

# CPF: A New Era

Become a Super Quant with Reproducible, Production-Grade Skills

Dr. Yves J. Hilpisch

The Python Quants GmbH

CPF Program



# The World Our Programs Prepare You For

## On the one hand

- Darkening outlook for many white-collar roles as automation pressure rises.
- Entry-level and junior opportunities shrinking in traditional analyst and developer tracks.
- Rapid advances in GenAI, agents, and automation compressing mid-skill tasks into software.

## On the other hand

- Companies investing billions in AI infrastructure, data platforms, and crypto rails.
- Experienced professionals with production-grade skills see outsized productivity and compensation.
- As capital spending on technology grows, true experts capture a disproportionate share of the value.

## What this means for you

The gap between generic credentials and real capabilities is widening: recent “safe bets” such as CS or finance degrees no longer guarantee durable opportunities when AI innovation cycles outpace academic cycles. The practical way to stay on the right side of this shift is focused upskilling and mastery of key technologies like Python, AI, and crypto — the core dimensions our programs are built around.

# How This Session Works

- You will learn about the specifics and common features of the CPF, TAE, and TCE programs.
- The focus is on what you get, who benefits most, and how to choose the right program (or bundle).
- The session highlights the fully revamped Core Track of the CPF and what our programs share in terms of approach and learning experience.

## Our Programs Are for You ...

... if you are excited about quantitative finance and algorithmic trading, enthusiastic about the potential of AI, and curious about crypto technologies — and want a coherent path that links CPF, TAE, and TCE.

# Agenda

- 1 The Python Quants (TPQ)
- 2 Three Programs in Comparison
- 3 Certificate in Python for Finance (CPF)
- 4 The AI Engineer (TAE)
- 5 The Crypto Engineer (TCE)
- 6 Common Features (Across Programs)

# The Python Quants (TPQ)

# The Python Quants

The Python Quants GmbH - tpq.io

THE PYTHON QUANTS  
Python - AI - Crypto - Quant Finance

PROGRAMS COMPANY ABOUT YVES CPF TAE TCE CHOOSE YOUR TRACK

THE PYTHON QUANTS GMBH

# Python, AI, Crypto and quant finance – under one roof.

The Python Quants GmbH, founded and led by **Dr. Yves J. Hilpisch**, is the home of high-impact programs at the intersection of Python, AI, Crypto and quantitative finance.

DR. YVES J. HILPISCH – FOUNDER & CEO – PROGRAM DIRECTOR OF CPF, THE AI ENGINEER & THE CRYPTO ENGINEER

Explore the three programs Learn more about Yves

Ask anything!

THE PYTHON QUANTS  
PYTHON AND AI FOR FINANCE

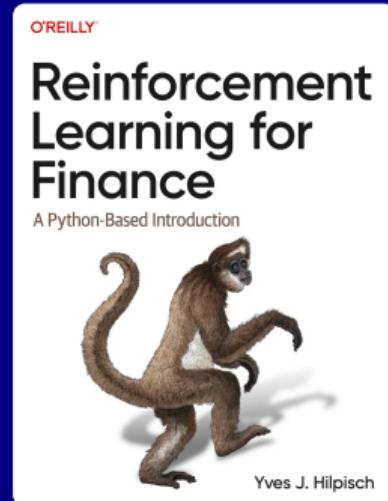
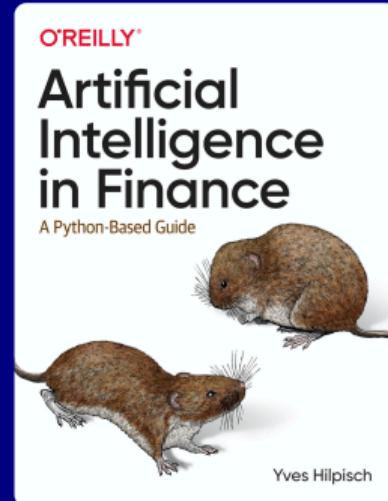
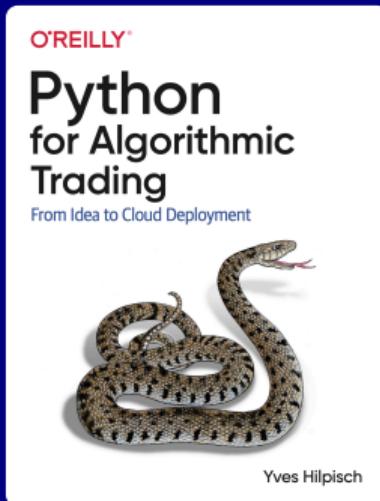
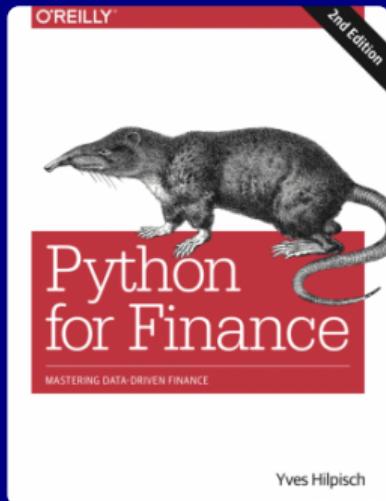
The Python Quants GmbH  
Independent, practice-first programs and platforms built around modern Python, applied AI, Crypto, and real quantitative finance.

<https://tpq.io>

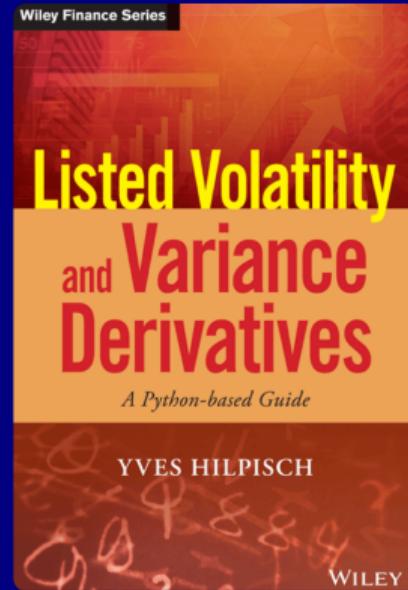
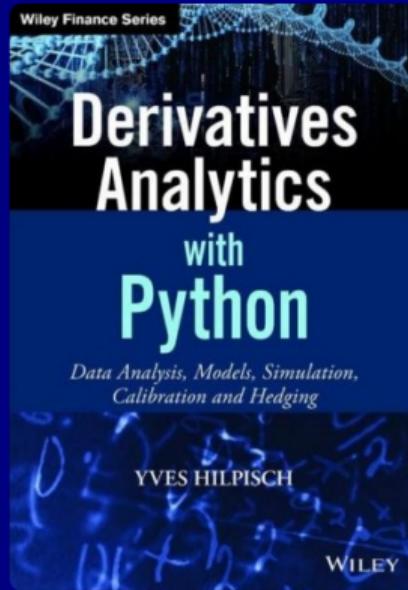
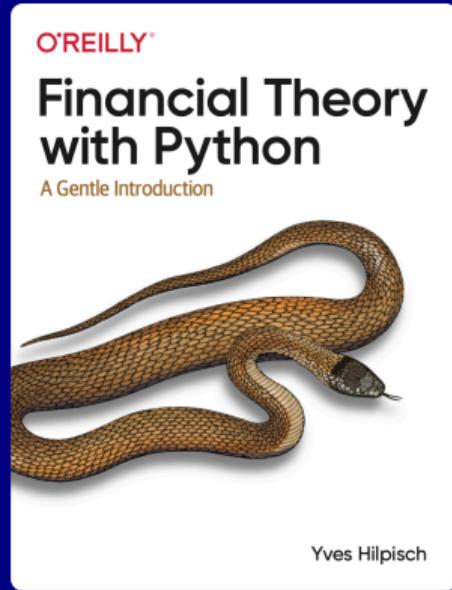
# The Python Quants in 60 Seconds

- Independent, practice-first education company led by Dr. Yves J. Hilpisch.
- Three coherent programs: CPF, TAE, TCE (plus bundle option).
- A single learning philosophy: books as the basis, guided practice, real code, real workflows.
- 7 Python & AI for Finance books published (5x O'Reilly, 2x Wiley).
- 1,000s of academics & professionals trained live and online.
- Corporate training for leading institutions (Citadel, Two Sigma, PDT Partners, and more).
- 20 years of Python & AI for quantitative finance.

# Python & AI for Finance Books (I)



# Python & AI for Finance Books (II)



# Three Programs in Comparison

# Three Programs in Comparison

The Python Quants GmbH - tpq.io

PROGRAMS COMPANY ABOUT YVES CPF TAE TCE CHOOSE YOUR TRACK

## Your launchpad: three focused programs

Select the program that best aligns with your next step - from deepening core Python-for-finance skills to building full AI systems or mastering Crypto engineering. All tracks are designed and directed by Dr. Yves J. Hilpisch.

After enrollment, platform login credentials are sent shortly after payment (at most one German business day later).

**CPF : CERTIFICATE IN PYTHON FOR FINANCE** Flagship

**Become a Super Quant**

Master Python, finance, and AI in one coherent, practice-driven curriculum.

400+ hours of video, 3,000+ pages, 600+ notebooks and capstone research - all integrated on the [Quant Platform](#).

[Explore the CPF Program](#)

**TAE : THE AI ENGINEER**

**Production-grade AI systems**

From Math and PyTorch to building LLMs from scratch - a portfolio-first AI engineering program.

Design, build, and ship end-to-end AI engineering projects and applications with a focus on **real deployments**, not isolated code snippets.

[Explore The AI Engineer](#)

**TCE : THE CRYPTO ENGINEER**

**Confident Crypto Operations**

Understand custody, markets, monitoring and risk through runnable labs and operational runbooks.

Four concise volumes, hands-on labs and a **practical, operational** approach to **Crypto engineering** for institutional markets.

[Explore The Crypto Engineer](#)

Ask anything!

<https://tpq.io>

# CPF vs TAE vs TCE (High-Level Comparison)

*“Professional Processes and Best Practices at Every Step.”*

	CPF	TAE	TCE
Outcome	Become a Super Quant (Python + finance + AI)	Build production-grade AI systems (LLMs, agents)	Engineer crypto systems with confidence (custody, markets, ops)
Best for	Quants, finance professionals, analysts, career changers	Engineers/ML practitioners building real AI apps	Builders, risk/ops leads, quants touching crypto rails
Core coverage	Quant research + data + pricing + investing + trading + applied AI	PyTorch, LLMs, agents, deployment patterns	Primitives, Bitcoin, market plumbing, monitoring, runbooks

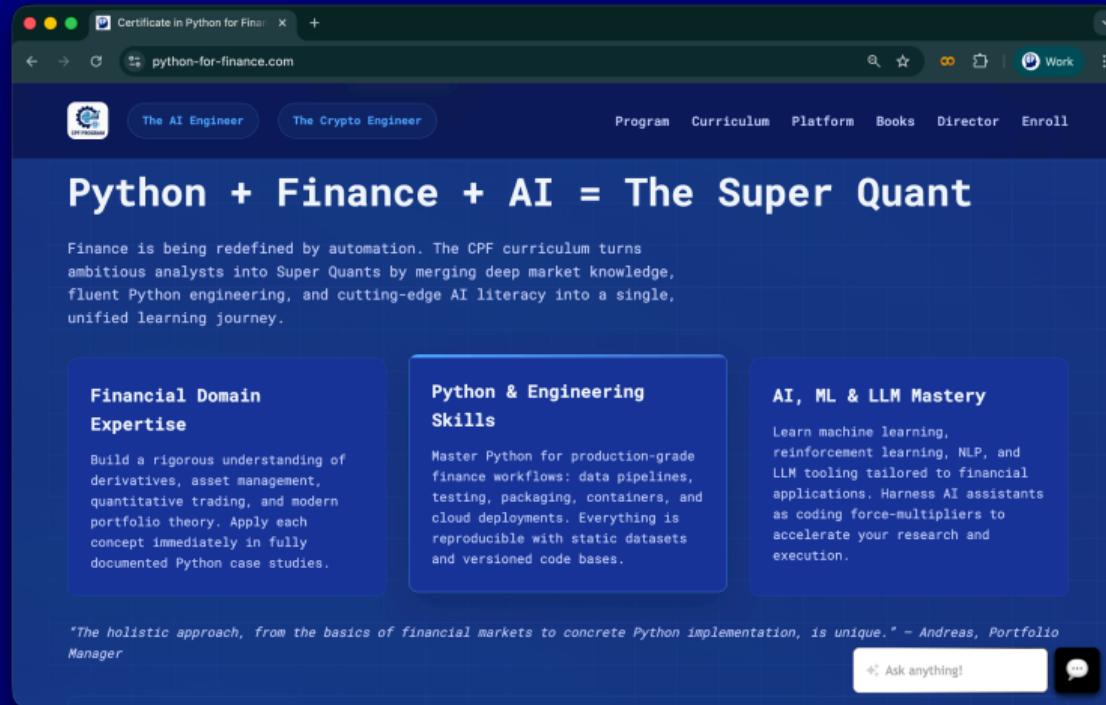
# CPF vs TAE vs TCE: Hard Facts

	CPF	TAE	TCE
Books/key PDFs (count)	7 published + 2 unpublished books	8+ custom textbooks and docs	4 books + 17 notes PDFs
Docs/books pages (approx.)	3,000+ pages across books and program PDFs	1,250+ pages of engineering docs and textbooks	341+ book & 100+ notes pages
Jupyter notebooks	600+ notebooks integrated with the CPF curriculum	100+ Colab-ready notebooks for AI engineering	51 notebooks tightly linked to the TCE books
Code/scripts	Full research and trading code embedded in 600+ notebooks	300+ Python scripts for models, agents, and tooling	30 scripts plus runnable labs for custody, markets, and monitoring

CPF core track (fully revamped): 61 PDFs, 2,903 pages.  
Text: 29 files, 1,968 pages; Slides: 32 files, 935 pages.

# Certificate in Python for Finance (CPF)

# Certificate in Python for Finance (CPF)



The screenshot shows the homepage of the Python for Finance (CPF) website. The URL in the browser is [python-for-finance.com](https://python-for-finance.com). The page features a dark blue header with the CPF logo and navigation links for Program, Curriculum, Platform, Books, Director, and Enroll. Below the header, a large banner with the text "Python + Finance + AI = The Super Quant". A subtext below the banner reads: "Finance is being redefined by automation. The CPF curriculum turns ambitious analysts into Super Quants by merging deep market knowledge, fluent Python engineering, and cutting-edge AI literacy into a single, unified learning journey." The page is divided into three main sections: "Financial Domain Expertise", "Python & Engineering Skills", and "AI, ML & LLM Mastery", each with a brief description and a callout to action.

**Financial Domain Expertise**

Build a rigorous understanding of derivatives, asset management, quantitative trading, and modern portfolio theory. Apply each concept immediately in fully documented Python case studies.

**Python & Engineering Skills**

Master Python for production-grade finance workflows: data pipelines, testing, packaging, containers, and cloud deployments. Everything is reproducible with static datasets and versioned code bases.

**AI, ML & LLM Mastery**

Learn machine learning, reinforcement learning, NLP, and LLM tooling tailored to financial applications. Harness AI assistants as coding force-multipliers to accelerate your research and execution.

*"The holistic approach, from the basics of financial markets to concrete Python implementation, is unique." — Andreas, Portfolio Manager*

Ask anything!

# CPF: Content, Target Groups, Benefits

This is the flagship program for quantitative finance with Python — designed to turn motivated learners into confident practitioners with reproducible, production-grade workflows.

## What it covers

- Python for finance workflows, data engineering, reproducible research, applied AI in finance.
- Focus on Financial Theory, Asset Management, Algorithmic Trading, and Computational Finance.

## Who benefits most

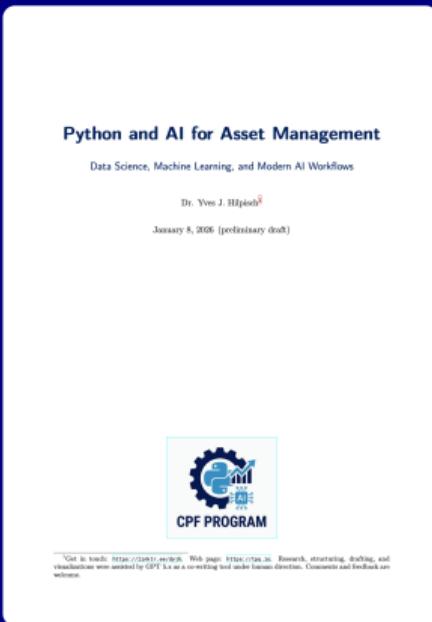
- Finance professionals, quants, analysts, researchers, lecturers, career changers, students.

## Why it matters now

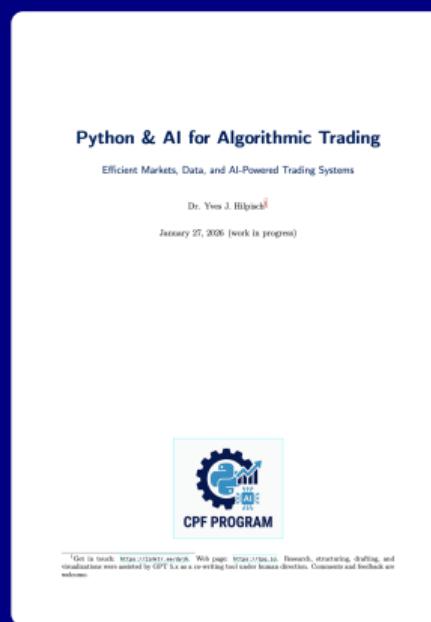
- Quantitative finance is increasingly based on Python, driven by AI, and influenced by crypto — CPF delivers on all three dimensions.
- The GenAI era rewards teams that can combine domain depth with reproducible, production-grade execution.

# CPF: New Books (200+ Pages Each)

Brand-new resources for the GenAI era in quantitative finance.



\*Get in touch: <https://1234567890.chat>. Web page: <https://1234567890.ai>. Research, structuring, drafting, and visualizations were assisted by GPT-3.5 as a co-writing tool under human direction. Comments and feedback are welcome.



\*Get in touch: <https://1234567890.chat>. Web page: <https://1234567890.ai>. Research, structuring, drafting, and visualizations were assisted by GPT-3.5 as a co-writing tool under human direction. Comments and feedback are welcome.

# CPF: New Options Simulator Apps

The Python Quants  
Derivatives Analytics Basics  
Interactive option apps for intuition and practice

These three apps form a compact, self-contained toolkit by The Python Quants for exploring classical option pricing and risk analysis in depth. They share a common visual language and let you move smoothly from discrete binomial trees, to closed-form Black-Scholes-Merton pricing, to full GBM path simulation and hedging.

**STEP 1 - DISCRETE INTUITION**  
**Binomial Option Tree** [tree.html](#)  
CRR binomial tree for equity options with European/American exercise and node-wise hedging/replication.

**WHAT IT DOES**

- Builds a CRR tree for  $S$  with user-defined steps,  $\delta, r, q$ .
- Shows option values at each node and compares tree price to BSM.
- Replication mode displays  $\Delta$  and bond  $B$  per node.

**YOU PRACTISE**

- Deriving risk-neutral probabilities and discounting through the tree.
- Understanding  $\Delta/\theta$  hedging and early exercise for American options.
- Seeing how binomial prices converge to Black-Scholes-Merton as  $N$  increases.

**STEP 2 - CLOSED-FORM PRICING**  
**BSM Option Pricer** [pricer.html](#)  
Single-option Black-Scholes-Merton pricer with continuous dividends, disturbance sliders, Greeks, and sensitivity views.

**WHAT IT DOES**

- Pricers European calls/puts under BSM with dividend yield  $q$ .
- Shows  $\Delta, \Gamma, \text{Vega}(1\%), \Theta(\text{day}), \text{p}(1\%)$  plus intrinsic/time value and moneyness.
- Lets you disturb base parameters and see the impact via a Greek bar chart and sensitivity curve.

**YOU PRACTISE**

- Linking each parameter ( $S, K, T, \sigma, r, q$ ) to price and Greeks.
- Interpreting  $\Delta, \Gamma, \text{Vega}, \Theta, \text{p}$  numerically and visually.
- Relating BSM pricing to arbitrage bounds and intrinsic vs time value.

**STEP 3 - DYNAMIC RISK**  
**BSM Path & Risk Simulator** [simulator.html](#)  
GBM spot path simulator with live BSM pricing, Greeks, and unhedged vs delta-hedged P&L over time.

**WHAT IT DOES**

- Simulates GBM paths under risk-neutral or real-world drift.
- Re-prices the option at each step and tracks Greeks along the path.
- Implements discrete  $\Delta$ -hedging with transaction costs and shows hedged vs unhedged P&L.

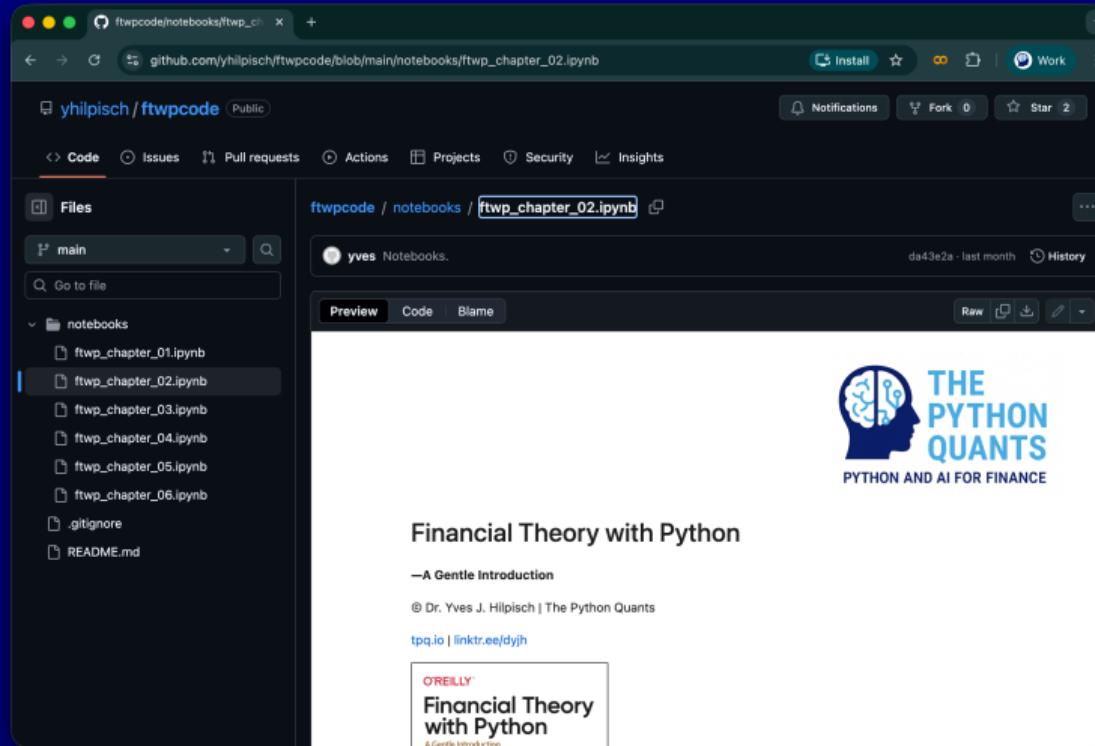
**YOU PRACTISE**

- Connecting static Greeks to dynamic P&L and path behaviour.
- Exploring hedge frequency, volatility levels, and rate assumptions.
- Discussing gamma/vega risk,  $\Theta$  vs realized variance, and hedging quality.

BSM with continuous dividend yield  $q$ . Educational analytics apps by The Python Quants.  
For educational use only; no investment advice and no warranty is given as to completeness, correctness, or fitness for any purpose.

One design - three apps - tree → pricer → path simulator.

# CPF: Sample GitHub Repository



<https://github.com/yhilpisch/ftwpcodes>

# The AI Engineer (TAE)

# The AI Engineer (TAE)

The screenshot shows the homepage of [theaiengineer.dev](https://theaiengineer.dev). The top navigation bar includes links for 'Program', 'Curriculum', 'Books', 'Features', 'Talent War', and 'Enroll'. The main content area is divided into two tracks: 'Core Track' and 'Engineering Track', each with four numbered modules.

**Core Track**

- 1 Python & Math for ML/DL**  
Master vectorization, linear algebra, calculus essentials, and NumPy for machine learning.
- 2 Deep Learning with PyTorch**  
Build and train neural networks, implement training loops, schedulers, and optimization.
- 3 LLM from Scratch**  
Implement tokenizers, attention mechanisms, and train your own GPT-like model.
- 4 AI Agents & Automation**  
Build tool-using agents with planning, guardrails, and production cost control.

**Engineering Track**

- 1 Foundation Primers**  
Python best practices, mathematical foundations, and development environment setup.
- 2 Software Engineering**  
Testing, packaging, CLI tools, CI/CD pipelines, and containerization.
- 3 ML Engineering**  
Data management, experiment tracking, metrics, evaluations, and reproducibility.
- 4 AI Engineering**  
Model serving, latency optimization, observability, safety, and privacy in production.

At the bottom right, there is a 'Ask anything!' button with a speech bubble icon.

# TAE: Content, Target Groups, Benefits

This program is the fastest path from foundations to shipping: you build real AI systems (LLMs, agents, tooling) with an engineering-first, portfolio-driven approach.

## What it covers

- Math + PyTorch foundations, LLMs, agents, and production-grade AI engineering patterns.
- Focus on Deep Learning, LLM Engineering, AI Agents, and Production AI Systems.

## Who benefits most

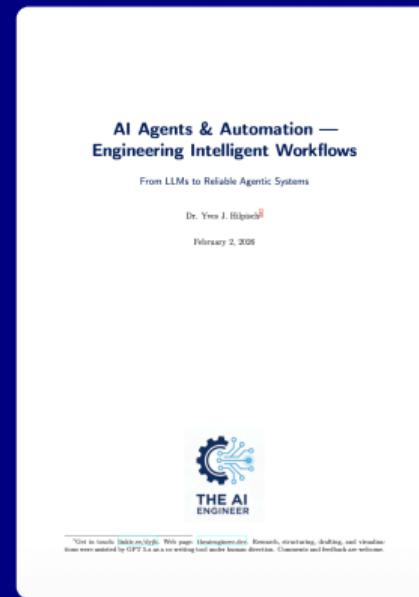
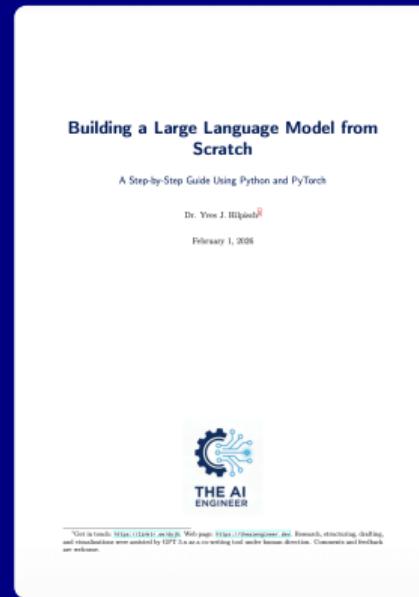
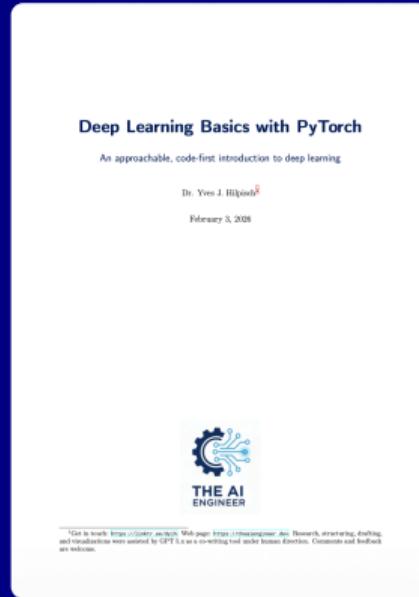
- Engineers and practitioners building real AI systems (including finance/quant contexts).

## Why it matters now

- AI engineering is among the highest-paid skills in 2026 — and demand is still accelerating.
- Shipping AI reliably is an engineering problem; TAE focuses on end-to-end systems and portfolios.

# Books about DL, LLMs, and AI Agents

Three focused books that support deep learning, LLM, and AI agent work across the TAE program (7 books total).



# The Crypto Engineer (TCE)

# The Crypto Engineer (TCE)

The screenshot shows a web browser window for 'The Crypto Engineer - Build C' on 'thecryptoengineer.dev'. The page features a navigation bar with 'Program', 'Curriculum', 'Books', 'Labs', 'Features', and 'Enroll' buttons. Below the navigation is a section titled 'Choose How You Traverse the Spine' with the subtext: 'Each path uses the same references and labs. You decide how much time to spend in drills versus deep reading.' Three program paths are listed in cards:

- Foundations Sprint**  
Fast Path · 4 Weeks
  - Guided read-through of Crypto Foundations & Bitcoin Foundations.
  - Weekly coaching guides (PDFs) point to the notebooks, labs, and priorities for each module, with day-by-day study suggestions.
  - Dedicated Discord server for guidance, content clarifications, and peer support.
  - Perfect for fast-moving security, quant, or infra engineers entering crypto.
- Systems Builder**  
Medium Path · 8 Weeks
  - Alternating book chapters with labs on nodes, wallets, fee policy, and markets.
  - Bi-weekly coaching guides (PDFs) highlight review prompts, lab focus areas, and practice priorities, with day-by-day study suggestions.
  - Asynchronous Discord discussions clarify tricky sections and keep labs on track.
  - Ideal for team leads running trading desks, custody stacks, or risk programs.
- Ops Residency**  
Exploration Path · 6 Months
  - Use the entire repo as an internal training hub for a single seat.
  - Self-paced timeline with detailed coaching guides to structure your practice, with day-by-day study suggestions.
  - Mix-and-match labs and incident walkthroughs to document progress or compliance evidence.
  - Built for busy engineers who need reusable drills over a long runway.

At the bottom right is a 'Ask anything!' button with a speech bubble icon.

<https://thecryptoengineer.dev>

# TCE: Content, Target Groups, Benefits

This program bridges crypto theory to operational confidence: the goal is to help you reason about custody, markets, and live systems under real constraints.

## What it covers

- Crypto primitives, Bitcoin foundations, markets/data plumbing, and production operations (monitoring/runbooks).
- Focus on Custody & Wallet Ops, Bitcoin Infrastructure, Market Data Pipelines, and Monitoring/Runbooks.

## Who benefits most

- Builders, quants, and teams operating crypto rails (custody, markets, risk, compliance interfaces).

## Why it matters now

- Beyond hype, speculation, and scandals, crypto requires production-focused skills for custody, markets, and operations.
- Institutions need operational confidence; TCE bridges theory to incident-ready practice.

# Common Features (Across Programs)

# Common Features

The screenshot shows a web browser window for 'CPF Launchpad' at the URL [tpq.io/cpf/](https://tpq.io/cpf/). The page title is 'Certificate in Python for Finance'. The main content is titled 'Bridge Topics: Covariance, Efficient Markets, FTAP'. It describes three bridge notes that strip central ideas from covariance geometry, efficient markets, and the Fundamental Theorems of Asset Pricing down to their smallest nontrivial setups. It links to a GitHub repository and a companion exercise set. Below this, three cards provide details for each topic:

- Covariance Geometry in a Two-Asset World**: A focused treatment of covariance in a two-asset setting that builds geometric intuition around payoff clouds, covariance matrices, eigenvalues, and portfolio variance; it bridges directly into the risk and diversification perspectives used later in asset management.
  - BRIDGE TOPIC - COVARIANCE
  - Notes: [covariance.pdf](#)
  - Slides: [covariancesl.pdf](#)
  - Recording
- Efficient Markets Hypothesis – Theory and Tests**: A compact note and slide deck on the Efficient Markets Hypothesis that formalize market efficiency as a conditional expectation restriction and walk through core empirical tests relevant for asset management and algorithmic trading.
  - BRIDGE TOPIC - EFFICIENT MARKETS
  - Notes: [markets.pdf](#)
  - Slides: [marketssl.pdf](#)
  - Recording
- Fundamental Theorems of Asset Pricing – Finite-State Markets**: A finite-state, one-period development of the Fundamental Theorems of Asset Pricing that makes no-arbitrage, state prices, martingale measures, and completeness fully explicit in low-dimensional payoff spaces.
  - BRIDGE TOPIC - ASSET PRICING
  - Notes: [ftap.pdf](#)
  - Slides: [ftapsl.pdf](#)
  - Recording

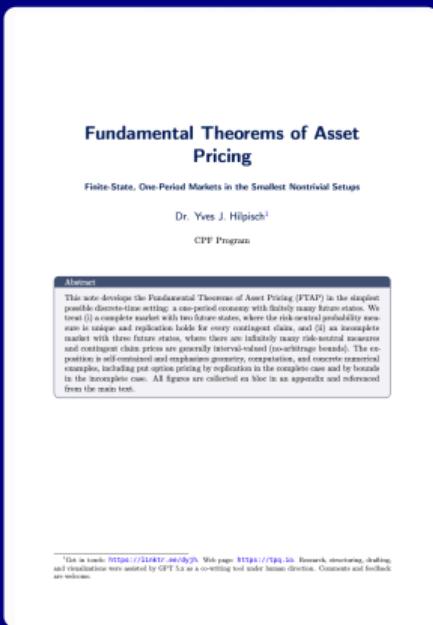
# Common Features (Learning Assets & Guidance)

These features are shared across CPF, TAE, and TCE — they keep learning coherent, fast, and oriented around real outcomes.

- Launchpads (fast onboarding + a single place for resources)
- Books as the basis (coherent reference layer)
- GitHub repositories with Colab-ready notebooks and code
- Recorded sessions (CPF, TAE)
- Slide decks for sessions and high-level summaries
- Weekly/bi-weekly coaching guides (with day-by-day study suggestions)
- Exercises and capstones per class (portfolio + real competence signal)
- Community and support (Discord + clarifications + accountability)

# Detailed Theoretical Notes & Books

For every core class, you get detailed theoretical notes/books that cover the key financial and mathematical concepts beyond the Python and AI coding work.



**Fundamental Theorems of Asset Pricing**  
Finite-State, One-Period Markets in the Smallest Nontrivial Setups  
Dr. Yves J. Hilpisch<sup>1</sup>  
CPT Program  
  
**Abstract**  
This note develops the Fundamental Theorems of Asset Pricing (FTAP) in the simplest possible discrete-time setting: a one-period economy with finitely many future states. We show that (i) the set of all strictly positive state prices is a convex cone, (ii) the no-arbitrage cone is unique and replication holds for every contingent claim, and (iii) an incomplete market with three future states, where there are infinitely many risk-neutral measures and contingent claim prices are generally interval-valued (no-cashflows bound). The exposition is based on the book *CPF: A New Era* (Hilpisch 2023) and includes several examples, including put option pricing by replication in the complete case and by bounds in the incomplete case. All figures are collected en bloc in an appendix and referenced from the main text.

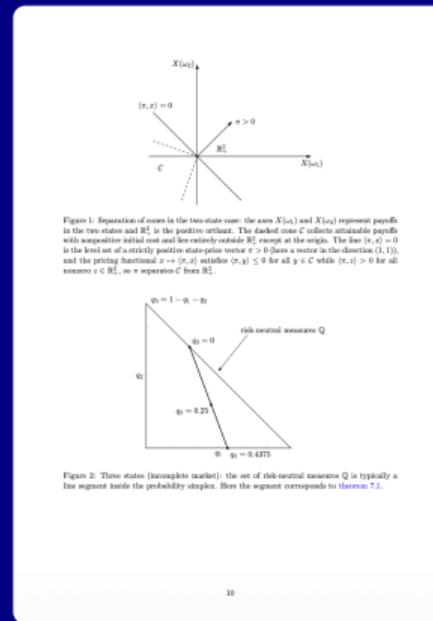


Figure 1: Separation of cones in the two-state case: the axes  $X(\omega_1)$  and  $X(\omega_2)$  represent portfolios in the two states and  $\mathbb{R}^2_+$  is the positive orthant. The dashed cone  $\mathcal{C}$  collects attainable portfolios with nonpositive initial cost and lies entirely outside  $\mathbb{R}^2_+$  except at the origin. The line  $(x, x) = 0$  is the level set of a strictly positive state-price vector  $x > 0$  (thus a vector in the direction  $(1, 1)$ ), and the pricing functional  $\pi \mapsto \langle x, \pi \rangle$  satisfies  $\langle x, \pi \rangle \leq 0$  for all  $y \in \mathcal{C}$  while  $\langle x, \pi \rangle > 0$  for all nonzero  $\pi \in \mathbb{R}^2_+$ , as  $x$  separates  $\mathcal{C}$  from  $\mathbb{R}^2_+$ .

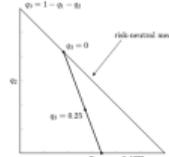


Figure 2: Three-state (incomplete market): the set of risk-neutral measures  $Q$  is typically a line segment inside the probability simplex. Here the segment corresponds to [theorem 7.1](#).

# Comprehensive Exercise Sets

Every core class comes with comprehensive exercise sets for nearly every chapter/topic: review questions, standard exercises, advanced exercises, capstone tasks, and interview-style questions.

## Bridge Week Exercises

CPF Program  
February 2, 2026

**Abstract**  
This exercise set supports the Certificate in Python for Finance (CPF) bridge week. It covers covariance geometry, the Efficient Markets Hypothesis (EMH), and the Fundamental Theorem of Asset Pricing (FTAP). Each topic provides five exercise types – review questions, exercises, advanced exercises, a capstone project, and interview-style questions—so you can scale the depth of your practice to the available time.



**Review Questions**  
These questions check that you can relate the FTAP logic without losing on long proofs.

1. Define no-arbitrage in a one-period, finite-state market.
2. Explain what a state price vector represents.
3. Describe the meaning of an equivalent martingale measure.
4. Explain the difference between completeness and incompleteness.
5. Summarize the statement of the first and second fundamental theorems.

**Exercises**  
These exercises ask you to compute state prices and identify arbitrage in small markets.

1. Construct a two-state market with one risk-free and one risky asset. Compute the state prices and verify no arbitrage.
2. Given a three-state payoff matrix, determine whether the market is complete.
3. Price a contingent claim using state prices and compare the result to a replication strategy.
4. Identify an arbitrage opportunity in a small market by constructing a zero-cost portfolio with positive payoffs.
5. Use a risk-neutral probability vector to price a payoff and confirm that it matches the state price approach.

**Advanced Exercises**  
These exercises connect FTAP concepts to bounds and robustness.

1. Define price bounds for a contingent claim in an incomplete market.
2. Discuss how a change in the exercise affects the martingale measure.
3. Explain how FTAP logic extends to multi-period trees in a conceptual way.
4. Compare the FTAP pricing approach to a utility-based pricing view in a single paragraph.

**Capstone Project**  
This project asks you to build a compact finite-state pricing menu.

1. Specify a three-state market with a risk-free asset and two risky assets.
2. Compute state prices, check for arbitrage, and classify the market as complete or incomplete.
3. Price two different contingent claims and show the steps clearly.
4. Write a short conclusion that links the results to the FTAP statements.

# Coaching Guides Across Programs

All programs (CPF, TAE, TCE) include weekly and bi-weekly coaching guides with clear, day-by-day study plans; below are CPF and TAE examples.

**CPF Week 4 Coaching Guide**

Bridge Week: Covariance, Efficient Markets, Fundamental Theorems of Asset Pricing

CPF Program  
January 30, 2026

**Abstract**

Work 4 is the CPF bridge week. You work through three compact modules: Covariance, Efficient Markets, and the Fundamental Theorems of Asset Pricing as the transition from Financial Theory with Python (FTWP) to the multi-task phase is smooth. This guide provides the study plan, links to the study materials, align them with quick Python checks, and integrate the Engineering Deck classes without losing the focus on the core bridge topics.



CPF PROGRAM

**TAE Week 3 Coaching Guide**

LLMs, Attention & Tiny Transformers Capstone

The AI Engineer Program  
November 15, 2025

**Abstract**

Work 3 connects transformer theory with practical LLM engineering. You will deepen your understanding of attention and tiny decoder-only transformer models. You will learn how to use the AI/ML Engineering Deck to build an LLM and how to use it in an engineering workflow using Part II of the AI/ML Engineering book and the LLM guide. The capstone centers on a from-scratch tiny transformer language model built and trained in a Colab notebook, with clear logging, checkpoints, and sampling workflow.



THE AI  
ENGINEER

# Contact and Links

Dr. Yves J. Hilpisch

The Python Quants GmbH

**Email:** [team@tpq.io](mailto:team@tpq.io)  
**CPF Program:** [python-for-finance.com](http://python-for-finance.com)  
**TAE Program:** [theaiengineer.dev](http://theaiengineer.dev)  
**TCE Program:** [thecryptoengineer.dev](http://thecryptoengineer.dev)  
**Linktree:** [linktr.ee/dyjh](http://linktr.ee/dyjh)

