

Gebruikershandleiding SuperPy

Zie ook de Technisch handleiding SuperPy

Inhoud

Gebruikershandleiding SuperPy.....	1
SuperPy	2
Gebruikersinterface	2
CLI-commando's	2
Hoofdprocessen	2
Aankoop van producten (buy)	2
Verkoop van producten (sell).....	2
Ondersteunende processen	2
Overzichten (report)	2
Beheersprocessen	3
Datum (date).....	3
Tijdreizen (shiftdays).....	3
Beginsituatie instellen (initiate).....	3
Csv-bestanden	3
SuperPy.csv	3
Bought.csv	3
Sold.csv	4
Inventory.csv	4
Expired.csv	4
Concreet uitgewerkt voorbeeld	5
Beginsituatie op 01 januari 2023	5
Op 02 januari 2023	5
Command line interface (CLI)	5
Inhoud cvs-bestanden	5
Op 03 januari 2023	6
Command line interface (CLI)	6
Inhoud cvs-bestanden	6
Op 04 januari 2023	7
Command line interface (CLI)	7
Inhoud cvs-bestanden	7
Op 05 januari 2023	7
Command line interface (CLI)	8
Inhoud cvs-bestanden	8

In geel: nieuw of aangepast in deze versie.

SuperPy

M.b.v. SuperPy wordt de aankoop en verkoop van producten gefaciliteerd. Tevens biedt SuperPy de mogelijkheid om overzichten op te vragen en te 'tijdreizen'.

Gebruikersinterface

- SuperPy is een zogenaamde *Command Line Interface* (CLI) waarmee de gebruiker direct opdrachten intypt om een bepaalde functionaliteit uit te laten voeren.

CLI-commando's

- Gebruik de help-functie om te zien welke mogelijke CLI-commando's er zijn. Type daarvoor in de CLI in **python super.py -h** en druk op <enter>.
- In het [concreet uitgewerkte voorbeeld](#) wordt getoond hoe elk CLI-commando gebruikt kan worden.

Hoofdprocessen

Aankoop van producten (buy)

M.b.v. het CLI commando buy

- om 1 product te kopen,
- waarbij productnaam, inkoopprijs en houdbaarheidsdatum (in die volgorde) moeten worden opgegeven.

De aankoop van een product leidt altijd tot een extra regel in Bought.csv.

Gebruik de help-functie om de syntax te bekijken. Type in de CLI in **python super.py buy -h** en druk op <enter>.



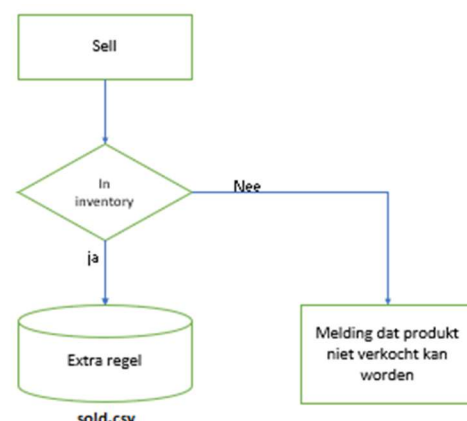
Verkoop van producten (sell)

M.b.v. het CLI commando sell

- om 1 product te verkopen,
- waarbij productnaam en verkoopprijs (in die volgorde) moeten worden opgegeven.

Gebruik de help-functie om de syntax te bekijken. Type in de CLI in **python super.py sell -h** en druk op <enter>.

De verkoop van een product kan alleen als dat product in voorraad aanwezig is (aanwezig in Inventory.csv). En leidt alleen in zo'n geval tot een extra regel in Sold.csv.



Ondersteunende processen

Overzichten (report)

M.b.v. het CLI commando report

- om m.b.v. 3 optionele argumenten (--type, --period, --csv) een specifieke stukje informatie op te vragen.

Let op: bij deze optionele argumenten moet voorafgaand aan de waarde de naam van het argument (incl. het voorvoegsel "--") opgegeven worden. Zie ook [Concreet uitgewerkt voorbeeld](#)..

--type	--date	--csv	Volledige commando
revenue (*)	[date]		python super.py report --type revenue --date [date]
profit (**)	[date]		python super.py report --type profit --date [date]
this_date_revenue (*)	[date]		python super.py report --type this_date_revenue --date [date]
this_date_profit (**)	[date]		python super.py report --type this_date_profit --date [date]
csv		superpy.csv bought.csv sold.csv expired.csv	python super.py report --type csv --csv superpy.csv python super.py report --type csv --csv bought.csv python super.py report --type csv --csv sold.csv python super.py report --type csv --csv expired.csv

		inventory.csv all (***)	python super.py report --type csv --csv inventory.csv python super.py report --type csv --csv all
--	--	----------------------------	--

(*) In SuperPy 'revenue' is defined as: the total price of all (and only) SOLD products, m.a.w. de opbrengst uit verkoop.
 (**) In SuperPy 'profit' is defined as: the total price of (all sold products - bought products - expired products).
 (***) Toont alle 5 csv-bestanden.

Gebruik de help-functie om de syntax te bekijken: Type in de CLI in **python super.py report -h** en druk op <enter>.

Beheersprocessen

Datum (date)

M.b.v. het CLI commando date

- kan de datum die SuperPy als “de dag van vandaag” gebruikt worden getoond.

Gebruik de help-functie om de syntax te bekijken. Type in de CLI in **python super.py date -h** en druk op <enter>.

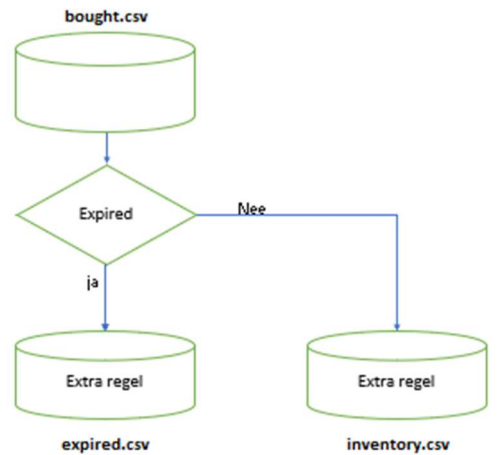
Tijdreizen (shiftdays)

M.b.v. het CLI commando shiftdays

- kan de datum die SuperPy als “de dag van vandaag” gebruikt worden opgeschoven. Dat gebeurt in hele dagen, verder weg de toekomst in bij een negatief getal, of meer recent bij een positief getal.

Gebruik de help-functie om de syntax te bekijken. Type in de CLI in **python super.py shiftdays -h** en druk op <enter>.

Nadat “de dag van vandaag” is gewijzigd worden automatisch op de achtergrond de bestanden Expired.csv en Inventory.csv opnieuw gevuld. Dat gebeurt ook na [aankoop van producten \(buy\)](#) en na [verkoop van producten \(sell\)](#).



Beginsituatie instellen (initiate)

T.b.v. het een eigen [concreet uitgewerkt voorbeeld](#): Type in **python super.py initiate own** en druk op <enter>.

Gebruik de help-functie om de syntax te bekijken. Type in de CLI in **python super.py initiate -h** en druk op <enter>.

Csv-bestanden

Alle gegevens die SuperPy gebruikt worden opgeslagen in csv-bestanden.

SuperPy.csv

SuperPy.csv
my_today
2023-01-01

- Bevat de SuperPy ‘dag van vandaag’.
 - Wordt m.b.v. het CLI-commando [initiate](#) gezet op 2023-01-01.
 - Kan m.b.v. [tijdreizen \(shiftdays\)](#) aangepast worden.

Bought.csv

Bought.csv				
bought_id	product_name	buy_date	buy_price	expiration_date

- Bevat alle gekochte producten.
- Krijgt altijd een extra regel bij [aankoop van producten \(buy\)](#).
- Het veld *bought_id* hoeft niet door de gebruiker via de CLI te worden ingevoerd. het systeem zorgt daarvoor en verhoogt het id steeds met + 1.
- Het veld *buy_date* hoeft niet door de gebruiker via de CLI te worden ingevoerd. Het systeem gebruikt daarvoor de “datum van vandaag” zoals SuperPy die hanteert (opgeslagen in SuperPy.csv).

De veldnamen in grijs hoeven door de gebruiker niet in de CLI opgegeven worden, want die worden door het systeem toegevoegd. Het betreft

- Beide id's: het systeem zorgt voor een id en gebruikt daarvoor per a tie de laatste uitgegeven id + 1.
- Buy_date en sell_date: het systeem gebruikt hiervoor de datum die in SuperPy is ingesteld als de dag van vandaag, opgeslagen in SuperPy.csv.

Sold.csv

Sold.csv			
sold_id	bought_id	sell_date	sell_price

- Bevat alle verkochte producten.
- Kan een extra regel krijgen bij [verkoop van producten \(sell\)](#). Er staat nadrukkelijk “kan”, want als met *sell* de opdracht wordt gegeven om een product te verkopen dat op dat moment niet in de voorraad aanwezig is (niet aanwezig is in Inventory.csv), dat wordt alleen de boodschap gegeven ‘Sorry, selling <product_name> is not possible. The product is not in inventory’.
- Het veld *sold_id* hoeft niet door de gebruiker via de CLI te worden ingevoerd. het systeem zorgt daarvoor en verhoogt het id steeds met + 1.
- Het veld *bought_id* hoeft ook niet door de gebruiker via de CLI te worden ingevoerd. Het systeem haalt die bij verkoop van een product op uit het csv-bestand met de voorraad (Inventory.csv).
- Het veld *sell_date* hoeft niet door de gebruiker via de CLI te worden ingevoerd. Het systeem gebruikt daarvoor de “datum van vandaag” zoals SuperPy die hanteert (opgeslagen in SuperPy.csv).

Inventory.csv

Inventory.csv		
date	bought_id	product_name

- Bevat de voorraad, d.w.z. gekochte producten die niet verkocht én waarvan de uiterste verkoopdatum niet is bereikt)ze mogen dus nog verkocht worden’.
- Wordt opnieuw opgebouwd na [aankoop van producten \(buy\)](#), [verkoop van producten \(sell\)](#) en [tijdreizen \(shiftdays\)](#).

Expired.csv

Expired.csv		
date	bought_id	buy_price

- Bevat de gekochte producten die nog niet zijn verkocht én waarvan de uiterste verkoopdatum is bereikt. Die producten mogen op die dag (en uiteraard later ook) niet meer verkocht worden.
- Wordt opnieuw opgebouwd na [aankoop van producten \(buy\)](#), [verkoop van producten \(sell\)](#) en [tijdreizen \(shiftdays\)](#).

Concreet uitgewerkt voorbeeld

Beginsituatie op 01 januari 2023

- De winkel opent op 01jan2023.
- Er is dan nog niets gekocht of verkocht.

In deze beginsituatie

- bevat SuperPy.csv 2 regels: de kolomnaam my_today en, op de tweede regel, de datum 2023-01-01.
- bevatten de 4 andere cvs's (bought, sold, expired, inventory) bevatten elk 1 regel: alleen de kolomnamen.

SuperPy.csv	Bought.csv					Sold.csv				Inventory.csv			Expired.csv		
my_today	bought_id	product_name	buy_date	buy_price	expiration_date	sold_id	bought_id	sell_date	sell_price	date	bought_id	product_name	date	bought_id	buy_price
2023-01-01															

Deze beginsituatie is in te stellen via de Command line interface (CLI): type in **python super.py initiate own** en druk daarna op <enter>.

Op 02 januari 2023

- Gekocht wordt een partij mandarijnen, voor een bedrag van 0,8 en met een uiterste houdbaarheidsdatum van 9 januari 2023.

Command line interface (CLI)

- Type in **python super.py shiftdays 1** en druk daarna op <Enter>.
- Je kan controleren dat de datum inderdaad is aangepast. Type in **python super.py date** of **python super.py report --type csv --csv superpy.csv** en druk op <enter>.
- Type in **python super.py buy orange 0.8 2023-01-09** en druk daarna op <Enter>.

Inhoud cvs-bestanden

Je kan controleren dat de csv-bestanden correct zijn aangepast. Type in **python super.py report --type csv --csv all** om alle bestanden te tonen.

SuperPy.csv	Bought.csv					Sold.csv				Inventory.csv			Expired.csv		
my_today	bought_id	product_name	buy_date	buy_price	expiration_date	sold_id	bought_id	sell_date	sell_price	date	bought_id	product_name	date	bought_id	buy_price
2023-01-02	1	orange	2023-01-02	0.8	2023-01-09					2023-01-02	1	orange			

Op 03 januari 2023

- De sinaasappelen die gisteren gekocht zijn worden vandaag verkocht, voor een bedrag van 2.
- Vandaag worden appels gekocht, voor een bedrag van 2 en met een houdbaarheidsdatum van 2023-01-13.
- En ook bananen voor een bedrag van 2.5 en met een houdbaarheidsdatum van 2023-01-05

Command line interface (CLI)

- Type in **python super.py shiftdays 1** en druk daarna op <Enter>.
- Je kan controleren dat de datum inderdaad is aangepast. Type in **python super.py date** of **python super.py report --type csv --csv superpy.csv** en druk op <enter>.
- Type in **python super.py sell orange 2** en druk daarna op <Enter>.
- Type in **python super.py buy apple 2 2023-01-13** en druk daarna op <Enter>.
- Type in **python super.py buy bananen 2.5 2023-01-05** en druk daarna op <Enter>.

Inhoud cvs-bestanden

Je kan controleren dat de csv-bestanden correct zijn aangepast. Type in **python super.py report --type csv --csv all** om alle bestanden te tonen.

SuperPy.csv	Bought.csv					Sold.csv				Inventory.csv			Expired.csv		
my_today	bought_id	product_name	buy_date	buy_price	expiration_date	sold_id	bought_id	sell_date	sell_price	date	bought_id	product_name	date	bought_id	buy_price
2023-01-03	1	orange	2023-01-02	0.8	2023-01-09	1	1	2023-01-03	2	2023-01-03	2	apple			
	2	apple	2023-01-03	2	2023-01-13					2023-01-03	3	banana			
	3	banana	2023-01-03	2.5	2023-01-05										

Revenue en Profit

De omzet (*revenue*) t.m. vandaag is op te vragen: type in **python super.py report --type revenue --date 2023-01-03** en druk op <enter>.

Dit levert: 2.0.

Ook kan de omzet (*revenue*) van alleen de opgegeven dag opgevraagd worden: type in **python super.py report --type this date revenue --date 2023-01-03** en druk op <enter>.

Dit levert ook: 2.0.

De winst (*profit*) t.m. vandaag is op te vragen: type in **python super.py report --type profit --date 2023-01-03** en druk op <enter>.

Dit levert: -3.3.

Ook kan de winst (*profit*) van alleen de opgegeven dag opgevraagd worden: type in python **super.py report --type this date profit --date 2023-01-03** en druk op <enter>. Dit levert: -2.5.

Op 04 januari 2023

- Verkocht worden de appels voor een bedrag van 2.5.
- Gekocht worden mandarijnen voor een bedrag van 1.0 en met een uiterste houdbaarheidsdatum 2023-01-11

Command line interface (CLI)

- Type in **python super.py shiftdays 1** en druk daarna op <Enter>.
- Je kan controleren dat de datum inderdaad is aangepast. Type in **python super.py date** of **python super.py report --type csv --csv superpy.csv** en druk op <enter>.
- Type in **python super.py sell apple 2.5** en druk daarna op <Enter>
- Type in **python super.py buy orange 1.9 2023-01-11** en druk daarna op <Enter>

Inhoud cvs-bestanden

Je kan controleren dat de csv-bestanden correct zijn aangepast. Type in **python super.py report --type csv --csv all** om alle bestanden te tonen.

SuperPy.csv	Bought.csv					Sold.csv				Inventory.csv			Expired.csv		
my_today	bought_id	product_name	buy_date	buy_price	expiration_date	sold_id	bought_id	sell_date	sell_price	date	bought_id	product_name	date	bought_id	buy_price
2023-01-04	1	orange	2023-01-02	0.8	2023-01-09	1	1	2023-01-03	2	2023-01-04	3	banana			
	2	apple	2023-01-03	2	2023-01-13	2	2	2023-01-04	2.5	2023-01-04	4	orange			
	3	banana	2023-01-03	2.5	2023-01-05										
	4	orange	2023-01-04	1	2023-01-11										

Revenue en Profit

De omzet (*revenue*) t.m. vandaag is op te vragen: type in **python super.py report --type revenue --date 2023-01-04** en druk op <enter>.

Dit levert: 4.5.

Ook kan de omzet (*revenue*) van alleen de opgegeven dag opgevraagd worden: type in **python super.py report --type this date revenue --date 2023-01-04** en druk op <enter>.

Dit levert: 2.5.

De winst (*profit*) t.m. vandaag is op te vragen: type in **python super.py report --type profit --date 2023-01-04** en druk op <enter>.

Dit levert: -2.7.

Ook kan de winst (*profit*) van alleen de opgegeven dag opgevraagd worden: type in python **super.py report --type this date profit --date 2023-01-04** en druk op <enter>.

Dit levert: 0.6.

Op 05 januari 2023

- Vandaag geen in- of verkoop, maar een andere gebeurtenis: de bananen hebben hun uiterste verkoopdatum (*expiration_date*) bereikt en mogen vandaag niet meer verkocht worden.

Dat heeft geen impact op Bought.csv, Sold.csv of de opbrengst uit verkoop (*revenue*), want gekocht of verkocht wordt er niets.

Het heeft wel impact op

- de voorraad (*inventory*), want de verlopen partij bananen komt daar niet meer op voor. Die partij wordt opgenomen in het nieuwe bestand Expired.csv.
- de winst (*profit*), want het bedrag dat betaald wordt voor de bananen wordt van de winst afgetrokken.

Command line interface (CLI)

- Type in **python super.py shiftdays 1** en druk daarna op <Enter>.
- Je kan controleren dat de datum inderdaad is aangepast. Type in **python super.py date** of **python super.py report --type csv --csv superpy.csv** en druk op <enter>.
- Je kan controleren dat nu ook het bestand Expired.csv een gegevensregel bevat en dat de csv-bestanden correct zijn aangepast. Type in **python super.py report --type csv --csv all** om alle bestanden te tonen.

Inhoud cvs-bestanden

Je kan controleren dat de csv-bestanden correct zijn aangepast. Type in **python super.py report --type csv --csv all** om alle bestanden te tonen.

[illegible]

Revenue en Profit

De omzet (*revenue*) t.m. vandaag is op te vragen: type in `python super.py report --type revenue --date 2023-01-05` en druk op <enter>.

Dit levert: 4.5.

Ook kan de omzet (*revenue*) van alleen de opgegeven dag opgevraagd worden: type in `python super.py report --type this_date revenue --date 2023-01-05` en druk op <enter>.

Dit levert: 0.

De winst (*profit*) t.m. vandaag is op te vragen: type in `python super.py report --type profit --date 2023-01-05` en druk op <enter>.

Dit levert: -5.2 .

Ook kan de winst (*profit*) van alleen de opgegeven dag opgevraagd worden: type in python `super.py report --type this_date profit --date 2023-01-05` en druk op <enter>.

Dit levert: 0..