**Session 10: Modelling with CANDY – export & import relevant tables and create a new simulation run**

**Aims:**

1. Create a new Database with an SqLite Dump
2. Basics: import tables via
   1. DBeaver
   2. R Studio
3. Set up a new simulation run, including the import of relevant input data
4. Along the way: understand upcoming errors and mistakes

**Important to note:**

While it is easier to process data via the graphical user interfaces, this will eventually backfire when you have to process many simulation-runs and tons of data! This is why we strongly recommend using R software as soon as possible. If you are not familiar with R, we are here to help and teach you, plus there are easy to understand pre-made scripts for nearly all tasks! We wish you lots of fun and enjoy modelling!

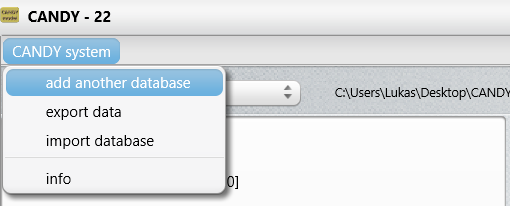
**CREATING A NEW DATABASE**

Now that Eric provided us with a good database to work with, let us revisit the steps for creating a new database based on a dump file.

FIRST we will create a NEW EMPTY database:

**OPEN THE CANDY EXE**

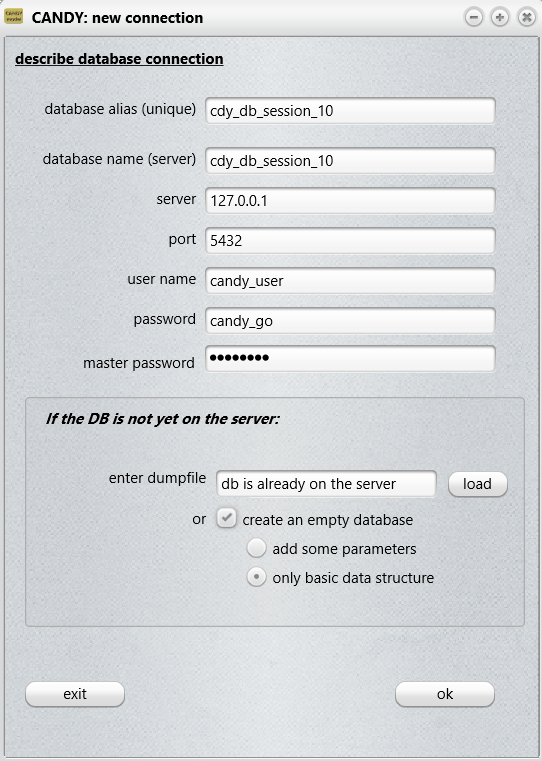
**In the top left corner, click on [CANDY system] -> [add another database]**

****

A new window will pop up!

**FILL OUT database alias and database name with:**

* ***cdy\_db\_session\_10***

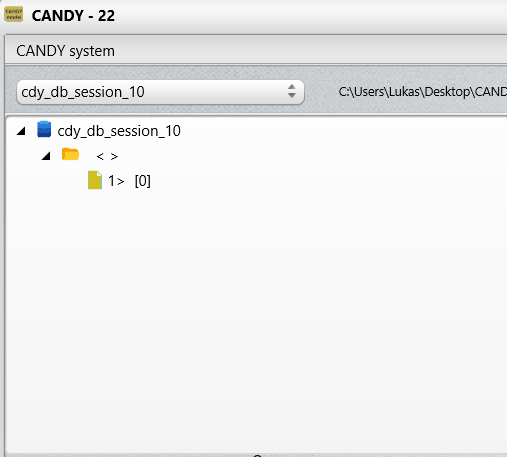


Since we want to IMPORT an EXISITING database in the form of a dump, we create the new database ‘cdy\_db\_session\_10’ with ‘only basic data structure’

**TICK the boxes [create an empty database] -> [only basic data structure]**

**CLICK [OK] at the bottom right**

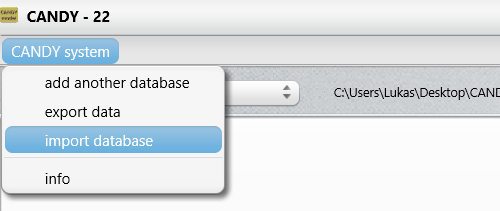
Now, the new EMPTY database should be pre-selected. CANDY should look like this:



Now we want to fill this database with data! For that, we can load in a SQLite dump.

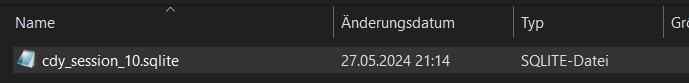
FOLDER xxx

AGAIN, i**n the top left corner, click on [CANDY system] -> [import database]**



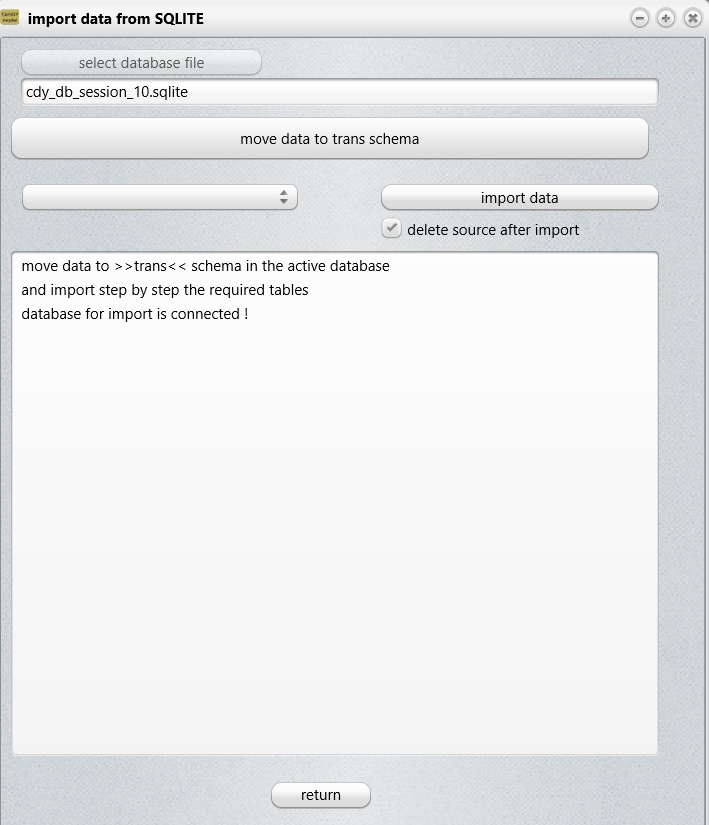
A new window will pop up, asking you to browse your folders where to find the SQLite dump.

OPEN THE FOLDER xxx AND SELECT THE FILE ‘cdy\_session\_10.sqlite’



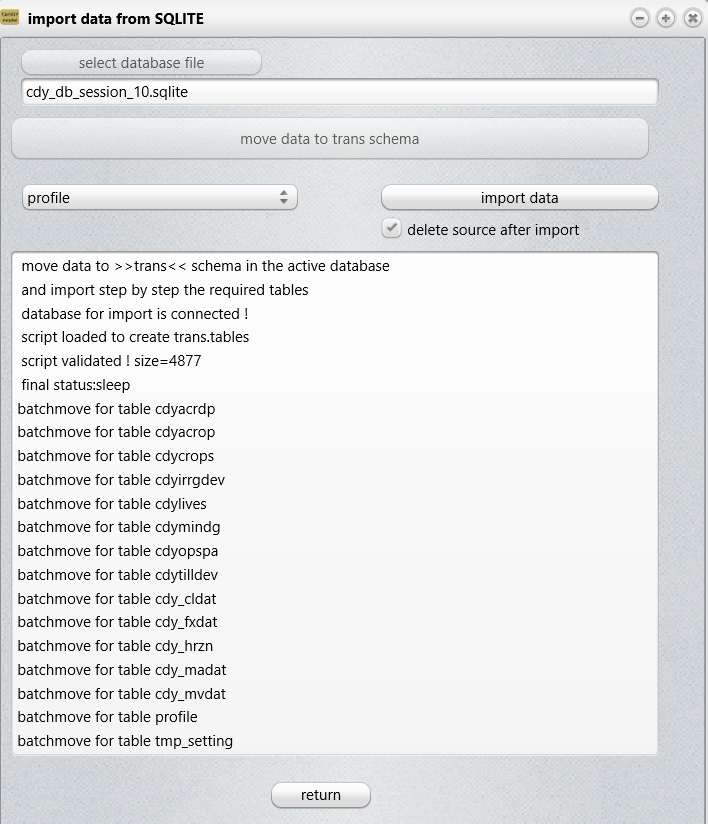
When you selected the sqlite file you should be automatically directed to the next window:

**CLICK ON [move data to trans schema]**



The related .sql files that come with the .sqlite dump are loaded into the import window.

It should look like this:



Let us start to import the tables!

The CURRENT table to IMPORT (1) is listed on the left

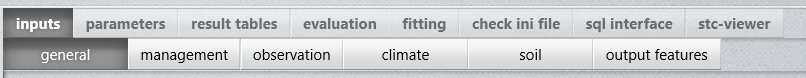
**To import this table CLICK [import data] until there are no more tables left to import**

**ALL IMPORTED? Close the window and close CANDY**

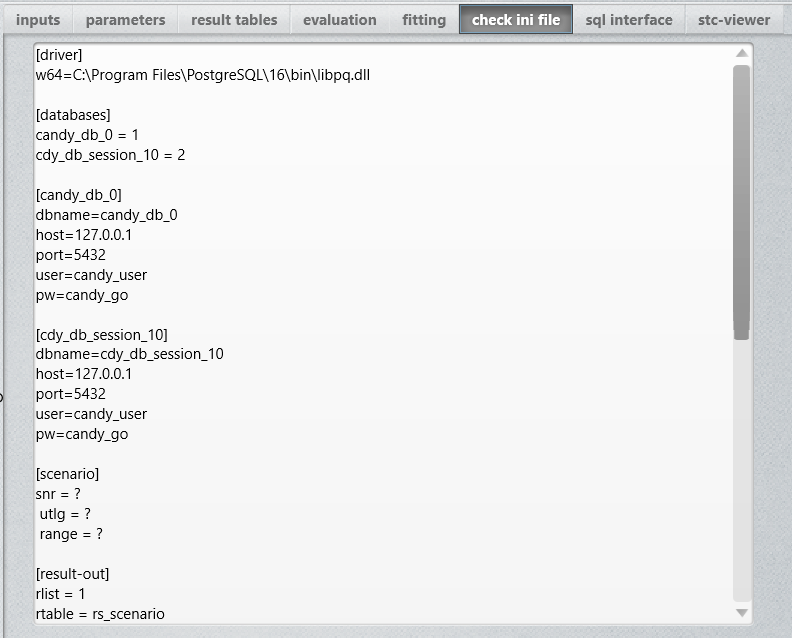
The good thing is, CANDY automatically updated our initialization file (pg4candy.ini)!

Lets have a look at it and confirm, that our new database is indeed present!

**OPEN CANDY -> click [check ini file]**



The databases are listed atop, with detailed information listed below.   
The window should look like this:



DATABASE ‘cdy\_db\_session\_10’ PRESENT?

GOOD, now you successfully CREATED your own new database AND filled it with all the information needed 😊.

**CLOSE CANDY FOR NOW**

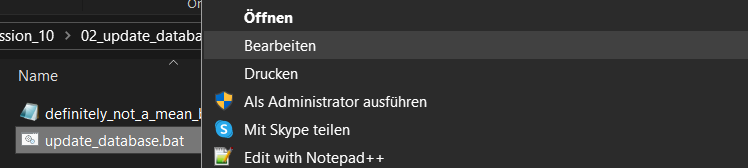
BEFORE we continue, there is one more thing we need to do! Just to add some minor details to our pristine, shiny newly created database, I prepared a script that we can execute!

**CAREFUL NOW**

IN FOLDER ***02\_update\_database*** there are TWO files. One of them is a **.bat file (.sh for MAC)!**

**!!! DO NOT DOUBLE CLICK THIS FILE !!! (yet)**

**RIGHT CLICK on the file -> edit**



(FOR WINDOWS 11 USERS, after right clicking the file, got to the very bottom and select ‘MORE’ -> edit)

For mac users, this applies to .sh files.

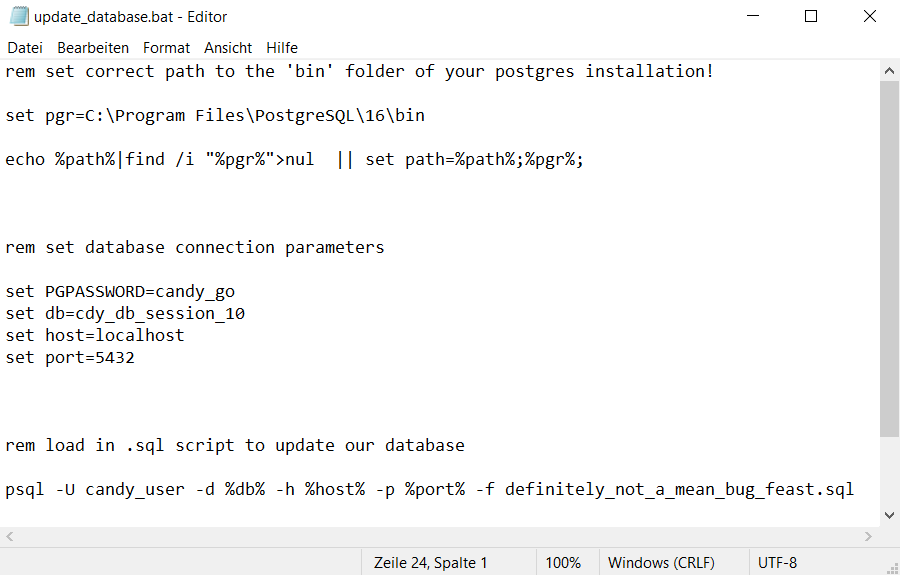
**The bat file should open.**

**WINDOWS USERS ONLY:**

**Remember that very important path, that you will ALWAYS need? The path to the libpq.dll file of your postgres installation!**

**COPY THIS PATH SOMEWHERE, WHERE YOU CAN EASILY ACCESS AND COPY IT. YOU WILL NEED IT A LOT**

Okay, back to the file. **Change the path to your libpq.dll**:



After you done it, **SAVE THE FILE AND CLOSE IT.**

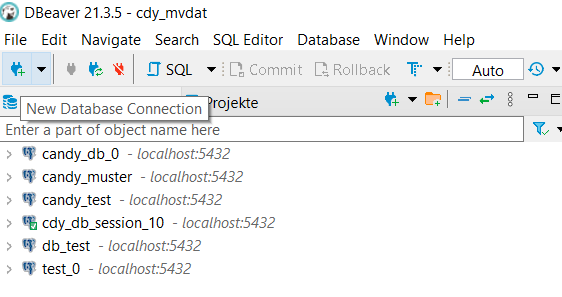
**DOUBLE CLICK the .bat | .sh**

**DONE? Beautiful, now that we are finally prepared, let us start with some simulations 😊!**

**SET UP A SIMULATION SCENARIO**

Okay, first lets look at our new database! I hope you remember how to connect a new database to DBeaver?

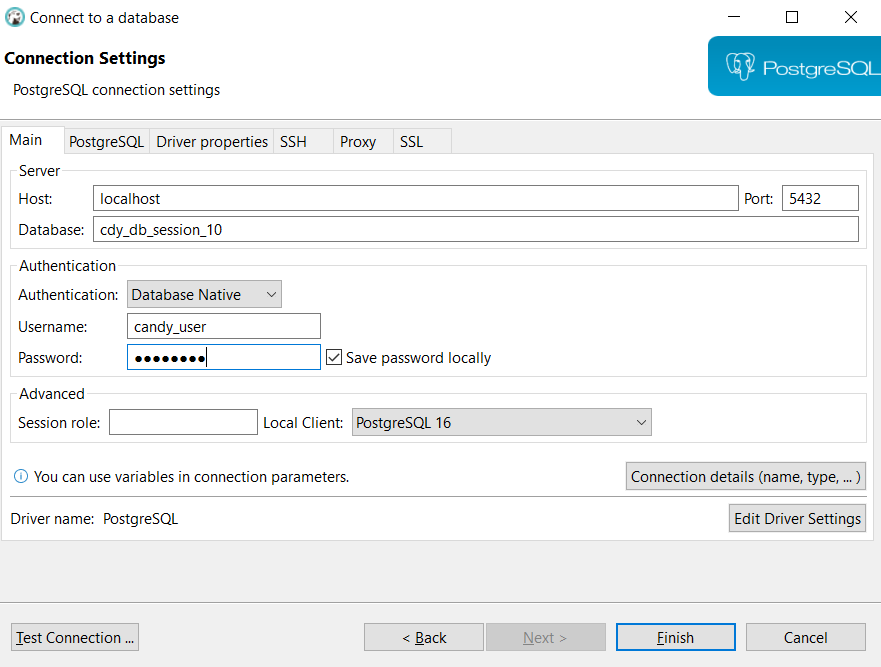
**OPEN DBeaver and connect the new database to the server!**



Database: cdy\_db\_session\_10

Username: candy\_user

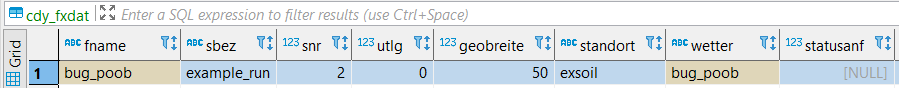
Password: candy\_go



SUCCESSFULLY CONNECTED?

Okay then, let us briefly look at our new database.

**OPEN TABLE cdy\_fxdat and look at the ….. WHAT THE…? That is NOT the filename that we previously wanted?!**



Awww snap seems like there were some really MEAN BUGS going absolutely HAM on our new database!!

Even WORSE. Apparently (and COMPLETELY by accident…) they partially or completely ate ALL of the tables that are ESSENTIAL for our simulation run!

* cdy\_fxdat – our general informations
* profile and cdy\_hrzn – our soil information
* cdy\_cldat – our weather data
* cdy\_madat – our agricultural management schedule
* cdy\_mvdat – our measured observations

AND

* our result record lists …
  + cdy\_rslt\_lst
  + cdy\_rslt\_sel
  + cdy\_prop\_lst
  + cdy\_prop\_sel

**Well, I think we have no choice but to clean up the mess AND learn how to import the most important data necessary for a simulation run, …what a coincidence!**

**CLEAN UP THE BUG-FEAST MESS**

**general information – cdy\_fxdat**

Luckily, there is not much to do in order to clean up the bug mess here! We will do required changes in the table in DBeaver.

**Simply overwrite the two columns**

fname -> **exrun**

wetter -> **ews**

**SAVE the table (if there is a pop up, select [use all columns]**

**soil information – cdy\_hrzn & profile**

General information about the soil profile like depth and the pointer to the detailed information about the horizons are stored in the table profile!

**OPEN table profile in DBeaver**

It seems, the first two horizons are missing! Let us recreate them with an SQL script within DBeaver!

**OPEN a NEW sql script in DBeaver and ADD THE MISSING horizons 1 and 2:**

INSERT INTO profile (profil, depth, horiz\_name) VALUES

('exsoil', 3, 'exsoil\_1'),

('exsoil', 5, 'exsoil\_2');

DONE? Good now we update the details of the horizon parameters! But ooff, do you see how many parameters these are? There must be a better way to do this but! Remember, Modellers are lazy (in this case you can be lazy, using my scripts :P)! Moreover, I forgot some of the original entries! CANDY has the awesome feature, to auto calculate certain soil parameters! Simply set a parameter you want to autocalculate to -99. But be sure, that you activated auto\_soil in the pg4candy.ini file by setting it to 1 (auto\_soil=1 ).

**OPEN R-Studio and the script ‘session\_10\_update\_cdy\_hrzn.R’.  
Lets explore it together!**

**climate data – cdy\_cldat**

Checking the climate information, the bugs must have eaten the complete year of 2022! Greedy little buggers!

Luckily, we still have this year as .csv (or .txt if you prefer) in the folder **03\_missing\_tables**!

**OPEN the file ‘missing\_weather.csv’ (missing\_weather.txt) from aboves folder**

Unfortunately, the weather data is still in its raw-shape – CANDY will not be able to read it.

We need to change that!

Luckily, there is a pre written R-script, which not only transforms the climate data but ALSO already imports it into our database! Amazing!

**OPEN R-Studio and the R-script ‘session\_10\_import\_climate.R’**

**Lets explore it together!**

**Agricultural Management Schedule – cdy\_madat**

Here, the last crop rotation of the original management is lost! I do not remember what the farmer did. Lets just add a new, simple management

Depending on the time we have left, we will do the lazy way and import the data with a pre-written SQL script OR, preferably, do the entries via the CANDY-GUI

ADD a simple Winter-Wheat crop rotation to the management!

STILL TIME? -> lets do it with the CANDY GUI

NO TIME LEFT? -> lets import the new management with R-Studio (script session\_10\_update\_cdy\_madat.R)

IF YOU HAVE TIME -> Add the new management via the CANDY-GUI (similar to yesterdays exercise with CNP)

NOTE: USE THE candy manual for help (CANDY22\_manual) for more information about management parameters!

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ADD to the management schedule** | | | | | | | |
| management action | crop | fertilizer | tillage | macode | date of year | | | origwert | unit |
|  |  |  | day | month | year |  |  |
| emergence | winter wheat | 1 | 15 | 10 | 2021 | 180 | [dt/ha] |
| fertilization | ammonium nitrate | 4 | 15 | 03 | 2022 | 80 | [kgN/ha] |
| fertilization | ammonium nitrate | 4 | 05 | 05 | 2022 | 100 | [dtFM/ha] |
| fertilization | ammonium nitrate | 4 | 30 | 05 | 2022 | 60 | [dtFM/ha] |
| harvest | winter wheat | 9 | 15 | 08 | 2022 | 180 | [dt/ha] |
| tillage | plough | 4 | 20 | 10 | 2022 | 20 | [cm] |
|  |  |  |  |  |  |  |  |

**Measured values – cdy\_mvdat**

R-SCRIPT ANPASSEN!!

Since we lost our measured values, we have to reimport them as well. Similar to the climate file, there are too many entries for a manual insertion. We will use an R-Script to import the measured values (which are already in a CANDY compliant form).

**OPEN R-Studio and the script ‘session\_10\_import\_tables.R’**

**Lets go through it together**

**Result recording**

The bugs completely devoured our result recording tables! Remember, that CANDY has TWO ways for result recording:

* virtual observations
  + variable depth selection
  + refer to field measured values in cdy\_mvdat
* simulation results
  + are exclusively calculated within the model
  + already set depths
  + great selection

Depending on the time and energy left we will either create result & virtual observation lists with the CANDY-UI or simply import pre-written SQL scripts to update our database!

**IF THERE IS NO TIME:**

**OPEN R-script** Session\_10\_update\_result\_records.R and execute it

**IF THERE IS TIME:**

**CREATE** new result scheme:

* example\_rs\_list

with:

* N2O -> resultnr=997
* Nmin\_0\_3 -> resultnr=112
* Nmin\_0\_6 -> resultnr=113
* C\_MIN -> resultnr=139

**CREATE** new observation scheme:

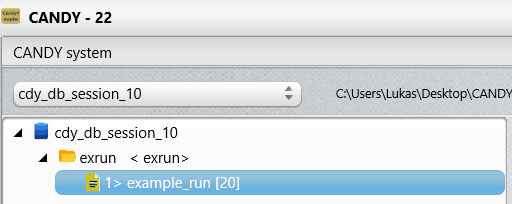
* example\_prop\_list

with:

* Nitrogen Gas Loss -> depthless, m\_ix=76
* Nitrate content of the upper soil -> depth 0-3 dm, m\_ix=1
* Ammonium content of the upper soil -> depth 0-3 dm, m\_ix=3
* Soil Organic Carbon of the upper soil -> depth 0-3 dm, m\_ix=7
* Total Nitrogen content of upper soil Ntot -> depth 0-3 dm, m\_ix=0
* Soil Moisture content in the upper soil -> depth 0-3 dm, m\_ix=10
* Soil Moisture content in the lower soil -> depth 3-6 dm, m\_ix=10

**Start a simulation run with the CANDY UI**

FINALLY we can start our simulation run

**OPEN the CANDY GUI -> expand the ‘exrun’ until fully expanded and click on the run ‘example\_run’**

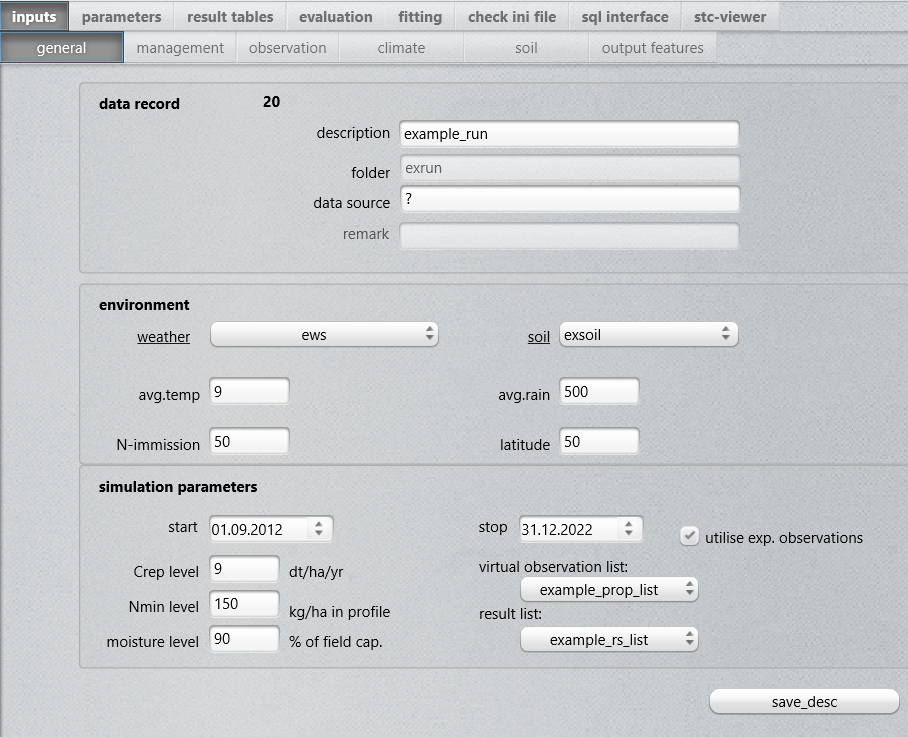
But just before we start the run, we check our run conditions on the right side. Make sure, that you have the correct inputs for

Weather: **ews**

Soil: **exsoil**

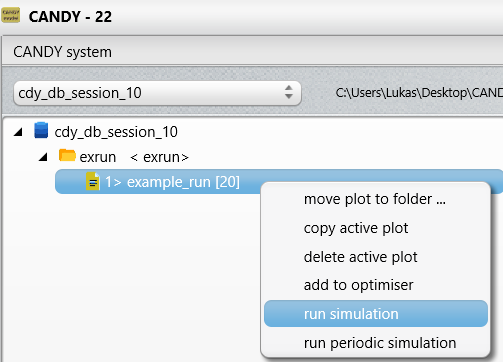
Virtual observation list: **example\_prop\_list**

Result list: **example\_rs\_list**



**Everything set?**

**Go back to the run selection on the left side -> right click on ‘example\_run’ and select [run simulation]**

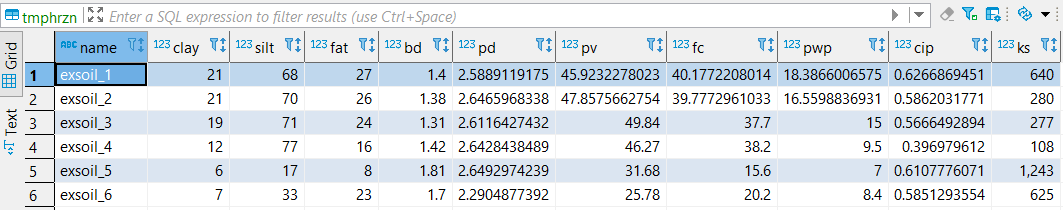
****

**ENJOY THE SHOW 😊**

**BONUS: Remember that we set some soil horizon parameters to -99? After a successful simulation run, a new table in the database is created:**

***Tmphrzn***

**OPEN DBeaver and the table tmphrzn (at the bottom of the table list) and check out the parameters!**



pd – pore density pv – pore volume ks – sat. hydr. conduct.

fc – field capacity pwp – permanent wilting point