

# R Notebook

## Install and Load RMySQL

```
#install.packages("RMySQL")  
library(RMySQL)
```

## Loading required package: DBI

## Connecting to MySQL:

Once the RMySQL library is installed create a database connection object.

```
mydb = dbConnect(MySQL(), user='stressdbuser',  
                  password='#October218',  
                  dbname='stressdb',  
                  host='35.194.173.89')
```

## Listing Tables and Fields:

Now that a connection has been made we list the tables and fields in the database we connected to.

```
dbListTables(mydb)
```

```
## [1] "applications_crashes"  
## [2] "applications_foreground"  
## [3] "applications_history"  
## [4] "applications_notifications"  
## [5] "aware_debug"  
## [6] "aware_device"  
## [7] "aware_log"  
## [8] "aware_observer"  
## [9] "aware_studies"  
## [10] "battery"  
## [11] "battery_charges"  
## [12] "battery_discharges"  
## [13] "calls"  
## [14] "cdma"  
## [15] "esms"  
## [16] "gsm"  
## [17] "gsm_neighbor"  
## [18] "messages"  
## [19] "mqtt_history"  
## [20] "mqtt_messages"  
## [21] "mqtt_subscriptions"  
## [22] "plugin_device_usage"  
## [23] "plugin_google_activity_recognition"  
## [24] "plugin_ios_activity_recognition"  
## [25] "plugin_studentlife_audio_android"  
## [26] "processor"  
## [27] "push_notification_device_tokens"
```

```
## [28] "screen"
## [29] "telephony"
```

This will return a list of the tables in our connection.

```
dbListFields(mydb, 'screen')
```

```
## [1] "_id"          "timestamp"    "device_id"    "screen_status"
```

This will return a list of the fields in some\_table.

## Running Queries:

Queries can be run using the dbSendQuery function. dbSendQuery(mydb, 'drop table if exists some\_table, some\_other\_table')

- Making tables: We can create tables in the database using R dataframes.

```
dbWriteTable(mydb, name='table_name', value=data.frame.name)
```

## Retrieving data from MySQL:

To retrieve data from the database we need to save a results set object.

```
rs = dbSendQuery(mydb, "select * from screen")
screendata = fetch(rs, n=-1)
screendata<-transform(screendata, screen_status=as.factor(screen_status))
head(screendata, 10)
```

##	X_id	timestamp	device_id	screen_status
## 1	1	1.503388e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	0
## 2	2	1.503388e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	0
## 3	3	1.503388e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	1
## 4	4	1.503388e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	1
## 5	5	1.503389e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	0
## 6	6	1.503389e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	2
## 7	7	1.503389e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	0
## 8	8	1.503389e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	2
## 9	9	1.503390e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	1
## 10	10	1.503390e+12	9d794d64-e4b0-4564-9321-d56ce6a52f55	1

## Plot Chart

```
plot(screendata$screen_status)
```

