

Connect databses in Julia

Masaya Kameyama

2021-07-29

```
versioninfo()
```

```
Julia Version 1.6.1
Commit 6aaedecc44 (2021-04-23 05:59 UTC)
Platform Info:
  OS: macOS (x86_64-apple-darwin18.7.0)
  CPU: Intel(R) Core(TM) i7-8557U CPU @ 1.70GHz
  WORD_SIZE: 64
  LIBM: libopenlibm
  LLVM: libLLVM-11.0.1 (ORCJIT, skylake)
Environment:
  JULIA_NUM_THREADS = 8
```

```
using GBQ
```

```
# use .env
using DotEnv
DotEnv.config(path = "/Users/kameyama/.env")
aurora="dbname='${ENV["WRITE_RDB_DATABASE"]}' host='${ENV["WRITE_RDB_HOST"]}' user='${ENV["WRITE_RDB_USER"]}'
redshift="dbname='${ENV["DWH_DATABASE"]}' host='${ENV["DWH_HOST"]}' user='${ENV["DWH_USER"]}'
project_name=ENV["BQ_PROJECT_NAME"]
dataset_name=ENV["BQ_DATASET_NAME"]
;
```

```
using LibPQ
using DataFrames
# next two packages conflict with each other at the macro @select
using DataFramesMeta
# using Queryverse #packages for dataframes
```

```
using Format
```

```
function sql(query,conn_str)
    conn = LibPQ.Connection(conn_str; options=Dict{String, String}())
    result =execute(conn,query)
    df = DataFrame(result)
    close(conn)
    return df
end
```

sql (generic function with 1 method)

```
function sql_time(query1,query2)
    print("Aurora: ")
    @time sql(query1,aurora)
    print("Redshift: ")
    @time sql(query1,redshift)
    print("Bigquery: ")
    @time gbq_query(query2)
end
```

sql_time (generic function with 1 method)

```
query="select * from {}companies"
query1=format(query,"")
query2=format(query,project_name*"."*dataset_name*".")
```

"select * from prod-tameike-219208.smart.companies"

```
sql_time(query1,query2);
```

Aurora: 0.485374 seconds (202.23 k allocations: 7.152 MiB)
Redshift: 0.265902 seconds (211.10 k allocations: 7.288 MiB)
Bigquery: 2.751780 seconds (142.51 k allocations: 14.153 MiB)