

Lesson 20 – Publishing and Sharing in Power BI (Advanced) with concise, written explanations for each question.

1. How does Power BI handle large datasets in the Online Service? Role of Premium Capacity

Large datasets: Power BI Pro has size limits (1 GB per dataset). Premium increases the dataset size limit (up to 400 GB per dataset depending on SKU).

Premium Capacity: Provides dedicated resources (memory & CPU), supports larger datasets, higher refresh rates (48/day vs. 8/day in Pro), and advanced features like incremental refresh and aggregation tables.

2. Import mode vs DirectQuery vs Live Connection

Import Mode: Data is loaded into Power BI's in-memory engine. Very fast queries, but data must be refreshed. Limited by dataset size.

DirectQuery: Data stays in the source system. Queries run in real-time. Useful for very large/real-time data but can be slower and limited in modeling features.

Live Connection: Connects directly to Analysis Services (on-prem or Azure) or another Power BI dataset. No data is stored in Power BI; all modeling happens at the source.

3. Deployment pipelines in Power BI Online

Purpose: Enable Dev–Test–Prod lifecycle management for reports, datasets, and dashboards.

Stages:

Development – Build and test new content.

Test – Validate with a subset of users; check data/permissions.

Production – Publish to broad audience.

Pipelines ensure version control, consistency, and safer rollout of content.

4. Integration with Microsoft Teams or SharePoint

Teams: Power BI app for Teams allows embedding reports directly in Teams channels, chats, or tabs. Users can collaborate in context.

SharePoint: Reports can be embedded in a SharePoint Online page using the Power BI web part. Useful for intranet and document-centric collaboration.

5. XMLA endpoint in Premium

XMLA endpoint: Allows connections to Power BI datasets using XML for Analysis protocol (the same used by SQL Server Analysis Services).

Benefits: Developers can connect with tools like SSMS, Tabular Editor, ALM Toolkit, automate deployments, manage partitions, and perform enterprise-level model management.

6. Usage metrics and audit logs

Usage Metrics: Built-in reports in Power BI Service that show how often reports/dashboards are viewed, by whom, and when.

Audit Logs: Available through Microsoft 365 compliance center; record detailed user activities (view, share, download, export). Essential for security and compliance monitoring.

7. Managing workspace access and permissions

In each workspace, assign roles:

Admin – full control, including adding/removing users.

Member – edit content.

Contributor – create/edit reports but no publish app.

Viewer – read-only access.

Permissions can also be managed via security groups in Azure Active Directory for easier governance.

8. Data governance in Power BI Service

Use sensitivity labels, certified datasets, and endorsements to control trust level of data.

Apply RLS (Row-Level Security) and object-level security.

Leverage dataflows for centralized ETL.

Admin portal settings enforce tenant-wide policies (sharing restrictions, export controls, integration settings).

Audit logs and usage metrics for compliance monitoring.

9. Limitations of RLS with DirectQuery or Live Connection

DirectQuery: RLS is applied, but performance may degrade because filters are translated into queries at runtime.

Live Connection (to Analysis Services or another dataset): RLS is not defined in Power BI Service—it must be implemented in the source model (e.g., in SSAS or the parent dataset).

Limitation: You cannot mix Power BI-defined RLS with Live Connection sources.

10. Refreshing a dataset via Power Automate or REST API

Power Automate: Use the “Refresh a dataset” action in the Power BI connector to trigger refresh on-demand or based on an event (e.g., file upload, Teams message).

REST API: Endpoint `POST /groups/{groupId}/datasets/{datasetId}/refreshes` triggers a refresh programmatically. Useful for automation, CI/CD, and integration with external scheduling tools.