

|  |
| --- |
| , RD Dep.  **Oracle DB for DWH and ETL building** |
| Python Task 01 by Anton Gridushko |

Contents

[**Oracle DB for DWH and ETL building** 1](#_Toc69438648)

[1. Python introduction. 3](#_Toc69438649)

[2. Installation. 3](#_Toc69438650)

[3. IDE. 3](#_Toc69438651)

[4.1 Task 1 3](#_Toc69438656)

[4.1.1 TASK 1.1 3](#_Toc69438657)

[4.2 Task 2 4](#_Toc69438658)

[4.2.1 TASK 2.1 4](#_Toc69438659)

[4.2.2 TASK 2.2 5](#_Toc69438660)

[4.2.3 TASK 2.3\* 5](#_Toc69438661)

[4.3 Task 3 5](#_Toc69438662)

[4.3.1 TASK 3.1 5](#_Toc69438663)

[4.3.2 TASK 3.2 6](#_Toc69438664)

[4.3.3 TASK 3.3\* 6](#_Toc69438665)

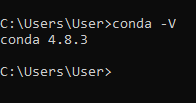
# 

# Python introduction.

Python is an open-source language.

# Installation.

Installed anaconda:

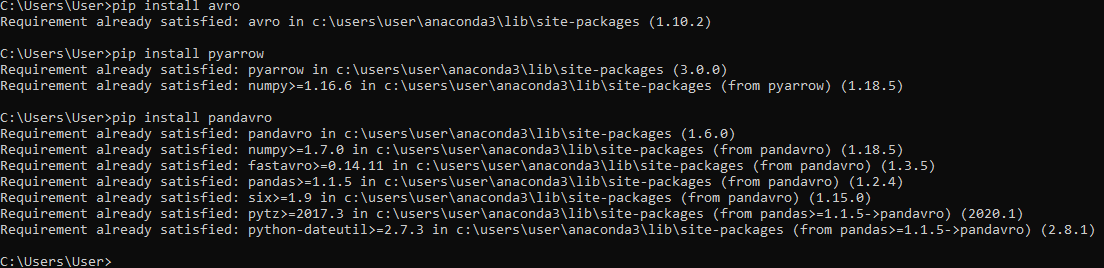


# IDE.

1. Used previously investigated Spyder IDE.



2. Have installed libraries avro, pyarrow and pandavro





## Task 1

Asset:

**1\_generate\_random\_arr.py** script will help you to generated csv file with random numbers from between 1-100, read this array, order it in thread mode, and write result back to csv file.

### TASK 1.1

Task:

In the function write\_dict\_to\_csv\_file header for csv as is a parameter, but it easily could be retrieved from the array arr\_dict\_tofile itself. Could you please retrieve headerRow from array arr\_dict\_tofile and use it instead?

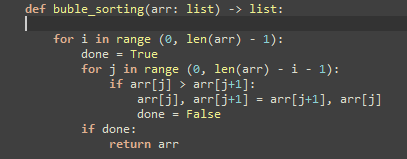
Implementation:



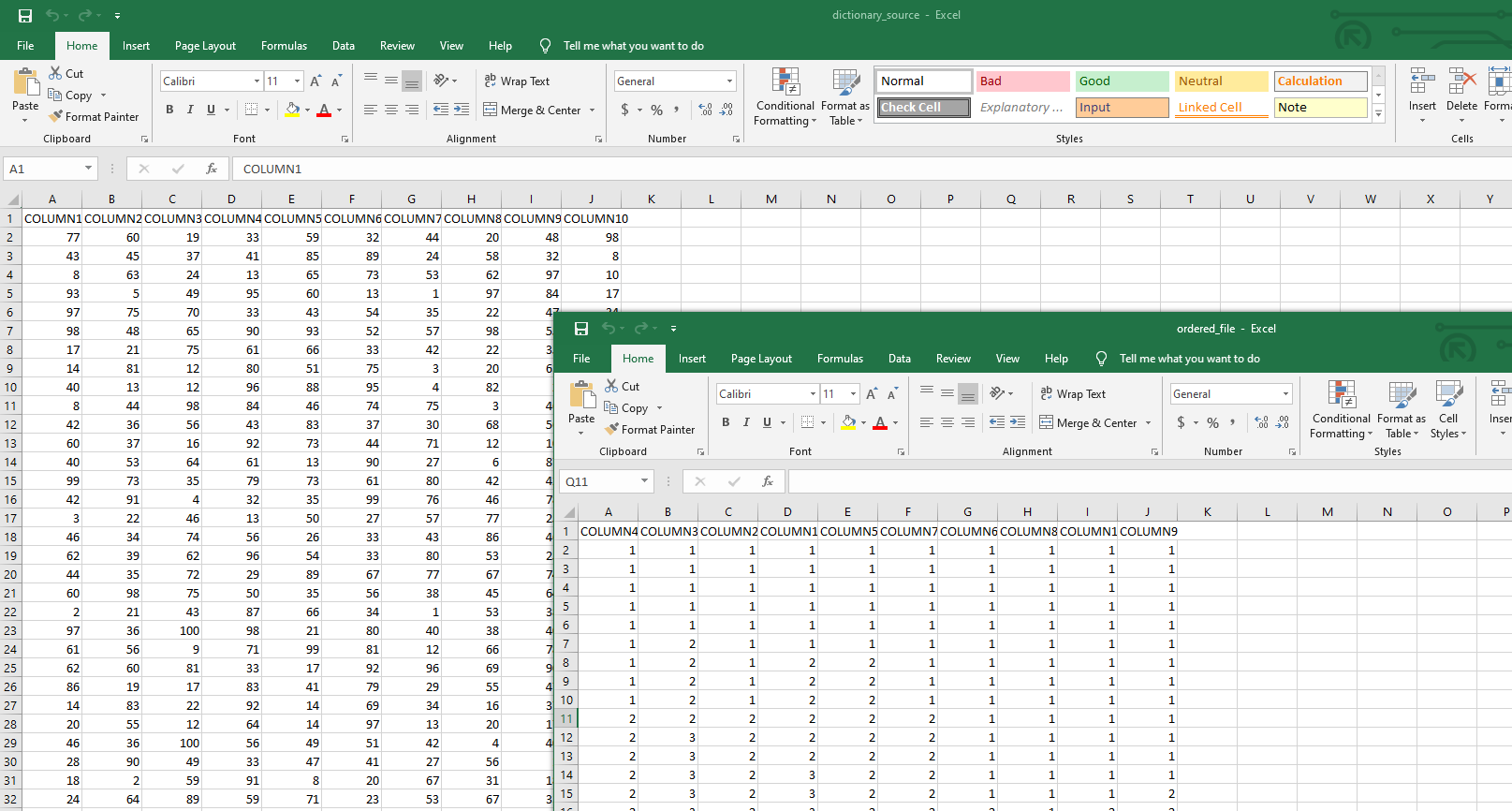
TASK 1.2

Please insert bubble sorting algorithm in the bubble\_sorting function

Implementation:



Proof in line to asset:



## Task 2

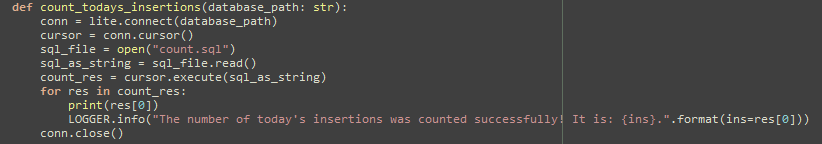
Asset:

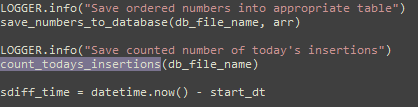
**2\_sqlite\_database\_utilies.py** script will read ordered numbers from the created csv file and loaded these values into the sqlite db file (this file with appropriate table will be created if not exists).

### TASK 2.1

You have to add procedure which will count rows inserted today. LOG this count.

Implementation:





Proof in line to asset:

LOG:



### TASK 2.2

You have only inserts into your table. What is the reason of no updates?

**Answer:**

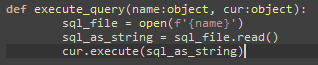
The reason is: every time we create database and the table in it - we use drop command for table **("drop table if exists t\_ordered\_items") that nulls table and all values in it.**

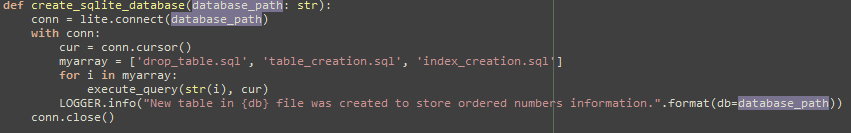
In **save\_numbers\_to\_database** method we have condition **if count > 0**, that checks if ORDERED\_ITEMS\_SURR\_ID exists. If exists - we will update, if not - we insert.

### TASK 2.3\*

Create sql queries as separate files ( \*.sql) and run them from files ( read file -> execute ) instead of hardcoding queries in code.

One of parts of implementation of the task:





Others scripts you can see in special folder named “scripts”. And implementation process in code from atomic commits on GitLab.

## Task 3

Assets:

**3\_write\_to\_utilities.py** will read table information from sqlite db file and write this data into the files with different formats.

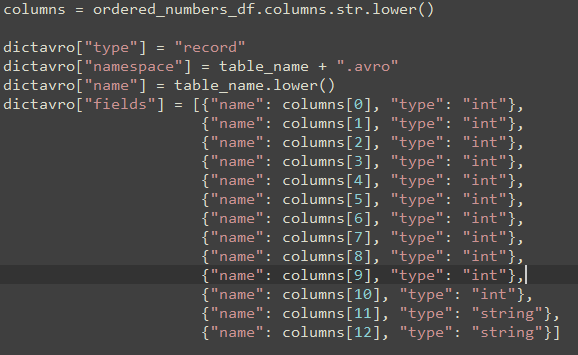
### TASK 3.1

Please use Pandas Dataframe ordered\_numbers\_df instead of using hardcoded column names. Column names could be easily taken from dataframe itself.

Optionally you could retrieve column types from dataframe as well. \*

Implementation:

I think, one of the approaches to avoid hardcode could be:



### TASK 3.2

In the method write\_dataframe\_to\_json please add code to export dataframe into JSON file

(example you could see in the provided folder)

Implementation:



### TASK 3.3\*

After you get dataframe from the sqlite database please leave in it only rows with all values > 20 and use this cut dataframe for further exporting.

Implementation:



One of numerous decisions is:

Take this one – because in my mind this one better to be understood.

