



Microsoft: DAT209x Programming in R for Data Science



Bookmarks



Bookmark

- ▶ 0. Start Here
- ▶ 1. Introduction
- ▶ 2. Functions and Data Structures
- ▶ 3. Loops and Flow Control
- ▶ 4. Working with Vectors and Matrices
- ▶ 5. Reading in Data
- ▶ 6. Writing Data to Text Files
- ▼ 7. Reading Data from SQL Databases

7. Reading Data from SQL Databases > Lab > Lab

For this lab, please use the following SQL Server.

- **Server name:** msedxeus.database.windows.net
- **Database name:** DAT209x01
- **User name:** RLogin
- **Password:** P@ssw0rd

Question 1

(1/1 point)

Let's first establish a connection to the SQL server. Drag and drop the texts to the objects to complete the code.

[Lecture](#)[Knowledge Checks](#)[Quiz](#)[Lab](#)[Lab](#)[Mid-course Survey](#)[Survey](#)

```
library(RODBC)
connStr <- paste("Server=msedxeus.database.windows.net",
  "Database=", DAT209x01,
  "uid=", RLogin,
  "pwd=", P@ssw0rd,
  "Driver={SQL Server}",
  sep=";"
)
conn <- odbcDriverConnect(connStr)
```

Question 2

(1/1 point)

Construct a data frame based on the **salesFact** table from the SQL server above, for the year of **2014**, grouped by each **ProductID**. Each line should the **total** of both **Revenue** and **units**.

You have the existing code below:

```
my.data.frame <- my_func(conn,  
                          "SELECT SUM(Revenue), SUM(Units), ProductID  
                          FROM bi.salesFact  
                          WHERE Date > '2013-12-31' AND Date < '2015-01-01'  
                          GROUP BY ProductID"  
)
```

Which function should you use to replace the my_func placeholder?

☐ sqlTables

☐ sqlFetch

☒ sqlQuery ✓

☐ sqlColumns

EXPLANATION

To submit a SQL query to an ODBC database such as SQL Server, you should use the `sqlQuery()` function.

Question 3

(1/1 point)

Now, name the column of the data frame accordingly. Given the code from before, which option should you use to name the columns of the data frame?

- ☐ `colNames(my.data.frame)<-c("SUM(Revenue)","SUM(Units)","ProductID")`
- ☒ `names(my.data.frame)<-c("SUM(Revenue)","SUM(Units)","ProductID")` ✓
- ☐ `colNames(my.data.frame)<-c("SUM(Units)","SUM(Revenue)","ProductID")`
- ☐ `names(my.data.frame)<-c("SUM(Units)","SUM(Revenue)","ProductID")`

EXPLANATION

You can use the following command to perform the task:

```
names(my.data.frame)<-c("SUM(Revenue)","SUM(Units)","ProductID")
```

Question 4

(1/1 point)

Now that you have constructed the data frame, find the **ProductIDs** with the top 5 most number of units using the `order()` function.

Which of the **productIDs** have the most number of units?

☒ 465 506 449 438 487 ✓

☐ 465 506 407 449 487

☐ 727 728 2407 1014 1812

☐ 40 92 130 155 187

EXPLANATION

You can use the following command to solve the problem:

```
my.data.frame$ProductID[order(my.data.frame$"SUM(Units)",decreasing=TRUE)][1:5]
```

Question 5

(1/1 point)

Now find the **ProductIDs** with the top 5 highest revenue using the `order()` function.

Which of the **productIDs** have the highest revenue? Are they the same with those that have the most number of units?

☐ 465 506 449 438 487

☒ 465 506 407 449 487 ✓

☐ 727 728 2407 1014 1812

☐ 40 92 130 155 187

EXPLANATION

You can use the following command to solve the problem:

```
my.data.frame$ProductID[order(my.data.frame$"SUM(Revenue)",decreasing=TRUE)][1:5]
```

Notice that only the top 2 products have the highest revenue and the most number of units.

© All Rights Reserved



© edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

