# Ezra Koreen

973-951-4954 | koreen.ezra@gmail.com | linkedin.com/in/ezra-koreen | github.com/ezrakoreen

## Education

**Cornell University** *GPA: 3.84* — Ithaca, NY
*B.S. in Computer Science* — *Expected Graduation: May 2028*

- Coursework: Deep Learning, Machine Learning, NLP, Algorithms, Computer Systems, Linear Algebra, Object-Oriented Programming and Data Structures, Discrete Math, Multivariable Calculus, Differential Equations
- Professional Development: Meet Millennium, IMC Prosperity 4 (216th/18,800)

## Experience

**Optiver** — June 2027 – Aug 2027
*Incoming Quantitative Trading Intern* — *Chicago, IL*

**SMBC** — June 2026 – Aug 2026
*Quantitative Research Intern – Systematic Trading* — *New York City, NY*

- Built a KDB pipeline to reconstruct inferred events and label 107M+ rows across 106 dates and 3 instruments
- Engineered 13 microstructure signals spanning a variety of event characteristics and order book dynamics
- Used regression and gradient boosted trees for next-tick prediction, improving OOS accuracy 6% over baseline
- Built 5-min forecasts using regularized logistic regression and shallow boosted trees with rolling validation

**Cornell University Extended Reality (CUXR)** — Sept 2025 – Present
*Lead Software Engineer* — *Ithaca, NY*

- Leading a 5-engineer team architecting a lifelong multimodal personal memory agent for AR smart glasses
- Implementing episodic memory and multimodal identity pipelines using GPT-5, RAG, and face/audio embeddings
- Designing a multi-service application using TypeScript, Python FastAPI, and PostgreSQL with pgvector

**General Catalyst | Cornell Venture Capital** — Feb 2026 – Present
*Software Engineer* — *Ithaca, NY*

- Built medical billing pipeline for General Catalyst to generate insurance codes from clinical notes via OpenAI API
- Integrated CMS fee schedules and billing constraints in SQLite to improve overall reimbursement by 17%
- Shipped React dashboard for note analysis, reimbursement optimization, and reviewing outputs for parsed notes

**Katz Quantum Information Lab** — May 2025 – Feb 2026
*Undergraduate Research Assistant* — *Ithaca, NY*

- Built Wigner function visualization and Gaussian wavepacket dynamics for trapped-ion GKP state simulations
- Simulated finite-energy constraints, validating analytical predictions and improving on SOTA gate fidelity
- Automated parameter sweeps and Wigner function, gate error, and fidelity visualization using QuTiP and NumPy

## Projects

**Prediction Market Trading** | *Python, AWS EC2* — July 2026 - Present

- Built a WebSocket-driven Polymarket US strategy using real-time order-book, spread, and liquidity signals
- Generated $22,500 in live PnL and a 1125× return on capital with 3.5% max drawdown since starting live trading
- Automated liquidity-based market selection, position sizing, and lifecycle management across 50+ markets
- Deployed a fault-isolated, multi-process execution system on AWS EC2 in us-east-1 for low-latency trading

**Byte Latent Transformer (BLT)** | *Python, PyTorch, AWS* — April 2026 – May 2026

- Implemented a tokenizer-free byte-level transformer operating directly on raw bytes with entropy-based patching
- Benchmarked BLT against a 49M-parameter GPT-2 baseline under a fixed FLOP budget on an AWS EC2 instance
- Adjusted parameter allocation and added patch-length embeddings to improve BPB by 5% on an 86M BLT model

**IMC Prosperity 4 (Algorithmic Trading Competition)** | *Python, Rust* — April 2026

- Developed trading strategies for IMC Prosperity 4, placing 216th/18,800 teams
- Implemented and tested market making, IV scalping, z-score mean reversion, and hybrid execution strategies
- Built a research pipeline around a Rust backtester to run parameter sweeps and evaluate strategies efficiently

**Memory-Optimized PoPE Attention Kernels** | *Python, PyTorch, CUDA, Triton* — March 2026

- Built 4 custom Triton forward-pass attention kernels for FlashAttention, RoPE, and 2 PoPE variants
- Reduced standard cartesian PoPE peak CUDA memory by 94.4%, from 89.02 MB to 5.00 MB at 4,096 tokens
- Validated causal masking, online softmax, and positional embeddings against PyTorch baselines with CUDA tests

## Technical Skills

**Languages**: Python, Java, C, C++, TypeScript, JavaScript, SQL, HTML/CSS, Rust, Linux/Bash, RISC-V Assembly
**Libraries**: PyTorch, CUDA, Triton, React, FastAPI, Node.js, Express.js, Next.js, OpenCV, sklearn, Pandas, NumPy
**Developer Tools**: Git, VS Code, AWS, IntelliJ, Google Colab, Jupyter, Cursor, Supabase