

# ETF Portfolio Calculator Ohjeet

## Työn aihe ja kuvaus

Excelin kautta valitaan ETF:iä, joista Python kokoaa erilaisia portfoliotyyppejä ja tulostaa ne Excel-tiedostoon. Pythonin luomat portfoliot ovat minimivarianssiportfolio sekä Sharpen luvun maksimoiva portfolio.

## Työn ratkaisuperiaate

Ohjelman ydintoiminnot liittyvät salkun optimointiin ja analysointiin. Se integroi Pythonissa toteutetun hintadatan analysoinnin Exceliin interaktiivisen prosessin luomiseksi. Käyttäjä voi syöttää Exceliin osakkeiden tikettejä, ja ohjelma suorittaa monimutkaisia laskelmia ja visualisointeja päivittämällä Excel-työkirjaa taloudellisilla mittareilla ja graafisilla esityksillä. Ohjelma on rakenteeltaan modulaarinen ja joustava, joten sen toimintoja on helppo muuttaa ja laajentaa.

## Työn rakenne: miten se on jaettu funktioihin

Suurin osa koodin toteuttavasta osasta on jaettu funktioihin. Datan lataaminen ja manipulointi on kaikki toteutettu eri funktioiden sisällä. Tietojen liittäminen Exceliin on toteutettu xlwingsin komentojen avulla.

## Mahdollisten ulkoisten kirjastojen käyttö

```
import xlwings as xw #xlwings excel tiedoston manipulointiin
import pandas as pd #datamanipulointiin dataframejen avulla
import yfinance as yf #datan lataamiseen yahoo fincesta
import warnings #piilottaa varoitukset
import matplotlib.pyplot as plt #datan kuvantamiseen
import numpy as np #datan manipulointiin
import scipy.optimize as sco #portfolion optimointiin
```

Jotta paketit toimivat, olemme huomanneet, että tulisi paketteja ladata seuraavanlaisesti:

```
###Pip install copy-paste:
pip install "xlwings[all]"
pip install openpyxl
pip install pandas
pip install yfinance
pip install warnings
pip install matplotlib
pip install scipy
```

## Käytetyt funktiot ja niiden keskinäiset suhteet

Seuraavaksi esittelemme pakettikohtaisesti käytettyjä funktiota, koska funktioita, joita käytämme, on niin monta ja omien funktioidemme käyttö on niistä vahvasti riippuvaisia.

### xlwings

Käytimme pakettia hallitaksemme exceliä, käytännössä solujen manipulointiin ja datan tulostamiseen.

Book()                                   #importataan excel tiedosto

sheets()                                #importataan välilehdet

<code>clear.content()</code>	<code>#poistetaan arvoja (ei formatointia)</code>
<code>picture.add()</code>	<code>#lisätään exceliin kuva (graafi ohjelman tapauksessa)</code>
<code>pandas</code>	
<code>Dataframe()</code>	<code>#luo 'dataframe', eli muuttujan, jonka luokka on dataframe</code>
<code>Series()</code>	<code>#Määritetään series muuttujaksi (yksiulotteinen)</code>
<code>Concat()</code>	<code>#käytetään pandas-objektien yhdistämiseen, joko horisontaalilla tai vertikaalilla akselilla</code>
<code>read.excel()</code>	<code>#luetaan excel tiedosto</code>
<code>yfinance</code>	
<code>download()</code>	<code>#ladataan tietoa tietyllä 'tickerillä'</code>
<code>warnings</code>	
<code>catch_warnings()</code>	<code>#käytetään varoitusten hallintaan</code>
<code>warnings.simplefilter()</code>	<code>#asettaa varoitukselle suodatussäännön</code>
<code>matplotlib</code>	
<code>figure()</code>	<code>#luo uuden graafin</code>
<code>plot()</code>	<code>#viivakaavio</code>
<code>title()</code>	<code>#graafin otsikko</code>
<code>xlabel()</code>	<code>#x-akselin otsikko</code>
<code>ylabel()</code>	<code>#y-akselin otsikko</code>
<code>legend()</code>	<code>#luo legendin/selitteen mitä dataa kuvaajassa on</code>
<code>gcf()</code>	<code>#palauttaa nykyisen kuvion, käytetään kuvion muokkaamiseen</code>
<code>numpy</code>	
<code>dot()</code>	<code>#laskee pistetulon</code>
<code>linspace()</code>	<code>#luo taulukon halutusta alkuarvosta loppuarvoon määritellyllä arvojen lukumäärällä</code>
<code>argmin()</code>	<code>#palauttaa pienimmän arvon indeksin x-akselilla</code>
<code>sum()</code>	<code>#laskee solujen summan</code>
<code>sqrt()</code>	<code>#laskee solujen neliöjuuren</code>
<code>scipy</code>	
<code>minimize()</code>	<code>#minimoi funktion (optimoi)</code>
<code>itse määritellyt funktiot</code>	
<code>reset_worksheet_dashboard()</code>	<code>#tyhjentää etukäteen merkatut solut Excelissä</code>
<code>download_data()</code>	<code>#lataa ETF:ien hintadatan yahoo financesta</code>
<code>plottaus()</code>	<code>#piirtää inputtina saadut tuottoluvut</code>

<code>equal_weight_returns()</code>	<code>#laskee tasapainotetun portfolion tuottoluvut</code>
<code>get_portf_rtn()</code>	<code>#laskee portfolion tuottoluvut</code>
<code>form_min_var_portfolio()</code> -funktioon	<code>#palauttaa minimivarianssiportfolion painot, viittaa "get_efficient_frontier"</code>
<code>form_max_sharpe_portfolio()</code>	<code>#palauttaa Sharpen luvun maksimoivan portfolion painot</code>
<code>get_efficient_frontier()</code>	<code>#käytetään minimivarianssiportfolion painojen laskemisessa</code>
<code>neg_sharpe_ratio()</code>	<code>#laskee sharpe ration max sharpe ration "form_max_sharpe_portfolio" - funktioille</code>
<code>get_portf_vol()</code>	<code>#laskee portfolion volatiliteetin</code>
<code>hintakaavio()</code>	<code>#plottaa Max Sharpe Portfolion weights muuttujasta</code>
<code>compare_portfolios()</code>	<code>#plottaa kahden portfolion tuotot samaan graafiin, joka myöhemmin printataan exceliin</code>
<code>plot_return_histogram()</code>	<code>#plottaa histogrammin tuotoista, joka myöhemmin printataan exceliin</code>
<code>get_combined_returns()</code>	<code>#luo dataframen laskettujen portfolioiden ja yksittäisten tickereiden tuotoista, joita käytetään calculate_metrics() ja täten print_to_excel() funktioissa</code>
<code>calculate_metrics()</code>	<code>#laskee metriikat, joita printtaamme käyttäjälle exceliin</code>
<code>print_to_excel()</code>	<code>#printtaa portfolioiden ja tickereiden statistisia tietoja exceliin, jotka saadaan calculate_metrics() funktiosta</code>

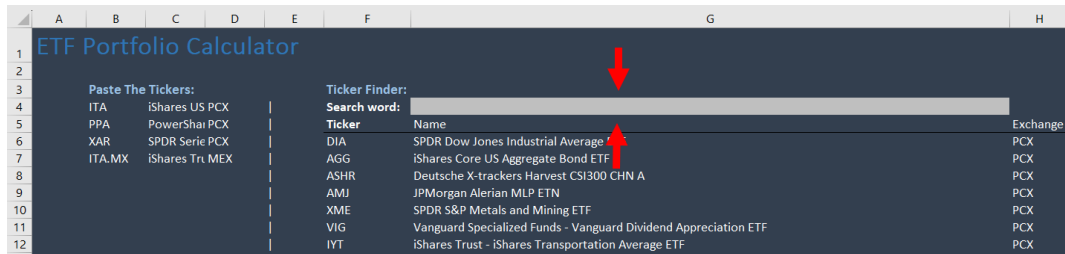
### Vastuualueiden jakautuminen, jos työ on parityö

Lassi vastasi Excelin ja Pythonin välisestä kommunikaatiosta. Varmisti datan laadun (alkuperäisessä datassa n. 22 000 ETF).

Otto vastasi datan keräämisestä sekä portfolioiden luomisesta ja optimoinnista.

## Ohjeet ohjelman käyttämiseksi

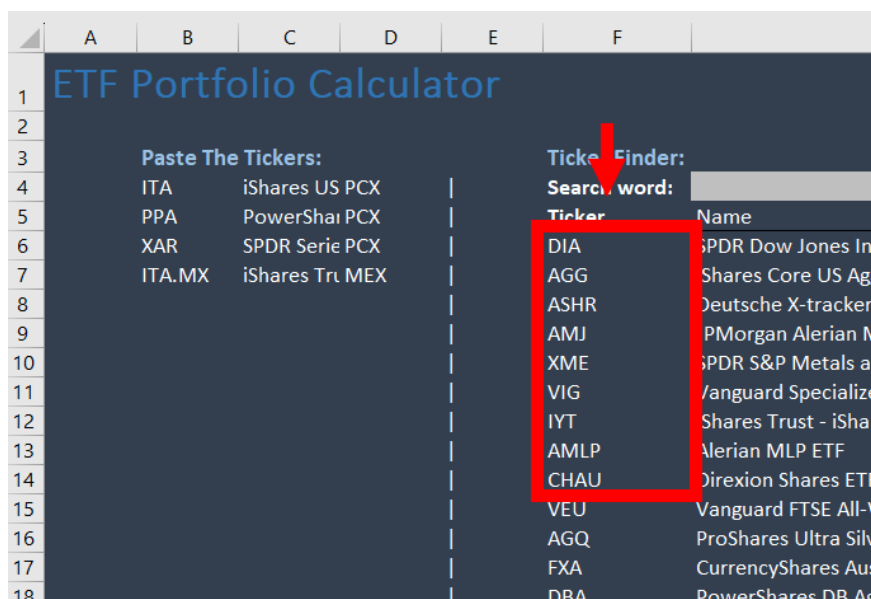
1. Koodin ajamista varten avaa Excel-tiedosto (Main\_i.xlsx). Excel-tiedoston solusta G4 (Dashboard-välilehti) on hakupalkki, jonka avulla voidaan hakea ETF:iä, jotka ovat olleet Yahoo Financen tiedoissa vuoden 2017 syyskuussa. Kirjoita soluun haluamasi ETF:n nimi (tai osa siitä)



The screenshot shows the 'ETF Portfolio Calculator' interface. On the left, under 'Paste The Tickers:', there is a list of tickers: ITA, PPA, XAR, and ITA.MX. On the right, under 'Ticker Finder:', there is a search bar and a table of results. A red arrow points to the search bar, and another red arrow points to the first row of the table.

Ticker	Name	Exchange
DIA	SPDR Dow Jones Industrial Average	PCX
AGG	iShares Core US Aggregate Bond ETF	PCX
ASHR	Deutsche X-trackers Harvest CSI300 CHN A	PCX
AMJ	JPMorgan Alerian MLP ETN	PCX
XME	SPDR S&P Metals and Mining ETF	PCX
VIG	Vanguard Specialized Funds - Vanguard Dividend Appreciation ETF	PCX
IYT	iShares Trust - iShares Transportation Average ETF	PCX

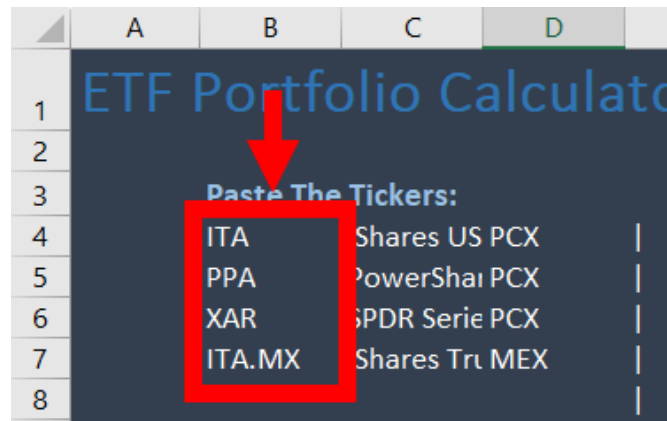
2. Kun olet löytänyt haluamasi indeksit, kopioi niiden ticker F-sarakkeesta. Voit myös kopioida ja liittää Ticker Finderista kaikki kolme saraketta kuten esimerkissä.



The screenshot shows the 'ETF Portfolio Calculator' interface. On the left, under 'Paste The Tickers:', there is a list of tickers: ITA, PPA, XAR, and ITA.MX. On the right, under 'Ticker Finder:', there is a search bar and a table of results. A red box highlights the 'Ticker' column of the table.

Ticker	Name	Exchange
DIA	SPDR Dow Jones Industrial Average	PCX
AGG	iShares Core US Aggregate Bond ETF	PCX
ASHR	Deutsche X-trackers Harvest CSI300 CHN A	PCX
AMJ	JPMorgan Alerian MLP ETN	PCX
XME	SPDR S&P Metals and Mining ETF	PCX
VIG	Vanguard Specialized Funds - Vanguard Dividend Appreciation ETF	PCX
IYT	iShares Trust - iShares Transportation Average ETF	PCX
AMLP	Alerian MLP ETF	PCX
CHAU	Direxion Shares ETF	PCX
VEU	Vanguard FTSE All-World	PCX
AGQ	ProShares Ultra Silver	PCX
FXA	CurrencyShares Australia	PCX
DBA	PowerShares DB Agriculture	PCX

3. Liitä kopioimasi tickerit B-sarakkeeseen 4. rivistä alkaen alaspäin, valitse vähintään kaksi tickeriä. Emme suosittele valitsemaan liian montaa tickeriä, koska optimointiprosessi voi muuttua hyvin raskaaksi tickereiden määrän kasvaessa. Kannattaa testata aluksi pienemmällä määrällä ja tarkastella suoritusajan kehitystä. Esimerkiksi 20 menee edelleen varsin mainiosti.



The screenshot shows the 'ETF Portfolio Calculator' interface. On the left, under 'Paste The Tickers:', there is a list of tickers: ITA, PPA, XAR, and ITA.MX. On the right, under 'Ticker Finder:', there is a search bar and a table of results. A red box highlights the 'Ticker' column of the table.

Ticker	Name	Exchange
DIA	SPDR Dow Jones Industrial Average	PCX
AGG	iShares Core US Aggregate Bond ETF	PCX
ASHR	Deutsche X-trackers Harvest CSI300 CHN A	PCX
AMJ	JPMorgan Alerian MLP ETN	PCX
XME	SPDR S&P Metals and Mining ETF	PCX
VIG	Vanguard Specialized Funds - Vanguard Dividend Appreciation ETF	PCX
IYT	iShares Trust - iShares Transportation Average ETF	PCX
AMLP	Alerian MLP ETF	PCX
CHAU	Direxion Shares ETF	PCX
VEU	Vanguard FTSE All-World	PCX
AGQ	ProShares Ultra Silver	PCX
FXA	CurrencyShares Australia	PCX
DBA	PowerShares DB Agriculture	PCX

4. Kun olet liittänyt haluamasi tickerit, tallenna Excel tiedosto. Nyt voit ajaa koodin Pythonin kautta. Tämän jälkeen sinun ei tarvitse enää tehdä mitään.

Koodi laskee Pythonin puolella optimaaliset portfoliot varianssin minimoimiseksi ja Sharpen luvun maksimoimiseksi. Portfolioiden kehityksestä liitetään kuvat sarakkeesta P alkaen, kuvia tulee kolme kappaletta. Portfolioista (ja yksittäisistä tickereistä) liitetään tuottolukuja, varianssilukuja sekä Sharpen luku sarakkeiden J ja M välille.

ChatGPT linkkejä:

<https://chat.openai.com/share/ef9419b5-5422-4822-a088-bd7a941dab50>

<https://chat.openai.com/share/23718651-8bff-419d-84c1-78fe139f67bb>

<https://chat.openai.com/share/8ae1d006-a167-44a3-92d1-9c30eeb20e4d>

<https://chat.openai.com/share/b5ac3ce8-f809-4815-8c24-7547fa6ddda2>