Hands■On Lab: Data API Builder (DAB) for Azure SQL + Power BI

Goal: Expose an existing data warehouse (Azure SQL or SQL Server) through Data API Builder (DAB) as REST/GraphQL endpoints, deploy to Azure, and connect to Power BI.

1) Prerequisites

- Azure SQL Database (or on-prem SQL Server with firewall opened).
- Azure CLI installed.
- .NET 8 SDK installed.
- Install DAB: `dotnet tool install --global Microsoft.DataApiBuilder`
- Power BI Desktop (for testing dashboard integration).

2) Initialize DAB Project

```
# Create folder and init
mkdir dab-lab && cd dab-lab
dab init --database-type "mssql"   --connection-string "Server=tcp:<server>.database.windows.net;
```

3) Configure Entities (dab-config.json)

```
{
  "data-source": {
    "database-type": "mssql",
    "connection-string": "Server=...;Database=...;User Id=...;Password=..."
  },
  "entities": {
    "SalesOrders": {
      "source": "dbo.SalesOrders",
      "permissions": [
        { "role": "anonymous", "actions": [ "read" ] }
      ]
    },
    "MonthlyRevenue": {
      "source": "dbo.vwMonthlyRevenue",
      "permissions": [
        { "role": "reportUser", "actions": [ "read" ] }
      ]
    }
  }
}
```

Tip: Use **views** for business logic (e.g., KPIs, aggregated metrics) instead of exposing raw fact tables directly.

4) Run DAB Locally

```
dab start
# REST endpoint:  http://localhost:5000/api/SalesOrders
# GraphQL endpoint: http://localhost:5000/graphql
```

5) Secure with Azure AD

- Register an Azure AD app for DAB in Azure AD.
- Update `dab-config.json` with authentication settings (`--auth aad`).
- Assign roles (`reportUser`) via app roles or group claims.
- Test with a user account to verify row-level security.

6) Deploy to Azure Container Apps

```
# Create ACA environment & app
rg=dab-lab-rg
loc=westeurope
acae=dab-env
ca=dab-api
acr=dablabacr123

az group create -n $rg -l $loc
az acr create -g $rg -n $acr --sku Basic
az acr login -n $acr

# build image
docker build -t $acr.azurecr.io/dab-lab:latest .
docker push $acr.azurecr.io/dab-lab:latest

# deploy
az containerapp env create -g $rg -n $acae -l $loc
az containerapp create -g $rg -n $ca --environment $acae   --image $acr.azurecr.io/dab-lab:latest
```

7) Connect to Power BI

- Open Power BI Desktop.
- Choose Data Source → Web → enter DAB REST endpoint (e.g., `/api/MonthlyRevenue`).
- If using GraphQL: select Data Source → Web → POST query to `/graphql`.
- Model relationships in Power BI if pulling multiple entities.
- Build dashboard visuals (e.g., revenue trend, top customers).

8) Advanced Scenarios

- Expose stored procedures for advanced reporting.
- Use APIM (API Management) to throttle or expose externally.
- Integrate with Microsoft Fabric by pulling API data into OneLake pipelines.
- Enable row-level security by user claims in Azure AD.