**Microsoft Purview Governance & Automation Study Guide**

**1. Collection Hierarchy & RBAC**

**Collection Hierarchy**:

* Collections are logical containers that help organize data sources, assets, and metadata.
* They enable scoping of governance efforts by domain, department, or data zone (e.g., bronze/silver/gold).
* Sub-collections can be nested to create hierarchical structures.

**Role-Based Access Control (RBAC)**:

* Provides fine-grained access control at collection level.
* Common built-in roles:
  + **Collection Admin**: Full control over collection.
  + **Data Curator**: Manage assets and metadata.
  + **Data Reader**: View-only permissions.
* Supports Azure AD integration for role assignment.

**2. Scan Configuration**

**Purpose**:

* Scans extract metadata from data sources into Microsoft Purview.

**Components**:

* **Data Source**: E.g., Azure Blob Storage, SQL DB, Synapse.
* **Authentication**: Managed Identity or Key Vault reference.
* **Rule Set**: Includes classification rules.
* **Schedule**: One-time or recurring scan intervals.
* **Incremental Scanning**: For large datasets, enables scanning only new/changed assets.

**Monitoring**:

* Scan status and logs can be tracked via Purview portal.

**3. Custom Classification Rules**

**Use Case**:

* Identify organization-specific sensitive data like internal IDs, contract codes, or PII.

**Features**:

* Defined using Regular Expressions (regex).
* Can be prioritized over built-in classifiers.
* Can be scoped to specific file types or column names.
* Helps in compliance reporting and data protection.

**4. Glossary Terms**

**Definition**:

* Glossary terms provide standardized business context to data assets.

**Key Features**:

* Terms can have descriptions, synonyms, acronyms.
* Terms are linked to assets (tables, files, etc.) for discoverability.
* Support for term hierarchy and governance workflows.
* Promotes consistent vocabulary across business units.

**5. Purview ↔ Databricks Lineage Push**

**Objective**:

* Enhance end-to-end lineage by integrating transformation logic from Databricks.

**How It Works**:

* Databricks jobs or notebooks push lineage metadata to Purview using REST API.
* Captures dependencies and transformations between source → logic → output.

**Benefits**:

* Improved visibility into data flow.
* Enables better impact analysis.
* Supports data compliance and auditing.

**6. Governance Scorecard**

**Purpose**:

* Measures and visualizes governance adoption across the catalog.

**Metrics Tracked**:

* Scan coverage (how much of your environment is scanned).
* Classification coverage (how many assets have sensitivity labels).
* Glossary coverage (how many assets are linked to business terms).

**Use Cases**:

* Governance performance tracking.
* Compliance program support.
* Identifying gaps for improvement.

**7. Stretch Lab: Automate Scan via DevOps Pipeline**

**Goal**:

* Automate metadata scanning as part of CI/CD for scalable governance.

**Tools**:

* Azure DevOps or GitHub Actions.
* PowerShell or REST APIs.

**Steps**:

1. Store scan configurations in parameterized ARM templates.
2. Authenticate using Managed Identity or Service Principal.
3. Trigger scan via REST API post-deployment.
4. Log scan results to storage or dashboard.

**Benefits**:

* Standardized deployment across environments.
* Version-controlled governance configurations.
* Faster onboarding of new data sources.

**Tip**: Combine this guide with hands-on labs using Microsoft Purview Studio and Azure DevOps to reinforce learning.