

Querying ICES database

Introduction

This page is a result from a question from one of the VECTORS members.

The method provided creates a list of averaged temperatures, averaged salinity for each icessquare, for each year, for each month and for each depth class.

Datasource:

- postgresx03.xtr.infra.deltares.nl (database name = ICES, username = dbices, password = vectors) (!!privileges are read-only!!)

Necessary modules:

For Python psycopg2 (<http://www.stickpeople.com/projects/python/win-psycopg/psycopg2-2.4.4.win32-py2.7-pg9.1.2-release.exe>)

For R the library RPostgreSQL is used.

Below the code and the scripts of the 2 possibilities to query the database:

- [#The snippet for Python](#)

- [#The snippet for R](#)

The snippet for Python

Please note that if you are using the code, the indents are very important in Python.

```
# Description: Use of psycopg2 module to query database ICES
# -----
# Created by Gerrit Hendriksen (gerrit.hendriksen@deltares.nl)
# v1.0 created on 21-05-2012 (ddmmyyyy)
#
# Description: retrieving data per icessquare form the ICES database
# -----

# import modules
import psycopg2

# create connection to ices database
conn = psycopg2.connect("dbname=ICES host=postgresx03.infra.xtr.deltares.nl user=dbices password=vectors")

# create a cursor object called cur
cur = conn.cursor()

# construct a query string
strSql = """
SELECT i.statsq, year, month, avg(temperature) as avg_T, avg(salinity) as avg_Sal,sdepth_bin
FROM ocean o,
      icessquares i
WHERE st_within(the_point,the_geom) AND
      sea_region LIKE 'North Sea%'
      AND year > 1980
      AND temperature IS NOT NULL and salinity IS NOT NULL
GROUP BY i.statsq, year, month, sdepth_bin
ORDER BY statsq,year, month
"""

# execute the query
cur.execute(strSql)

# store the result of the query into Tuple c
c = cur.fetchall()
print (c)

# closes the connection
cur.close()
conn.close()

# do something with the query result
# -----
```

The entire code can be downloaded as [Python Code](#).

The snippet for R

```
## Opens connection to ICES oceanographic database (online PostGreSQL database, copy of original <2010)
## hosted by Deltares, The Netherlands
## Fetches table with selected entries. In this case table nox (nitrate+nitrite average per ICES rectangle for
the North Sea
##

library(RPostgreSQL)

## loads the PostgreSQL driver
drv <- dbDriver("PostgreSQL")

## Open a connection
con <- dbConnect(drv, dbname="ICES", host="postgresx03.infra.xtr.deltares.nl", user="dbices", password="
vectors")

## Submits a statement
rs <- dbSendQuery(con, "SELECT i.statsq, year, month, avg(temperature) as avg_T, avg(salinity) as avg_Sal,
sdepth_bin
FROM ocean o,
      icesquares i
WHERE st_within(the_point,the_geom) AND
      sea_region LIKE 'North Sea%'
      AND year > 1980
      AND temperature IS NOT NULL and salinity IS NOT NULL
GROUP BY i.statsq, year, month, sdepth_bin
ORDER BY statsq,year, month")

## fetch all elements from the resultSet into a data.frame
df <- fetch(rs, n = -1)

## Check number of records
dim(df)

## write comma-separated data to file
write.table(df, file = "Filename.txt", sep = ",")

## Closes the connection
dbDisconnect(con)

## Frees all the resources on the driver
dbUnloadDriver(drv)
```

Please try out and adjust to your needs and of course share it with the OpenEarth community. You are challenged to share your modifications.