

Synapse Demo - Table Uploads

Aryton Tediario

2021-03-17

This R markdown is used for showcasing basic interaction with Synapse using RStudio. It will cover on how to utilize our Synapse R client to create table entity, which is a SQL query-able tables in Synapse interface.

Full Documentation: <https://r-docs.synapse.org/articles/tables.html>

Import Libraries

```
library(data.table)
library(tidyverse)
library(synapser)
library(synapserutils)
synLogin()
```

```
## Welcome, Aryton Tediario!
```

```
## NULL
```

Global Variables

```
SYNAPSE_PROJECT_ID <- "syn20816722" # where you will be saving all your files
FILEPATH_1 <- file.path("data", "sample_2.tsv")
FILEPATH_2 <- file.path("data", "sample_3.tsv")
TABLE_RESULTS <- "Synapse Demo Table"
```

```
data <- fread(FILEPATH_1) %>% tibble::as_tibble()
data
```

```
## # A tibble: 5 x 4
##   user      createdOn      x      y
##   <chr>      <date>      <dbl> <dbl>
## 1 2ujCq00TaL 2020-01-26 -0.0925 -2.06
## 2 ZjFIjuDMRu 2020-07-09  0.638  -1.06
## 3 OW711K02MZ 2020-10-17  0.202  -1.16
## 4 gyh0Xpo5Ex 2020-11-09 -0.0699  1.65
## 5 crmp5Sbvvy2 2020-11-26 -0.713  0.449
```

QnD storing table

A quick way to store table is to use available wrapper from Synapse R Client. It will infer data type of each column in the schema.

```
#' wrapper create table function
table <- synapser::synBuildTable(
  TABLE_RESULTS, SYNAPSE_PROJECT_ID, data
)

#' store to synapse
table <- synStore(table)
```

```
## Uploading [-----]0.00% 0.0bytes/351.0bytes file2185432373ae Uploading [#####]
```

Custom Columns Types

A more manual way is to define each of the column schema individually.

```
#' delete table
synapser::synDelete(table$tableId)
```

```
## NULL
```

```
#' set custom columns
cols <- list(
  Column(name = "user", columnType = "STRING", maximumSize = 20),
  Column(name = "createdOn", columnType = "DATE"),
  Column(name = "x", columnType = "DOUBLE"),
  Column(name = "y", columnType = "DOUBLE"))

# set schema entity
schema <- Schema(name = TABLE_RESULTS,
  columns = cols,
  parent = SYNAPSE_PROJECT_ID)

#' recreate table entity
table <- Table(schema, data)
table <- synStore(table)
```

```
## Uploading [-----]0.00% 0.0bytes/351.0bytes file218558de1f73 Uploading [#####]
```

Appending new rows to Synapse Tables

Appending new data can be achieved by storing a new data (given matching columns) to the previous table id.

```
#' what is the table Id?
tableId_in_synapse <- table$tableId
```

```
#' read data for appending
append_data <- fread(FILEPATH_2) %>%
  tibble::as_tibble()

#' append to table
synStore(Table(tableId_in_synapse, append_data))
```

```
## Uploading [-----]0.00% 0.0bytes/346.0bytes file21856fc509a3 Uploading [#####]
```

```
## <synapseclient.table.CsvFileTable object at 0x1239315c0>
```