# User guide Albert Gein SuperPy Software

# 1. Installing the software

Place the folder: superpy in a secure folder.

Example:  
C:\Users\username\Winc

### Additional Python Packages used:

* pyparsing
* rich
* Matplot
* Numpy
* pandas

Make sure they are installed, run pip list in windows Powershell to check.

# 2. Reach the software package

Open a terminal like Windows PowerShell for this example

Change directory to the superpy folder using the cd command.

In the above example use:

cd Winc\superpy

which returns:

C:\Users\username\Winc\superpy

Try reaching it using the following command:

python -m main -h

This should return the Albert Gein SuperPy Software with available options!

## 3. Time manipulation

This software uses local time manipulation which is simply stored in a text file timestatus.txt

|  |  |  |
| --- | --- | --- |
| All command lines require the text: python -m main in front of the option | | |
| **Short** | **Long** | **Help** |
| -t | --advance\_time | Changes the date of the software. There are three available options here.  update  Updates the software time to the local time.  current  Shows the current date saved in the software.  YYYY-MM-DD  Insert a date according to this format to change the date of the software. |
| Example: python -m main -t 2022-09-26 (Changes date to 26 september 2022)  Example: python -m main -t current (Shows software’s date)  Example: python -m main -t update (Update software to local time) | | |

## 4. Adding products

|  |  |  |
| --- | --- | --- |
| **Short** | **Long command** | **Help** |
| -a | --add | Adds or updates a product to the pricelist.csv file. If it already exists it will change the inserted prices.  Requires 3 more arguments:  -p product name  -b buy price  -s sell price |
| -p | --product\_name | Name of the product. |
| -b | --buy\_price | Buy/cost price of a product. |
| -s | --sell\_price | Sell price of a product. |
| Example: python -m main --add -p apple -b 0.5 -s 1 | | |

|  |  |  |
| --- | --- | --- |
| **Short** | **Long command** | **Help** |
| -l | --product\_list | Returns a list of available products from pricelist.csv file. |
| Example: python -m main -l | | |

## 5. Buying and selling products

NOTE. You can not sell items that are not in the pricelist or when you are out of stock!

|  |  |  |
| --- | --- | --- |
| **Short** | **Long command** | **Help** |
| -c | --command | Used to buy or sell a product. One of the following arguments must be given: buy or sell  Requires 3 more arguments.  -p product name  -q quantity  -e expiry (not required when selling a product). |
| -q | --quantity | Amount of the product. |
| -e | --expiry | Expiry date of a product formatted: YYYY-MM-DD. |
| Examples:  python -m main -c buy -p apple -q 5 -e 2022-10-14  python -m main -c sell -p banana -q 2 | | |

## 6. Report and chart

|  |  |  |
| --- | --- | --- |
| **Short** | **Long command** | **Help** |
| -r | --report | Reports revenue profit or inventory status. One of the following arguments must be given:  revenue or profit or inventory or chart\_total or chart\_products  For inventory you can specify which product you are looking for using the -p product name command.  To receive a chart from the current year use the chart argument. |
| -d | --date | Specify a date for report. Default = today .  Allowed are:  today  yesterday  YYYY-MM-DD |
| Example: python -m main -r inventory (returns full inventory list)  Example: python -m main -r inventory -p apple (returns how many apples in stock)  Example: python -m main -r revenue -d yesterday (returns yesterday’s profit)  Example: python -m main -r profit -d 2022-09-29 (returns profit of 29 September 2022)  Example: python -m main -r chart\_total (returns revenue and profit from 2022)  Example: python -m main -r chart\_products (returns revenue from 2022 per product) | | |