Polaris AI Studio · Foundry Team Training

Palantir Foundry curriculum for engineering teams

97 hands-on lessons across three bootcamps, about 107 hours of work. Every lesson has a step-by-step tutorial, an interactive diagram and homework, and each bootcamp builds working software on Foundry.

Palantir Hands-On Bootcamp: Property Maintenance Operations

60 lessons · ~49 h

What your team can ship: A working operations system on Foundry: data synced in and checked, an Ontology that Workshop apps and AIP agents can both use, an OSDK website outside Foundry — and the platform run properly, with projects, markings and cost under control.

Covers: Pipeline Builder · Ontology · Workshop · AIP Agent Studio · AIP Logic & Evals · OSDK external apps · Data Connection · Compass & projects · Permissions & markings · Cost control · Data lineage

1 The Palantir Story — From Seeing Stones to Enterprise AI
2 What is Palantir Foundry?
3 Uploading & Exploring Data
4 Pipeline Builder & Transformations
5 What is the Ontology?
6 Creating Object Types
7 Links, Actions & Object Explorer
8 Introduction to Workshop
9 Building the Maintenance Dashboard
10 Dashboard Charts & Analytics
11 AIP: AI Meets Your Data
12 Building an AIP Agent
13 AIP Logic, Automate & Full Integration
14 Foundry Developer Experience (FDE)
15 Data Lineage & Governance
16 OSDK — Build External Apps with the Ontology
17 Capstone Project — Full-Stack Foundry
18 Work Order Scheduling in Workshop
19 VROOM + OSRM Routing Engine
20 Automated Scheduling Pipeline + Capacity Engine
21 OSDK Mobile Apps — Technician & Resident Portals
22 Capstone 2 — Autonomous Dispatch System
23 Building a Custom Widget for Workshop
24 Custom Widgets 2 — Interactive Tables & Multi-Select
25 Measuring AI with AIP Evals
26 Exploring Data with AIP Analyst
27 AI FDE — Building Foundry by Conversation
28 AI FDE Skills — a Reusable Troubleshooting Library
29 Auto-Ingest & Refresh — Hands-off Data Updates
30 Reading Foundry Data from the Command Line
31 Writing to Foundry via Actions
32 OSDK — the Typed SDK
33 Auto-Alert: Analyst → Logic → Automate
34 Global Branching & Proposals — One Change, Three Apps
35 Code Repositories CI/CD — Protected Branches, Red Builds, Tagged Releases
36 Automate Part 3 — Chaining, Retries & Failure Monitoring
37 Workshop Advanced — Embedded Modules, Routing, State & Events
38 Document Intelligence — From Scanned PDFs to Searchable Chunks
39 Quiver — Point-and-Click Analysis, Dashboards & a Workshop Embed
40 AIP Logic Part 2 — Loops, Reduce, Conditionals & Tools
41 AIP Evals Part 2 — Experiments, Branches & Run History
42 Managing an Ontology Over Time — Value Types, Shared Properties & Cleanup
43 Schedules Part 2 — Failure Handling: Expectations, Retries & Health Checks

44 Compute Modules — Run VROOM Inside Foundry as a Function
45 Release to Production — DevOps, Marketplace & the Admin-Owned Blockers
46 MCP Hub — Palantir MCP vs Ontology MCP, Tokens & Governance
47 AIP Evolve — Agents That Improve Your AI Systems (and How to Review Them)
48 Compass, Projects & the Foundry Filesystem
49 Data Connection — Sources, Syncs, Runtimes & Virtual Tables
50 Contour — Point-and-Click Analysis, Dashboards & Saved Datasets
51 Permissions & Markings — Mandatory Access Control
52 Cost Control & Resource Management
53 Ontology Interfaces & Complex Relationships — Contracts, Links and Rules That Follow Them
54 Multi-Object State Machines — One Submission, Several Objects, Only Legal Moves
55 Pipeline Builder at Scale — Vendor Harmonization: Golden Records, a Crosswalk and a Scorecard
56 External-System-Backed Objects — CRM Sync: Mirror Vendors from Another System of Record
57 Write-Back with an Outbox — Send Work to Vendors Exactly Once
58 Vendor Portal — Let Vendors Answer Jobs, with the Rules in Foundry
59 Feature Flags on the Ontology — Switch Features On, Off, or for Some Vendors, Without a Redeploy
60 Work Order Control Tower — A Website That Doesn't Look Like Foundry

TypeScript for Foundry: From First Script to OSDK

27 lessons · ~46 h

What your team can ship: Typed, tested Foundry code from engineers who start with no TypeScript: OSDK queries and actions, Code Repository Functions, a production client with retries and safe writes, and a Next.js app on the OSDK.

Covers: TypeScript from zero · OSDK queries & actions · Code Repository Functions · Debugging · Zod validation · Testing with Vitest · Next.js on the OSDK

Assumes some Foundry background; the Palantir bootcamp covers it.

1 Why TypeScript & Your Dev Environment
2 Variables, Primitive Types & Type Inference
3 Arrays, Tuples & Object Types
4 Functions in TypeScript
5 Union Types, Literal Types, Enums & Narrowing
6 Interfaces vs. Type Aliases
7 Classes in TypeScript
8 Generics — the Basics
9 Async/Await & Promises
10 Modules, npm, tsconfig & Reading Compiler Errors
11 OSDK Setup, Auth & Typed Queries
12 Typed Actions & the OSDK Idiom
13 Code Repository Functions — Write & Type It Right
14 Unit-Testing and Invoking Functions for Real
15 Generating Your SDK Locally with @osdk/foundry-sdk-generator
16 Reading & Fixing TS Compiler Errors in a Foundry Project
17 Debugging OSDK & Foundry Runtime Errors
18 Capstone Debugging Lab — Diagnose & Fix Broken OSDK Code
19 Pilot — Generate an OSDK App, Then Read & Debug Its TypeScript
20 Types at the Boundary — Utility, Mapped & Template-Literal Types, satisfies
21 Runtime Validation with Zod — Env, Wire Contracts & Unknown Errors
22 A Production Foundry Client — Tokens, Retries, Timeouts & Error Classes
23 Safe Actions — Validate, Apply, Recover the Key, Never Duplicate
24 Querying at Scale — Links, Aggregations, Object Sets & Interfaces
25 Testing Foundry Code — Fakes, Mock Clients, Fake Timers & Live Gates
26 Shipping a Node Service — tsconfig, Lint Boundaries, Bundling, Secrets & Logs
27 Capstone — A Next.js App on the OSDK

Jev × Foundry: Calibrated AI Decisions on the Ontology

10 lessons · ~12 h

What your team can ship: A calibrated decision step inside a Foundry workflow: answers with probabilities, low-confidence cases routed to a Workshop review queue, and an evaluation harness that checks every change.

Covers: Calibrated answers · Question design · Ontology in, Action out · Workshop review queue · Evaluation harness · Jev vs. an LLM in AIP

Builds on the Palantir and TypeScript for Foundry bootcamps.

1. What Is Jev? Decisions, Not Text
2. Your First Jev Call: State, Questions, Answers
3. Designing Questions Jev Answers Well
4. Confidence, Calibration & Human Review
5. Wiring Jev into Foundry: Ontology In, Action Out
6. Capstone: Dispatch with Jev + Constraints
7. Evaluating Jev: A Harness on Saved Answers
8. Cost, Latency & Failure: Keeping Jev Useful When Things Go Wrong
9. The Human Review Queue in Workshop
10. Jev vs. an LLM on the Same Questions

Formats and pricing (USD)

Team Licence	$750 per seat per year; $600 per seat at 10+ seats · $450 at 25+. Self-paced. All 97 Foundry lessons. Minimum 5 seats. Seats can move to a colleague.
Live Cohort	$2,800 per seat. 6 weeks, fixed start date. Weekly live Q&A. Homework review. Deliverable: Your team ships a working Work Order Control Tower on Foundry, built in Lesson 60 (Work Order Control Tower).
Private Team Workshop	From $18,000. 3 days, remote or on-site. Up to 12 people. Illustrated hands-on handouts included.

Book a scoping call: calendly.com/polarisaistudio/introduction-call · Email: polarisaistudio@gmail.com