

Author: Andreas Traut

Date: 22.02.2021

[Download as PDF](#)

Algorithms, Data Structures and Coding

0. Introduction

- a) Aim of this repository
- b) Motivation for this repository
- c) Structure of this repository
 - (i) First part: How to improve your coding skills: Certificates and Challenges
 - (ii) Second part: Examples

I. How to improve your coding skills: Certificates and Challenges

- 1. Earn a certificate
- 2. Get into coding challenges

II. Examples

- 1. Python-Examples
- 2. Excel Example
- 3. Access-Example

Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License

Algorithms, Data Structures and Coding

0. Introduction

a) Aim of this repository

The aim of this repository is to share my coding skills, knowledge in data-structures (e.g. classes), abilities in algorithmic thinking (e.g. recursion) and tool-building skills (e.g. Excel/VBA tools). At the same time I will give you lots of hints and solution templates, which will help you to enhance your skills in these topics as well.

b) Motivation for this repository

I am programming in different languages and environments for nearly my whole life:

- Starting in the 1980s / 1990s with [GW-Basic](#) and the integrated development environment (IDE) [Turbo Pascal](#).
- Then in the 2000s / 2010s I started with [C++](#), where I understood the **object oriented way of thinking**. I learnt a lot in C++ at my final years at university as well as during the first years at my first employer. Today my 8 cm thick "*C++ programming bible*" serves as an elevation for my second monitor, which I had to set up due to the Corona-related home office.
- I used [SQL](#) a lot and also got quite skilled in finding solutions with [Visual Basic \(VBA\)](#). VBA (applied in Excel or [Access](#)) is fun for me and served me a lot during my whole professional and private.
- In 2019 I learnt the advantages of the [Jupyter-Notebooks](#): beautiful, intuitive, easy to use and build. But there is something, I don't like in Jupyter-Notebooks, which I will explain below.
- And today I am a big fan of [Python](#): it's so much more fun to use Python instead of C++: I enjoyed not having these opening brackets `{` and closing brackets `}` and `;` at the end of a line! Such a relieve for my eyes.

I am glad, that lots is similar in all these decades: **the way of thinking as a programmer**. My motivation is to give you some basic hints, advises and guidance to improve your coding skills.

c) Structure of this repository

(i) First part: How to improve your coding skills: Certificates and Challenges

In the *first part* I will explain, how certificates and coding challenges can be useful for you to improve your coding skills.

(ii) Second part: Examples

In the *second part* I will work on some interesting examples.

I. How to improve your coding skills: Certificates and Challenges

1. Earn a certificate

A good way for improving your coding skills are by going through some online courses and trying to earn a certificate. There are a lot of other resources: maybe start getting an overview on [Coursera](#). These courses are nice because the teachers are usually highly skilled (from universities) and the technical infrastructure for the courses is rather advanced: there are videos with subtitles and transcript and you can easily navigate through these videos by reading across these transcripts and jumping to the positions in the video, which you want to listen to. You can monitor your learning curve and weekly progress. But the Coursera certificates usually cost some money.

If you want to find something cheaper, then I can recommend the ["Data Structures and Algorithms in Python"](#) from Jovian. When I worked for it in 02/2021 it was for free. It uses Jupyter-Notebooks and is definitively a lot of fun! You will learn in video tutorials and practice with well documented Jupyter-Notebooks how to work with python to solve coding problems systematically.

I am holding the *Certificate "Data Structures and Algorithms in Python"* (see [here](#)) which covers important data structures and algorithms like [binary trees](#), [recursion](#), [directed graphs](#) and the [knapsack problem](#). I think knowing about these topics is very important, because a high performance cluster can solve your problem only sometimes: often a good data structure and algorithm will be a lot more powerful than any hardware solution.

There are various other resources for earning a certificate and listing up, what I found is not very helpful at the end for you: try to find **the certificate which YOU want to earn!** I promise: working for it is a lot of fun.



Issued April 8th, 2021



CERTIFICATE OF ACCOMPLISHMENT

This is awarded to

Andreas Traut

For successfully completing

Data Structures and Algorithms in Python

an online course offered by Jovian, representing approximately 60 hours of coursework

AAKASH N S
COURSE INSTRUCTOR
FOUNDER, JOVIAN

*Authenticity of this certificate can be verified at <https://jovian.ai/certificate/MFQTNRV04>

2. Get into coding challenges

Another advice I can give you is to get into coding challenges. When you accept a coding challenge, then a problem will be shown and would have to solve it in your preferred programming language (python, java, C++,...). I tried [LeetCode](https://leetcode.com/) and you will find a lot of other websites, which provide similar concepts. On the left is the problem, on the right some place to program a solution:

The screenshot shows the LeetCode interface for the problem "973. K Closest Points to Origin". The problem is marked as "Medium" and has 2733 likes and 141 comments. The description states: "We have a list of points on the plane. Find the K closest points to the origin (0, 0). (Here, the distance between two points on a plane is the Euclidean distance.) You may return the answer in any order. The answer is guaranteed to be unique (except for the order that it is in)."

Example 1:
Input: points = [[1,3],[-2,2]], K = 1
Output: [[-2,2]]
Explanation: The distance between (1, 3) and the origin is sqrt(10). The distance between (-2, 2) and the origin is sqrt(8). Since sqrt(8) < sqrt(10), (-2, 2) is closer to the origin. We only want the closest K = 1 points from the origin, so the answer is just [[-2,2]].

Example 2:
Input: points = [[3,3],[5,-1],[-2,4]], K = 2
Output: [[3,3],[-2,4]]
(The answer [[-2,4],[3,3]] would also be accepted.)

Note:
1. 1 <= K <= points.length <= 10000
2. -10000 < points[i][0] < 10000
3. -10000 < points[i][1] < 10000

Accepted 421,006 | Submissions 651,819

Can this reaction in a real interview? ☐ Yes ☒ No

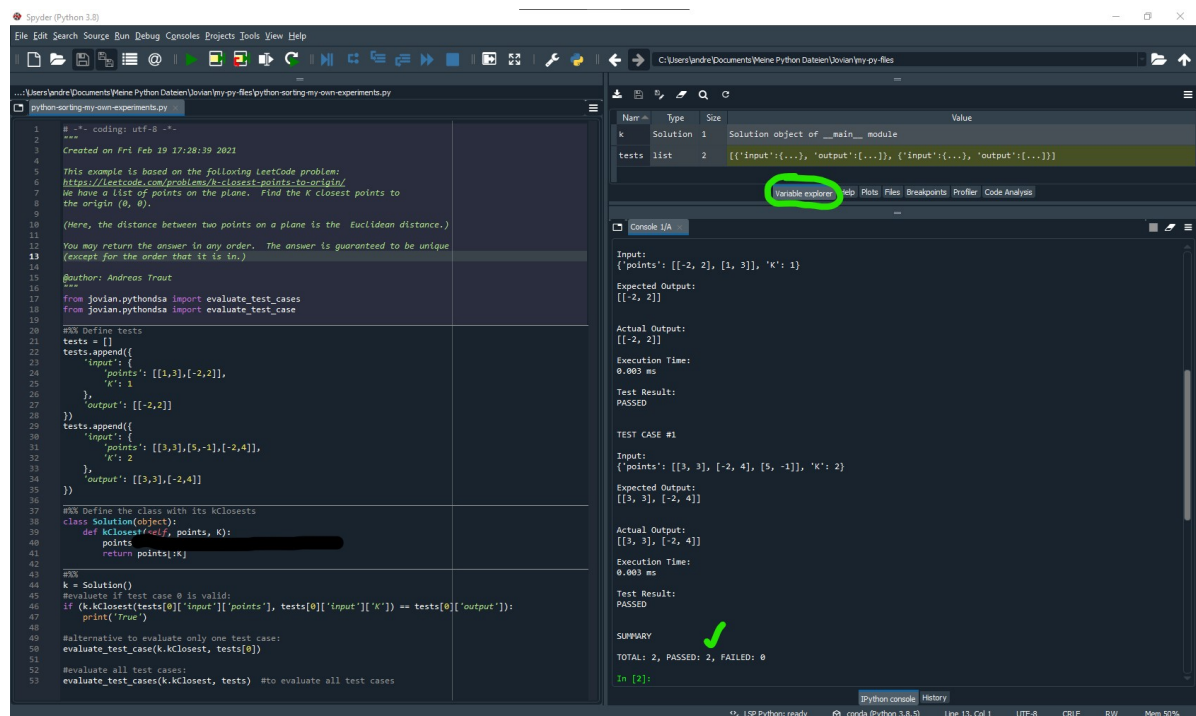
31/76

Run Code Submit

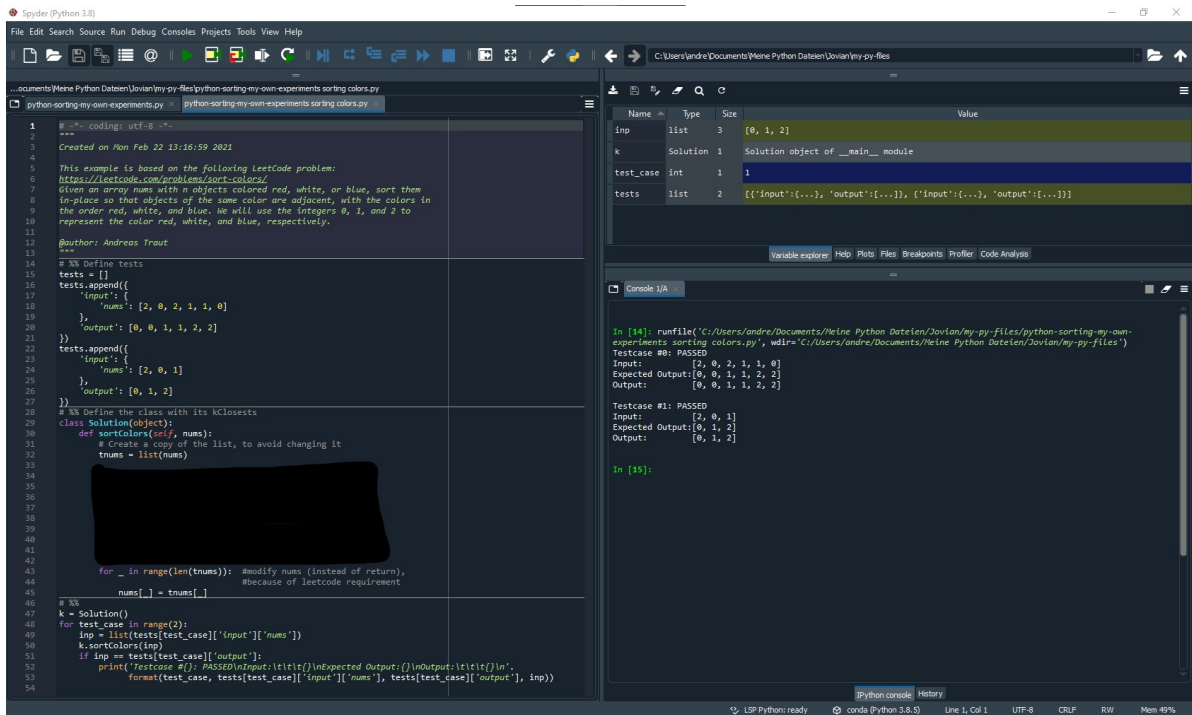
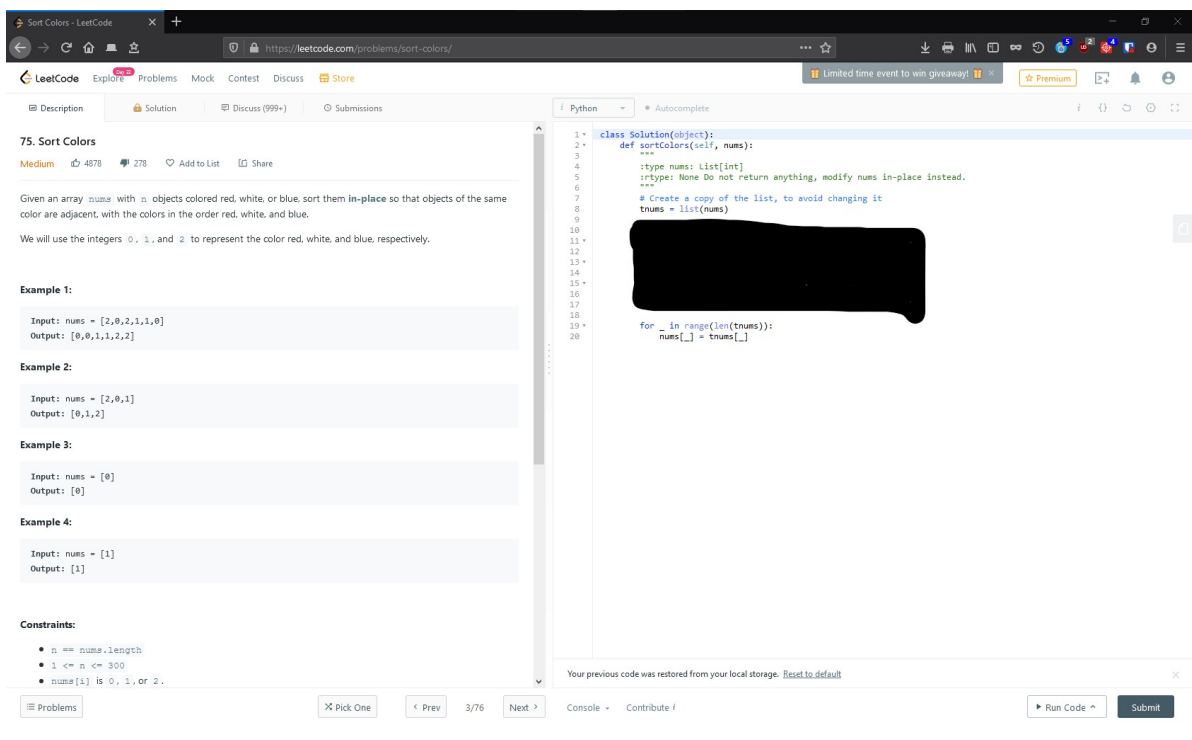
```
class Solution(object):
    def kClosest(self, points, K):
        points = sorted(points, key=lambda x: x[0]**2 + x[1]**2)
        return points[:K]
```

As I am not allowed to publish solutions for these LeetCode problems I had to black out my solutions. Some of these problems were quite interesting for me so I wanted to have them in my integrated development environment (IDE) [Spyder-IDE](#) in order to debug through the code and extend the examples a bit. I recommend to use an integrated development environment (IDE) as often as you can, instead of always going through Jupyter Notebooks. In my opinion Jupyter Notebooks are **not** always the best environment for learning to code! I agree, that Jupyter Notebooks are nice for doing documentation of python code. It really looks beautiful. But I prefer debugging in an IDE instead of a Jupyter Notebook: having the possibility to set a breakpoint can be a pleasure for my nerves, specially if you have longer programs. Some of my longer Jupyter Notebooks feel from the hundreds line of code onwards more like pain than like anything helpful. And I also prefer having a "help window" or a "variable explorer", which is smoothly integrated into the IDE user interface. And there are a lot more advantages why getting familiar with an IDE is a big advantage compared to the very popular Jupyter Notebooks! I am very surprised, that everyone is talking about Jupyter Notebooks but IDEs are only mentioned very seldom. But maybe my preferences are also a bit different, because I grew up in a [MS-DOS](#) environment. :-)

Here is how the problem from above looks like in the Spyder-IDE:



Another example from LeetCode: the [Sort-Colors](#) problem:



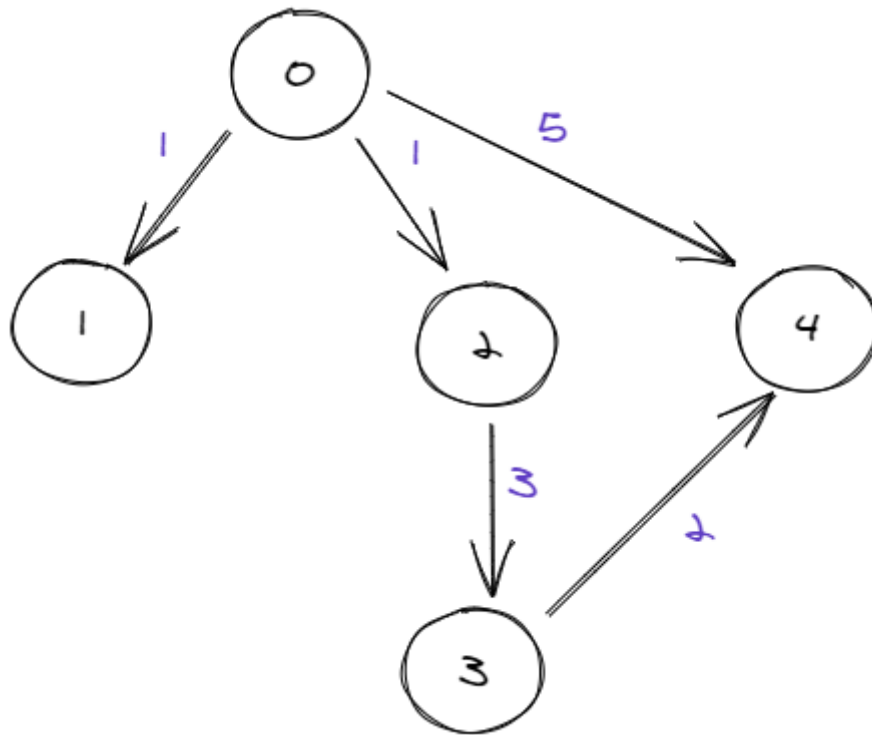
II. Examples

In the *second part* I will work on some interesting examples, which will be available as `.py` Python-Files, Jupyter-Notebooks or will be Tools like Excel/VBA or Access.

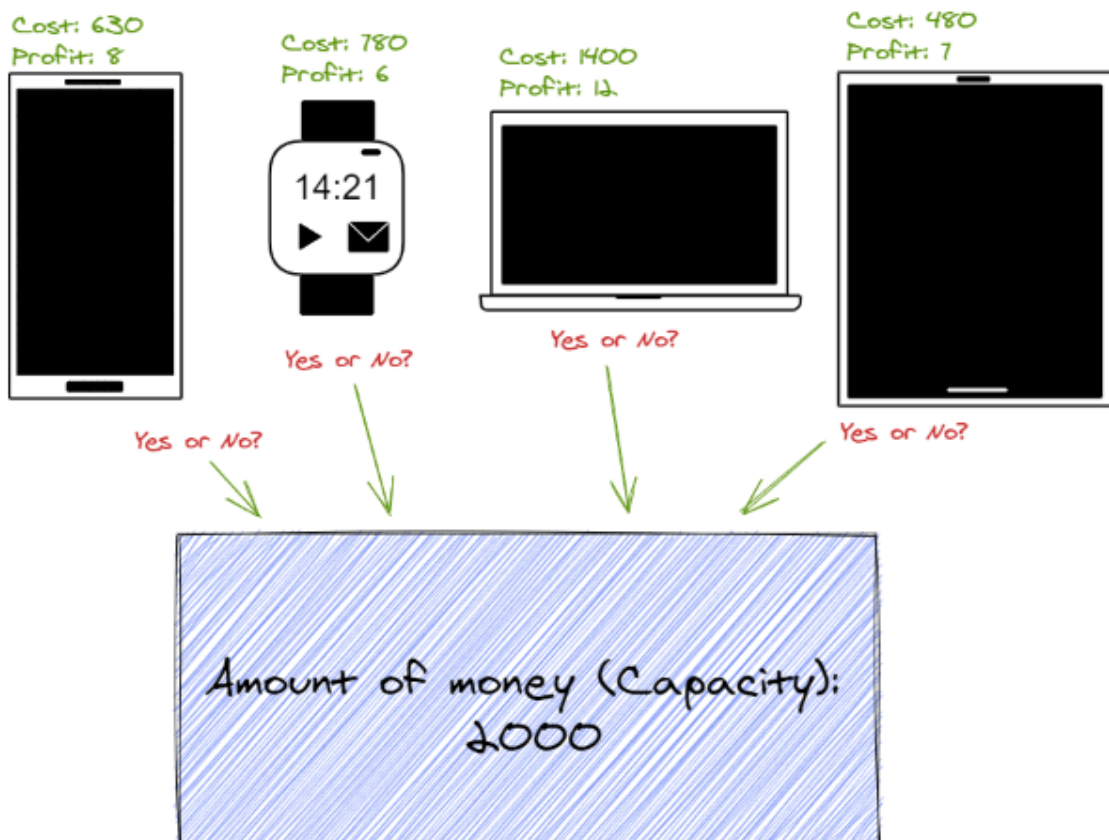
1. Python-Examples

I solved several problems, which require algorithmic thinking (e.g. recursion) or knowledge about data-structures (e.g. casses) and I shared my solutions as `.py` Pyhton-Files here. These problems are for example:

- Find the K nearest points to the origin, given some points in the plane.
- Sort the list of color-codes.
- Calculate the delay time which is necessary to send a signal from the source "zero" to all other nodes in a network graph (a directed, weighted graph):



- A "Rucksack/Knapsack" problem: imagine, that you have a limited amount of money (capacity=2000 Euros) and you have the choice between several devices, each having it's own cost (mobile phone: 630 Euros, smartwatch: 780 Euros, computer: 1400 Euros, tablet: 480 Euros). Assume, that you have assigned to each of these devices an "individual profit" (the value, which this device creates for you). Which devices should you buy in order to maximize your profit? What would be the best choice?



Each time I will build some test-cases before going into the solution and also use the Jovian "evaluate_test_cases" module to efficiently perform the tests.

https://github.com/AndreasTraut/Algorithms-Data-Structures-and-Coding/tree/main/Python_Examples

During my career I implemented a lot of Excel/VBA solutions: one was a Excel/VBA project management tool, which organized and structured a complex project flow of a team of 7 people. My Excel/VBA solution is used on a daily basis and is running for already 2 years now.

The first step is to define the three steps ("1. Basket Items", "2. Quality Check", "3. Delivery") and assure in the tab "configuration" that the predefined dropdown cells and color codes, are always **clear**. Like this you will get **consistency in your processes and data**. Changing the color codes or status description here will automatically update the whole Excel/VBA solution and therefore you will always have consistency.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1																			today:	08.03.2021	
2																			enter number		
3																			of days shown		
4																			in the past:	10	
5																					
6																					
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					

responsible	color code	color
Andreas	0	white
Tom	1	green
Jerry	2	yellow
Mickey Mouse	3	orange
Donald Duck	4	red
Tic	5	light yellow

priority	color code
high	4
medium	2
low	5

Basket items	
status	color code
todo	4
request sent	5

1

2

3

Order from client please fill in an order identifier, client name, priority and deadline				Request to suppliers: please fill in the basket items, status, date, responsible and (if needed) comments				Quality Check: please fill in the status, date, responsible and (if needed) comments for the internal quality check of the items				Delivery / Shipping: please fill in here the status, date and responsible and (if needed) comments for the shipping of the items to the client							
Orders								Basket Items				Quality Check				Delivery			
Identifier	client name	priority	deadline	basket item description	status	date	responsible	comment	status	date	responsible	comment	status	date	responsible	comment			
order001	Clientusmaximus	medium	05.04.2021	paper, A4, 100 pieces	request sent	17.03.2021	Andreas												
order001	Clientusmaximus	medium	05.04.2021	paper A3, 50 pieces	request sent														
order001	Clientusmaximus	medium	05.04.2021	nails, 5mm, 50 pieces	confirmed														
order002	Clientusmaximus	medium	05.04.2021	nails, 4mm, 70 pieces	ok, received				ok, to be delivered										
order002	Clientusmaximus	medium	05.04.2021	wires, 15m lengths	ok, received				item nok - reclamation sent		Donald Duck								
order002	Clientusmaximus	medium	05.04.2021	plates, 5m x 5m, 100 pieces	ok, received						Donald Duck								
order002	Clientusmaximus	medium	05.04.2021	bricks, 100 pieces	confirmed														
order003	Clientusmaximus	medium	05.04.2021	grids, 10cm x 10cm	todo		Andreas												
order003	Clientusmaximus	medium	05.04.2021	fittings, 60 pieces	todo		Andreas												
order003	Clientusmaximus	medium	05.04.2021	seals, 3 pieces	confirmed														
order004	Clientusdominus Ltd	high	07.04.2021	nails, 4mm, 70 pieces	request sent														
order004	Clientusdominus Ltd	high	07.04.2021	wires, 15m lengths	request sent														
order004	Clientusdominus Ltd	high	07.04.2021	plates, 5m x 5m, 100 pieces	ok, received				item nok	08.03.2021	Donald Duck								
order005	Clientadri	low	31.03.2021	paper, A4, 100 pieces	ok, received	08.03.2021	Andreas		ok, to be delivered	08.03.2021	Andreas								
order006	Engine Client	medium	01.04.2021	grids, 10cm x 10cm	ok, received	08.03.2021	Tom		ok, to be delivered										
order006	Engine Client	medium	01.04.2021	plates, 5m x 5m, 100 pieces		08.03.2021	Tom		item nok										

configuration

Clients

Orders

OrderStatistics

OrderBasketItems-ToDo

QualityCheck-Items NOK

Delivery-ToDo

Additionally you may want to inform your client about the intermediate status of their orders by automatically generated Outlook-Emails. I implemented this in Excel/VBA and pressing one button will create an Outlook-Email, where email-address, subject and email-text is filled automatically by my VBA code as follows:

Order number: order004 - Nachricht (HTML)

Senden

Von: andreas.traut@outlook.com

An: dominik@mailyou.com

Cc:

Betreff: Order number: order004

Dear Dominik,

I would like to inform you about the current status of the order number order004:

1 out of 3 items have already been delivered by our suppliers and are ready for our internal quality check
 0 out of 3 items have successfully passed our internal quality check and are ready for being shipped.

Please don't hesitate to contact me in case of any questions.

Kind regards
 Andreas

Furthermore some statistics should help you to see, where you have issues in your order process (like failed quality checks,...):

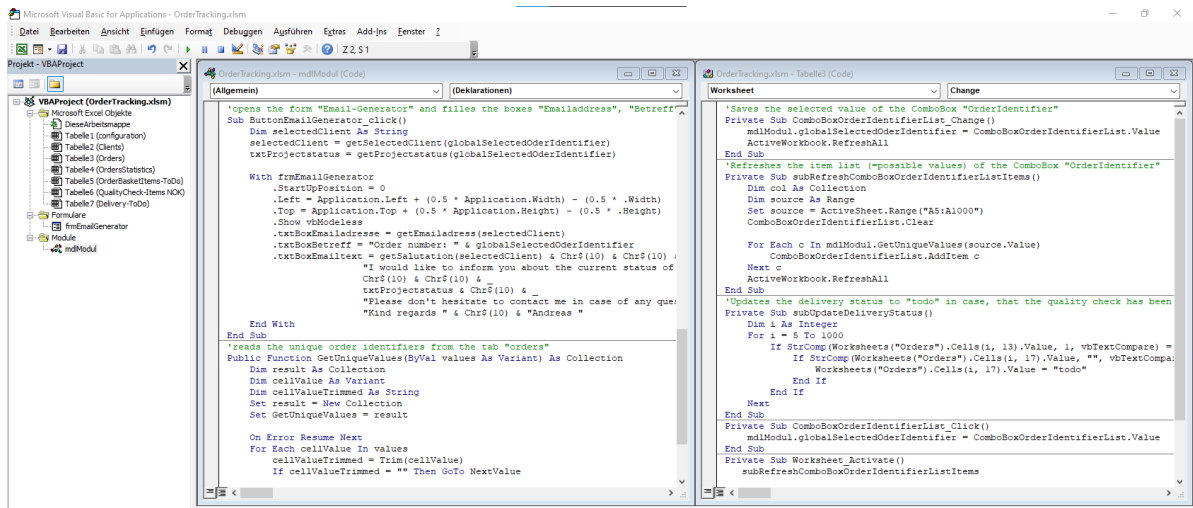
Order Statistics:
Thes overview show the statistics for the orders and status. These values are also shown in the Email (see "Generate Email" Button)

Press <Ctrl>+<Alt>+<F5> to refresh all tabs.

Anzahl von status	Spaltenbeschriftungen	confirmed	request sent	todo (Leer)	ok, received	esamtergebnis
order001		1	2		3	3
order002		1		2	3	4
order003		1			1	1
order004		2			1	3
order005					1	1
order006					1	1
order007					1	1
Gesamtergebnis		3	4	3	6	16

Anzahl von status	Spaltenbeschriftungen	item nok	item nok - reclamation sent (Leer)	ok, to be delivered	Gesamtergebnis
order001					
order002		1		1	3
order003					
order004		1			1
order005				1	1
order006		1		1	2
order007					
Gesamtergebnis		3	1	3	7

Let's have a short look into the VBA code:



You can download this example from my repository:

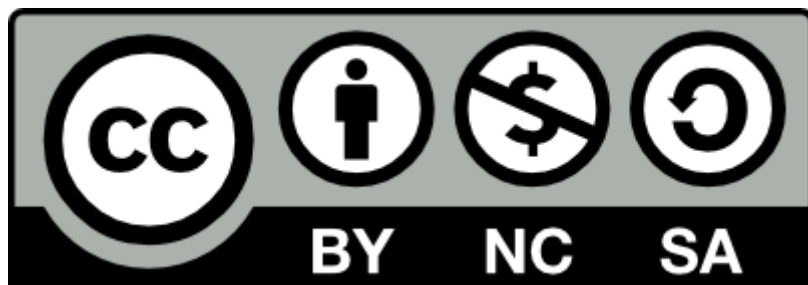
https://github.com/AndreasTraut/Algorithms-Data-Structures-and-Coding/tree/main/Excel_Example

3. Access-Example

During my career I also worked with Access solutions. One was for a team of 20 people who were working simultaneously with their Access-frontends on one Access-backend. The aim was to assure a structured data-entry of the whole team into the Access backend database by using Access-Forms. At the end of the project the filled database tables had been joined with SQL queries to further databases in order to aggregate a very specific result table.

I also worked on other Access solutions and will provide an example of an Access solution in short time here.

Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

