

# Foundry IQ + Fabric IQ Workshop

## Deployment Guide

### Prerequisites

- Azure subscription with Contributor access
- Microsoft Fabric workspace (F2+ capacity) with admin permissions
- VS Code, Azure Developer CLI (aka.ms/azd), Python 3.10+, Git

### Deployment Steps

#### 1. Clone Repository

```
git clone https://github.com/nchandhi/nc-iq-workshop.git && cd nc-iq-workshop
```

#### 2. Login to Azure

```
azd auth login
```

#### 3. Deploy Azure Resources (~7 min)

```
azd up
```

Environment: iq-workshop-yourname | Location: eastus2 or westus2

#### 4. Configure Fabric

Go to app.fabric.microsoft.com > Open workspace > Copy Workspace ID from URL

#### 5. Configure Environment

```
cp .env.example .env
```

Edit .env: Set FABRIC\_WORKSPACE\_ID and DATA\_FOLDER=data/default

#### 6. Setup Python

```
cd scripts && python -m venv .venv && .venv\Scripts\activate && pip install uv && uv pip install -r requirements.txt
```

#### 7. Build Solution (~5 min)

```
python scripts/00_build_solution.py --from 02
```

#### 8. Test the Agent

```
python scripts/08_test_foundry_agent.py
```

### Sample Questions to Try

Structured: "How many outages last month?" | "Average resolution time?"

Unstructured: "What are the policies for notifying customers?"

Combined: "Which outages exceeded our policy thresholds?"

### Customize for Your Use Case

After testing the default scenario, build your own:

#### 1. Run with your industry

```
python scripts/00_build_solution.py --clean --industry "Insurance" --usecase "Claims processing"
```

#### 2. Or set in .env

```
INDUSTRY=Insurance
```

```
USECASE=Claims processing and policy management
```

Then run: python scripts/00\_build\_solution.py --clean

Sample industries: Telecommunications, Insurance, Finance, Retail, Manufacturing, Energy

### Troubleshooting

Tip: Use GitHub Copilot Chat (Ctrl+I in VS Code) for help with errors.

### Resources

Workshop: nchandhi.github.io/nc-iq-workshop | Repo: [github.com/nchandhi/nc-iq-workshop](https://github.com/nchandhi/nc-iq-workshop)