

# Using Databases

2020-04-23



# Contents

<b>1</b>	<b>5</b>
<b>2 Introduction</b>	<b>7</b>
<b>3 Connecting to a Database</b>	<b>9</b>
3.1 Prerequisite Steps . . . . .	9
3.2 Connecting and Querying Data . . . . .	10



# Chapter 1



## Chapter 2

### Introduction





## Chapter 3

# Connecting to a Database

### 3.1 Prerequisite Steps

The manual takes uses the ODBC driver/ connection string method to connect to the Stats repository. These drivers are proprietary to the SQL service you are using, so they need to be installed for each server type you wish to connect to. Connection strings refers to the passwords and information needed to connect to the database, as well as the format that that information needs to be specified in. Connection strings are also specific to the database you are connecting to. For this example the driver and connection string can be found at the link below.

Microsoft SQL Server Drivers can be found at: <https://docs.microsoft.com/en-us/sql/connect/odbc/microsoft-odbc-driver-for-sql-server?view=sql-server-ver15>

After downloading, the drivers can be installed by running the following commands in a terminal.

#### 3.1.1 Linux - Ubuntu

```
sudo su
curl https://packages.microsoft.com/keys/microsoft.asc | apt-key add -

#Download appropriate package for the OS version
#Choose only ONE of the following, corresponding to your OS version

#Ubuntu 16.04
curl https://packages.microsoft.com/config/ubuntu/16.04/prod.list > /etc/apt/sources.list.d/mssql

#Ubuntu 18.04
curl https://packages.microsoft.com/config/ubuntu/18.04/prod.list > /etc/apt/sources.list.d/mssql
```

```
#Ubuntu 19.10
curl https://packages.microsoft.com/config/ubuntu/19.10/prod.list > /etc/apt/sources.list.d/mssql-release.list

exit
sudo apt-get update
sudo ACCEPT_EULA=Y apt-get install msodbcsql17
# optional: for bcp and sqlcmd
sudo ACCEPT_EULA=Y apt-get install mssql-tools
echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bash_profile
echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bashrc
source ~/.bashrc
# optional: for unixODBC development headers
sudo apt-get install unixodbc-dev
```

### 3.1.2 Mac OS

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
brew tap microsoft/mssql-release https://github.com/Microsoft/homebrew-mssql-release
brew update
HOMEBREW_NO_ENV_FILTERING=1 ACCEPT_EULA=Y brew install msodbcsql17 mssql-tools
```

### 3.1.3 Windows

download and open msodbcsql.msi file from: <https://docs.microsoft.com/en-us/sql/connect/odbc/download-odbc-driver-for-sql-server?view=sql-server-ver15#download-for-windows>

### 3.1.4 Other Connection Options

Once drivers are installed, you can set a dsn (data source name) for the connection to the database using the ODBC/DSN manager for your system. This is OS-specific and while it will allow a single user to access the database more easily, it must be configured for every user who wishes to connect to the data source. Therefore, to more easily share code and to not have to mess with their behind-the-scenes settings, I use the connection string method from here further.

## 3.2 Connecting and Querying Data

### 3.2.1 Julia

```
using DataFrames, DataFramesMeta, ODBC, Lazy
dsn=ODBC.DSN("Driver={ODBC Driver 17 for SQL Server};Address=24.205.251.117;Database=master")
df=ODBC.query(dsn,"select * from events ")
ODBC.disconnect!(dsn)
```

### 3.2.2 SAS

```
/*from proc sql directly*/
/*this is pretty convoluted*/
proc sql;
connect to odbc as conn
required="Driver={ODBC Driver 17 for SQL Server};Address=24.205.251.117;Database=ntsb;UID=ntsb;PW

create table event as
select * from connection to conn
(select * from events) ;

disconnect from conn;
quit;

/*by using a libname*/
libname conn2 odbc
required ="Driver={ODBC Driver 17 for SQL Server};Address=24.205.251.117;Database=ntsb;UID=ntsb;PW
proc sql;
create table event as
select * from conn2.events ;
quit;
```

### 3.2.3 R

```
library(RODBC)
# tried the tidyverse ODBC package, but it kept giving me errors
library(tidyverse)
conn=odbcDriverConnect(connection="Driver={ODBC Driver 17 for SQL Server};Address=24.205.251.117;
dat =conn%>% sqlQuery("select TOP 50 * from events")
close(conn)
```

### 3.2.4 Python

```
import pyodbc
conn = pyodbc.connect("Driver={ODBC Driver 17 for SQL Server};Address=24.205.251.117;Database=ntsb;UID=ntsb;PW
cursor = conn.cursor()
cursor.execute("select TOP 20 * from events" )
dat =cursor.fetchall()
conn.close()
```

### 3.2.5 JSL

```
open database(
    "Driver={ODBC Driver 17 for SQL Server};Address=24.205.251.117;Database=ntsb;UID=ntsb;PW
```

```
      "select top 20 from events"
    )

<!--chapter:end:02-connections.Rmd-->
```