data

- Load CSV data using WBImport command
- Full documentation at www.sql-workbench.net/manual/command-import.html
- Requires parameters for data loading:
 - Filename and path
 - File type
 - Delimiter
 - Destination table
 - Mapping of data to table fields
 - Date format
 - Batch size (impacts import speed)

Code for Importing Data

```
WbImport
```

```
-file='/Users/Craig_Sakuma/Desktop/yelp.csv'
```

```
-type=text
```

```
-table=public.yelp
```

```
-delimiter=|
```

```
-timestampFormat='yyyy-MM-dd'
```

```
-filecolumns=business_id, review_date,review_id, stars,  
review_text, review_type, user_id, cool, funny, useful
```

```
-batchsize=1000;
```

Amazon Pop-Up Loft

- 1446 Market Street, San Francisco, CA
- aws.amazon.com/start-ups/loft/sf-loft/
- Free help desk and working space
- Host training events

Exercise #4

- Connect to your new AWS databases
- Create a table in your database for the yelp.csv data
- Upload data from the yelp.csv file to the table you've created



QuantSprout

yelp.com/biz/quantsprout-san-francisco