

Introduction to Fabric

Identify requirements for a solution, including components, features, performance, and capacity stock-keeping units (SKUs)

License Model	Acronym	What you can use	Note
Free			Cannot view shared Power BI objects
Pro		Power BI	
Premium per-user	PPU	Power BI	
Premium per capacity (up to June 2024)	P	Fabric	You can use a free Microsoft Fabric license to viewer shared Power BI objects.
Embedded	A / EM	Power BI	
Fabric capacity	F	Fabric	If F32 or below, you need a Power BI paid license to view shared Power BI objects. If F64 or above, you can use a free Microsoft Fabric license to viewer shared Power BI objects.

DP-600: Fabric Analytics Engineer Associate
Introduction to Fabric

SKU	Capacity Units (CU)	Power BI SKU	Power BI v-cores	Cost per hour
F2	2	-	0.25	\$0.36
F4	4	-	0.5	\$0.72
F8	8	EM/A1	1	\$1.44
F16	16	EM2/A2	2	\$2.88
F32	32	EM3/A3	4	\$5.76
F64	64	P1/A4	8	\$11.52
F128	128	P2/A5	16	\$23.04
F256	256	P3/A6	32	\$46.08
F512	512	P4/A7	64	\$92.16
F1024	1024	P5/A8	128	\$184.32
F2048	2048	-	256	\$368.64

- Reservation (for a year): 41% discount.
- OneLake storage: 2.6 cents/Gb/month, or 4.68 cents if you have enabled Business Continuity and Disaster Recovery.
- You can buy Fabric either pay monthly or with a capacity reservation (reserved for a year in advance).
 - You may be able to cancel a capacity reservation, but there may be a termination fee.
 - It would probably be refunded as a credit.
- You can buy a Microsoft Fabric Capacity reservation by:
 - Going to the Azure portal,
 - Searching for Reservations in the top bar.
 - Click +Add, then Microsoft Fabric

DP-600: Fabric Analytics Engineer Associate
Implement and manage a data analytics environment

- A Capacity Unit is a pool of compute power, needed to run queries, jobs or tasks in Fabric.
- If you need additional capacity, you can add it on a pay-as-you-go basis.
 - You can scale this additional capacity up or down, or pause it.
- A Power BI Pro or Premium Per User license is needed to view Power BI content, unless you have a Premium capacity or an F64 or larger capacities (the new equivalent).
- You can use other Fabric content without a Power BI Pro license. So you can use pipelines, create data warehouse, use notebooks etc with a Fabric Free license, as long as:
 - There is a Fabric capacity, and
 - You have at least a Viewer role for the workspace.
- To check whether Fabric is enabled, go to the Admin portal – tenant settings, and enable “Users can create Fabric items”.

Implement and manage a data analytics environment

1. Implement workspace-level access controls

- Microsoft Fabric uses Power BI roles for Microsoft Fabric capabilities.
- The following is what each role does for Microsoft lakehouses/warehouses and related apps
- Viewer
 - View/read content of data pipelines and notebooks
 - Execute/cancel execution of data pipelines (not notebooks)
 - View execution output of data pipelines and notebooks
 - Connect to and Read data/shortcuts through Lakehouse/Warehouse SQL analytics endpoint
 - Reshare items in a workspace, if you have Reshare permissions
- Contributor (as Viewer, plus)
 - Read Lakehouse/Warehouse data/shortcuts through OneLake APIs and Spark.
 - Read Lakehouse data through Lakehouse Explorer.
 - Write/delete data pipelines and notebooks
 - Execute/cancel execution of notebooks
 - Schedule data-refresh via the on-premises gateway
 - Modify gateway connection settings
- Member
 - Add members, contributor and viewers
 - Allow others to reshare items
- Admin
 - Update and delete the workspace
 - Add/remove admins, members, contributors and viewers.
- To give access to your workspace:

DP-600: Fabric Analytics Engineer Associate
Implement and manage a data analytics environment

- In the workspace, click on Manage Access (it may be in the ... section)
- Click “+Add people or groups”.
- Enter name/email and role, and click Add.
- You can view/modify access later if needed.

2. Implement item-level access controls

- To share items via a link:
 - click the Share button.
 - In the “Create and send link” dialog, click “People in your organization can view”.
 - In the “Select permissions” dialog, choose either:
 - People in your organization
 - This allows for read-only access (as a minimum).
 - It does not work for external or guest users.
 - People with existing access
 - This generates a link. It does not change the access.
 - Specific people.
 - This also allows guest users in your organization’s Microsoft Entra ID.
 - You can also allow for re-share and edit permissions.
 - If you re-share, you can only re-share based on the permissions you have.
 - Then click Apply.
- In the “Create and send link” dialog, you can copy the sharing link, generate an email, or share via Teams.
- You can manage item links:
 - Click on “Manage permissions” in the “Create and send link” or click on “Manage permissions” in the ... next to the item in the workspace.
 - You can either:
 - Click on the Edit next to “Links that give access”,
 - You can then modify the permissions for the link, or delete the link.
 - Or
 - Click on “Advanced” to:
 - View/manage/create links,
 - View/manage/grant who has direct access,
 - Apply filters or search for links/people.
- Note: The Browse – “Shared with me” does not show Fabric items which are not Power BI objects.

- You can also manage permissions for lakehouses by clicking on the ... next to the lakehouse (in the Workspace) and going to Manage permissions. You can assign the following permissions:
 - Read all of the lakehouse table data (not files) using the SQL endpoint,
 - Read all of the underlying data files with Apache Spark,
 - Build reports on the default semantic model.
- For data warehouse:
 - They cannot see any of the data unless at least one additional permission is selected.
 - the "Apache Spark" permission is expanded to "Read all of the data warehouse's underlying OneLake files using Apache Spark, Pipelines, or other apps which access the OneLake data directly".
- You can also share notebooks with the following permissions:
 - Share (or Reshare) the notebook with others,
 - Edit (or Write) all notebook cells, and
 - Run (or Execute) all notebook cells.

3. Implement dynamic row-level, column-level, object-level and file-level access control

Row-level control

- To implement row-level security in a Data Warehouse, you need to have:
 - A security function. This needs to RETURNS TABLE WITH SCHEMABINDING, and returns SELECT 1 where the appropriate conditions are true. For example:
 - A security policy, which references the security function.

```
CREATE FUNCTION securityfunction(@UserName as VARCHAR(50))
RETURNS TABLE
WITH SCHEMABINDING
AS
RETURN SELECT 1 AS securityfunction
WHERE @UserName = LEFT(USER_NAME(), LEN(@UserName)) OR
LEFT(USER_NAME(),4) = 'Jane'
```

```
CREATE SECURITY POLICY SecurityPolicy
ADD FILTER PREDICATE dbo.securityfunction(UserName)
ON dbo.tblActual
WITH (STATE = ON);
```

Column-level control

- You can implement column-level control by granting SELECT permissions to only some of the columns in a table or query:
 - GRANT SELECT ON NameOfTable(ColumnNames) TO [UserName]

Object-level control

- There are three main commands to manage permissions:

DP-600: Fabric Analytics Engineer Associate
Implement and manage a data analytics environment

- GRANT permission ON object TO user/security group – this permits access to an object, as long as there isn't a DENY
- DENY permission ON object TO user/security group – this prevents access to an object. It overrules a GRANT.
 - DENY does not apply to owners of objects or members of the sysadmin fixed server role.
- REVOKE permission ON object TO user/security group – removes a GRANT or DENY
- Examples of objects include Table, View, Function, Stored Procedure.
- Permissions include SELECT, INSERT, UPDATE, DELETE and EXECUTE (run stored procedures)
- You can also add WITH GRANT OPTION to allow that user to GRANT the permission to other user/security groups.
- A list of the principals (user/login/group) can be found by running
 - SELECT * FROM sys.database_principals
 - The type column includes:
 - E – External user from Microsoft Entra ID,
 - S – SQL user
 - X – External group from Microsoft Entra group or applications

Object-level security using Tabular Editor

- You can create object-level security (OLS) using Tabular Editor.
 - The “object” can be a table or column.
- First of all, create a role in the same way as using RLS.
 - Go to Modeling – Manage roles.
 - Click Create.
 - Define the roles as per RLS, but you don't need to define any RLS rules.
- Then, go to Tabular Editor.
 - If you are securing tables:
 - In the model, go to Roles and click on the relevant role.
 - In Permissions, expand Security – Table Permissions.
 - Change the “Table Permissions” from Default to None.
 - If you are securing columns:
 - Expand Tables – [Name of Table] – [Name of column].
 - In Permissions, expand Object Level Security – [Name of role].
 - Change to “None”.
 - Then click Save.
 - Go back to Power BI Desktop, and publish to the Power BI Service.
 - In the Power BI Service, go to the semantic model, ..., Security, and assign rules.

DP-600: Fabric Analytics Engineer Associate
Implement and manage a data analytics environment

- Note:
 - OLS does not apply to Admin, Member or Contributors.
 - You cannot use OLS and any of the following visuals: Q&A, Quick Insights or Smart Narrative.
 - You cannot secure a table which is in the middle of a relationship.
 - For example, DimProductSubcategory links to DimProductCategory and DimProduct,
 - However, you could add an additional relationship between the two affected tables.
 - You can use OLS for columns in a table which is part of the relationship, as long as the key column is not secured.
 - You cannot use RLS using one role and OLS using another role.
 - It could lead to unexpected access to secure data.
 - Calculations (such as measures) do not work if they refer to a secure table or column.

File-level control

- To configure access to folders in a Lakehouse, click on “Manage OneLake data access”.
 - You need to have Contributor, Member or Administrator access for the workspace.
 - Once enabled, it cannot be turned off.
- A new role “DefaultReader” will be created, which gives you access to read all files.
 - You can delete this default role by checking it and clicking on “Delete”.
 - You will also be granted all permissions: Read, Write, ReadAll, Reshare, Execute, ViewOutput and ViewLogs.
- You can create a role by clicking on “New Role”.
 - The role name must start with a letter, contain only letters or numbers (up to 128 characters), and be unique.
- You can then select “All folders” or “Selected folders”.
 - Folders also include subfolders within the “Tables” folder. This means that you can control access to the data within tables.
 - Only check a folder if you want the user to have permissions for that folder.
- You can then click “Assign role” to add people, groups, or email addresses to the role.
 - You can also access this by clicking on the ... next to the role and click on “Assign”.
 - You can assign users by clicking:
 - inside the “Add people or groups” box, or
 - the “Add users based on Lakehouse permissions” (these are called “virtual members”).
 - The Lakehouse permissions are: Read, Write, Reshare, Execute and ReadAll.

DP-600: Fabric Analytics Engineer Associate
Implement and manage a data analytics environment

- These permissions use an AND – if you select “Read, Execute”, then this role will only apply to users who have both the Read and Execute permissions (not OR).

4. Apply sensitivity labels to items

- You can apply sensitivity labels to Fabric items, if you have Power BI Pro or Premium Per User license.
 - These labels do not affect access to content from Fabric or the Power BI Service.
 - They can affect access to content in Power BI Desktop.
 - They are also applied to any content which is exported:
 - Export to Excel/PowerPoint/PDF,
 - Analyze in Excel from the Power BI Service,
 - Creating a PivotTable from a semantic model, or
 - Download to .pbix from the Power BI Service.
- This is for items including lakehouses, data warehouses, pipelines, semantic models, notebooks, eventhouse, eventstreams, reports, and dashboards, and Dataflow Gen2.
- There are two ways to do this:
 - It may be at the top of the screen, next to the item name
 - This does not apply to Dataflow Gen2.
 - It may be in the Settings:
 - click ... next to the item,
 - go to Settings
 - go to the Sensitivity label section or tab.
 - You may also be able to select “Apply to downstream items”. If so, the label will be applied to items created downstream.
 - For example, a semantic model created from a Lakehouse, or a report created from that semantic model, or a dashboard which has a tile which is pinned from the report.
 - If, for some reason, it cannot be so applied on a downstream item, the new item can still be created.
- The default sensitivity labels are:
 - Highly Confidential
 - Confidential
 - No Sensitivity
 - None (this could be any sensitivity, but hasn't been assessed yet).
- Once set, they will appear in a column in the workspace, and at the top of the screen when the item is opened.
- In Power BI Desktop, you can also add sensitivity labels in Home – Sensitivity.
 - It is then visible in the status bar.

- To enable sensitivity labels to be set, they need to be enabled in the Tenant:
 - Go to the Fabric admin portal, and then Tenant settings.
 - In the Information protection section, enable “Allow users to apply sensitivity labels for content”.
 - To allow for the downstream of updated sensitivity labels, enable “Automatically apply sensitivity labels to downstream content”.
 - You can also “allow workspace admins to override automatically applied sensitivity labels”, and “Restrict content with protected labels from being shared via link with everyone in your organization”.
- This list can be updated in Microsoft Purview Information Protection portal (purview.microsoft.com).
 - You can also create a Policy to apply a default label for new Power BI reports, dashboards and semantic models can be applied in the Microsoft Purview compliance portal.
- If you create semantic models which are connected to data sources which have got sensitivity labels, then the semantic models can inherit those labels. These data sources are:
 - Excel files stored on OneDrive or SharePoint Online (not on premises behind a gateway),
 - Azure Synapse Analytics, and
 - Azure SQL Database.
- It requires additional settings in Microsoft Purview, and in the Fabric tenant:
 - Go to the Fabric admin portal, and then Tenant settings.
 - In the Information protection section, enable “Apply sensitivity labels from data sources to their data in Power BI”.

5. Endorse Items

- There are three different types of endorsement: promotion, certification and master data.
 - Promotion.
 - This can be applied to any item apart from Power BI Dashboards.
 - It advertises the item’s existence, and should only be used for data which is of sufficient quality to share with others.
 - Promotion can be done by any user.
 - Certification.
 - This can be applied to any item apart from Power BI Dashboards.
 - It is used to say this item is quality-controlled by a user who understands the data, and can be used in the entire tenant.
 - Certification can only be done by authorized reviewers.
 - Master data.
 - This can only be applied to items which have data, such as lakehouses, data warehouses, and semantic models.

- It is used for “single-source-of-truth” data. It is the central source for that data.
- Setting “Master data” status can only be done by authorized reviewers.
- To endorse an item:
 - Click on the ... next to the item and go to Settings.
 - Change the endorsement.
 - For semantic models, you can also check “Make discoverable”. This makes it shown in the OneLake data hub, and users who haven’t got access to it can request access.
- Once endorsed, you will see the endorsement:
 - in a column in the Workspace, and
 - in the drop-down at the top of the screen, when the item is opened.
- In the Admin portal – Tenant settings, you can adjust the following:
 - Enable “Certification”
 - You can also “Specify URL for documentation page”. This is the page that users will see when clicking “How do I get content certified?”
 - Enable the “Master data” badge.
 - “Make promoted content discoverable” and “Make certified content discoverable”.

Manage the analytics development lifecycle

6. Implement version control for a workspace

- Git integration allows you to integrate your development processes into Fabric. It works on a workspace level. Note:
 - this is used through Azure DevOps Git Repos with the same tenant as the Fabric tenant
 - not through GitHub Repos, and not the on-premises version of Azure DevOps.
- It allows you to:
 - Backup and version work,
 - Revert to previous stages if needed,
 - Collaborate with others,
 - Work alone using Git branches,
 - Use Git source control tools.
- You can use it for:
 - Data pipelines,
 - Lakehouse,
 - Notebooks,
 - Paginated reports,

DP-600: Fabric Analytics Engineer Associate
Manage the analytics development lifecycle

- Reports (except where the semantic model is in SSAS or Azure Analytics Services, or semantic models hosted in My Workspace),
- Semantic models (except live connections and models created from the Data warehouse/lakehouse).
- You need:
 - An active Azure account for the same user that uses the Fabric workspace,
 - Access to an existing Azure DevOps repository,
 - Power BI Premium license (for Power BI items only) or Fabric capacity (for all Fabric items),
 - In Settings – Admin Portal, have “Users can create Fabric items” enabled.
- To sign up to Azure Repos:
 - Go to the Azure Portal (portal.azure.com).
 - Search in the top bar for “DevOps”, and click on "Azure DevOps organizations".
 - Click on “My Azure DevOps Organizations”.
 - Click on “Create new organization”, and enter your organization details, including:
 - The organization name,
 - The location for hosting your projects
 - Create a new project.
 - Enter the project details, including:
 - The Project Name, and
 - The visibility.
 - In Repos – Files, click “Initialize” to create an empty branch.
- To connect your workspace to an Azure repo:
 - You will need Admin rights for the Workspace, and Read rights for the Git repo.
 - Go to the relevant workspace.
 - Click on “Workspace settings” (it might be in the ... section),
 - Go to Git integration. Select:
 - Organization,
 - Project,
 - Git repository,
 - Branch
 - You can click “+New Branch” to create a new branch.
 - You will need Admin rights for the workspace, and Write and Create branch rights for the Git repo.
 - Folder:
 - Use an existing folder,
 - Enter a name for a new folder, or

- Leave blank to use the root folder of the branch.
 - You can only connect a workspace to one branch and one folder at a time.
- You can disconnect by going to Git integration and click “Disconnect workspace”.
 - You will need Admin rights for the Workspace, but no rights are needed for the Git repo.
- After you connect, if the workspace or Git branch is empty, content will be copied.
 - It doesn’t sync data, but only the schema.
- Once connected, the Workspace includes a “Git status” column showing its status.
- To commit changes to Git:
 - You will need at least Contributor rights for the Workspace, relevant permissions for the items and external dependencies, and Read and Contribute rights for the Git repo.
 - In the workspace, click on the “Source control” icon.
 - Go to the Changes tab in the Source control pane.
 - A list shows the changed icons, with icons showing:
 - new (green +),
 - modified (brown non-equal sign),
 - conflict (red x), or
 - deleted (red -).
 - Select all the items you want to “commit” (transfer).
 - To commit all, check the top box.
 - You can add a comment in the Commit Message box.
 - You can then click "Commit".
 - Afterwards, the status of the selected items would change from “Uncommitted” to “Synced”.
 - You can also see the time of the last commit in the footer.
 - If you click "Update", then all changes in the branch will be updated.
- If changes have been made in the connected Git branch:
 - You will see a notification.
 - You can click on the “Source control” icon and go to the Updates tab to see a list of all changed items.
 - You can then click on “Update all”.
 - You will need at least Contributor rights in the Workspace, relevant permissions for the items and external dependencies, as well as Read rights for the Git repo.

7. Create and manage a Power BI Desktop project (.pbip)

- Power BI Project (.pbip) stores your work with report and semantic model item definitions as separate text files in a folder structure.

- The Semantic Model and Report will be saved in folders called [projectname].SemanticModel and .Report.
- There is also a [projectname].pbip file, which points to the report folder.
- There is also a .gitignore file, which tracks files that Git should ignore.
- This allows:
 - You can use source control such as Git to track version history, compare revisions, and revert to previous versions.
 - You can edit the definition files in Notepad or Visual Studio Code
 - VS Code can integrate with Git.
 - You can Tabular Model Scripting Language (TMSL) to make changes to your items.
 - This can be used in Tabular Editor.
 - You can use it in a Continuous Integration and Continuous Delivery (CI/CD) system.
- This is currently in preview. To enable it, go to Power BI Desktop – File – Options and settings – Options – Preview features, and check “Power BI Project (.pbip) save option”.
 - This stores it in Tabular Model Scripting Language (TMSL).
 - You can also enable "Store semantic model using TMDL format". Tabular Model Definition Language is a more human-friendly format, which is more readable and more easily editable.
- To save a project as a pbip, go to File – Save As, and change the “Save as type” from “Power BI file (*.pbix)” to “Power BI project files (*.pbip)”.
- When you have done so, the title bar shows “(Power BI Project)”.
 - If you click on the title bar, you will see the report and semantic model links and display names.
- You can then upload it to a Git branch.

8. Plan and implement deployment solutions

- You can create a deployment pipeline of between 2 and 10 stages (workspaces).
 - The workspaces must reside on a Fabric capacity.
- They would generally be in the categories of:
 - Development – create/design new content,
 - Test – release to testers, and
 - Production – share final version.
- It can be used to deploy items including:
 - Activator,
 - Dashboards,
 - Data pipelines,
 - Power BI dataflows (But not Dataflow Gen2),
 - Datamarts,

DP-600: Fabric Analytics Engineer Associate
Manage the analytics development lifecycle

- Environment,
- Eventhouse and KQL databases,
- EventStream,
- Lakehouse,
- Mirrored databases,
- Notebooks,
- Reports,
- Semantic models which originate from .pbix files (Power BI Desktop)
- SQL Databases, and
- Warehouses.
- To create a pipeline:
 - Go to Workspaces, and click “Deployment pipelines” (near the bottom).
 - Click “Create pipeline”.
 - Enter a name and optional description in the “Create a deployment pipeline” dialog box.
 - Enter the pipeline stages.
 - By default, there are 3 stages named Development, Test and Production.
- Pipeline admins who are also Workspace Admins can then assign workspaces.
 - In the pipeline, you should then select the workspaces next to the pipeline stage and click “Assign a workspace”.
 - Note: a workspace can only be assigned to one pipeline.
- Pipeline admins who are or are not Workspace Admins can unassign a workspace from a pipeline stage. To do this:
 - Open the pipeline,
 - In the relevant stage, click the ... and select Unassign workspace, then click Unassign.
- You can compare stages by looking at “Compare” next to a stage.
 - The icon compares that stage with the next stage. It shows:
 - Green check – metadata for all items in both stages is the same,
 - Orange – either some items have changed/updated, or the number of items are different.
 - Where it is orange, you can click on the Compare link to compare the items. It will show:
 - New – new item in the source stage,
 - Different – exists in both stages, but has been changed in the last deployment. This includes if you have changed folder location.
 - "Not in previous stage" – new item in the target stage.

- Note: deployment pipelines keep track of items which have been copied. Even if items have been renamed, they can still be paired together, and therefore compared.
- If something has been changed, then there is a “Review changes” button, which allows you to see the changes the item, either side-by-side or inline.
- To deploy content, you can either:
 - Click on “Deploy to X” – this deploys all content to the next stage,
 - Click on “Show more”. You can then select specific items to be deployed.
 - You can then add a note and click “Deploy”.
 - You can deploy content backwards. This is usually used when setting up the pipeline, as this only works if the previous stage is empty.
- To view the deployment history:
 - Go to the pipeline,
 - Click on “Deployment history”. It shows:
 - Deployed to – stage,
 - Date/time – at the end of the deployment,
 - Deployed by – person or service principal,
 - Items – the new, different and unchanged items, and the items which failed to deploy.
 - A note (if it exists)
 - Deployment ID
 - Deployment Status (Successful/Unsuccessful)
- When you deploy items from a previous stage to a later stage:
 - if any content has the same name in both stages, the content will be overwritten in the later stage.
 - Content in the later stage that is not in the earlier stage will remain (will not be deleted).
 - Up to 300 items can be deployed in a single deployment.
 - You can group items together in folders.
- If you are deploying (for example) a report and not the semantic model it relies on, then:
 - If the semantic model exists in the later stage, it will connect to the later stage model.
 - If the semantic model doesn’t exist in the later stage, then the deployment will fail.
 - Note: you cannot download a .pbix file after deployment.
 - You cannot deploy semantic models which have real-time data connectivity.
- Any user (free user) can view the list of pipelines.
- To create the pipeline, you would need the “pipeline admin” permission (as a minimum permission) in a Pro, Premium Per User, or Premium Capacity.

DP-600: Fabric Analytics Engineer Associate
Manage the analytics development lifecycle

- To give a user the "pipeline admin" permission, go to "Manage Access" and click on "Add people or groups".
- It allows:
 - Create a pipeline
 - View/share/edit/delete the pipeline,
 - Unassign a workspace from a stage,
 - Can see workspaces that are assigned to the pipeline,
 - View deployment history,
 - View the list of items in a stage,
 - Manage pipeline settings,
 - Add/remove a pipeline user
- Pipeline admins who are also Workspace Contributors can also:
 - Compare two stages
 - View or set a rule
 - Deploy items to a target stage, if:
 - The user is a workspace contributor/member/admin of both workspaces, or
 - The target stage is unassigned, and the user is a workspace contributor/member/admin of the source workspace.
- Pipeline admins who are also Workspace Members can also:
- Pipeline admins who are also Workspace Admins can also:
 - Assign a workspace to a stage
- Pipeline admins who are not Workspace Admins can:
 - Unassign a workspace to a stage.
- You can also configure deployment rules.
- These are used for changing the content but keeping some settings as per the deployment rule. It is used for:
 - Dataflow/semantic model/datamart – to specify the data sources or parameters for the dataflow/semantic model/datamart,
 - Paginated report – to specify the data source, and
 - Notebook – the default lakehouse for a notebook.
- To do this:
 - Next to the pipeline stage, click on the “Deployment rules” button.
 - You can’t create it in the first stage – it’s for the target stage.
 - Select the items to create the rule for.
 - Click on “+Add rule” next to:

- “Data source rules” – select from a list, or select Other and manually enter the new data source (of the same type).
 - This can be used for dataflows, semantic models, datamarts, paginated reports and mirrored databases.
 - You can use the following data source types:
 - SSAS or Azure Analysis Services,
 - Azure Synapse,
 - SQL Server or Azure SQL Server,
 - Odata Feed,
 - Oracle,
 - SapHana (using import mode, not direct query)
 - SharePoint, and
 - Teradata.
- “Parameter rules” – select the parameter and enter the value.
 - This can be used for dataflows, semantic models and datamarts.
- “Default lakehouse rules” – select the lakehouse to connect the notebook to in the target stage.
- Note:
 - if you delete an item, its rules are deleted as well, and cannot be restored.
 - if you unassign and reassign a workspace, its rules are lost.

9. Perform impact analysis of downstream dependencies from lakehouses, data warehouses, dataflows, and semantic models

- To create an impact analysis, either:
 - Open the item and click on Lineage – Impact Analysis, or
 - Go to the workspace, click on ... and select “View lineage”.
- You can see:
 - “Child items”: Direct children of the item (things which are directly dependent on it), or
 - “All downstream items”: All affected dependent items downstream.
- You can browse by item type or by workspace.
- If you are making changes, you can click on “Notify contacts” to notify the contact lists of any relevant workspaces.

10. Connect to and query semantic models by using the XMLA endpoint

- XMLA (XML for Analysis) allow you to connect from Power BI Service.
- It is available for Power BI Premium, Premium Per User, and Power BI Embedded workspaces.
- To enable it:
 - Go to Settings – Admin portal – Tenant settings,

DP-600: Fabric Analytics Engineer Associate
Manage the analytics development lifecycle

- Enable Integration settings – Allow XMLA endpoints and Analyze in Excel with on-premises semantic models.
 - If you want to “Analyze in Excel”, so you also Enable “Users can work with semantic models in Excel using a live connection”.
- By default, it is read-only.
 - So you can query semantic model data, metadata, events and schema.
- You can change it to read-write.
 - So you can perform management, governance, debugging, monitoring, and advanced semantic modeling.
 - To enable this in a Premium capacity:
 - Go to Settings – Admin portal,
 - Go to Capacity settings – Power BI Premium – [capacity name],
 - In Workloads, change the XMLA Endpoint setting to “Read Write”.
 - To enable this in a Premium Per User:
 - Go to Settings – Admin portal,
 - Go to Premium Per User,
 - In Semantic model workload settings, change the XMLA Endpoint setting to “Read Write”.
- To get the workspace connection URL.
 - In the workspace, go to ... - Workspace Settings.
 - In the General tab, click on “Copy” under “Workspace Connection”.
- It is the format:
 - `powerbi://api.powerbi.com/v1.0/[tenant]/[workspace]`
- If you need to specify an “Initial Catalog” (for example, in SQL Server Profile), use the semantic model name.
- The client applications for read-only include:
 - Microsoft Excel,
 - Power BI Report Builder,
 - DAX Studio
- The client applications which need read-write include:
 - Visual Studio with Analysis Service projects,
 - SQL Server Profiler version 18.9+,
 - Analysis Services Deployment Wizard, and
 - PowerShell cmdlets.
 - Please note: if you write to a semantic model from Power BI Desktop, you will no longer be able to download it back as a PBIX file.
- The client applications which can use either read-only (for query operations) or read-write (for scripting metadata) include:

DP-600: Fabric Analytics Engineer Associate
Manage the analytics development lifecycle

- SSMS version 18.9+,
- Tabular Editor,
- ALM Toolkit.
- To connect in Power BI Desktop:
 - The recommended way is:
 - Go to Home – Power BI semantic model.
 - Select the semantic model (you don't need the workspace URL).
 - Alternatively:
 - Go to Get Data – Analysis Services,
 - Enter the workspace name,
 - Use “DirectQuery” (not “Import”),
 - In the Navigator, select the semantic model.
- To connect in SSMS:
 - Go to Connect – Connect to Server,
 - Change the “Server type” to “Analysis Services”.
 - In “Server name”, enter the workspace URL.
 - In “Authentication”, choose “Microsoft Entra MFA” (Multi-factor authentication).
 - In “User Name”, enter your user ID.
- In SSMS, you can process individual databases (semantic models), tables, or partitions by right-hand clicking on the database/table/partition.
- You can choose from the following process modes:
 - Process Default
 - Performs the processing which is necessary to get it to a fully processed data.
 - For example:
 - data for empty tables/partitions are loaded,
 - Hierarchies, calculated columns, and relationships are recalculated/build/re-built.
 - Process Full
 - Drops all existing data, and then processes the object.
 - This is required when a structural change has been made.
 - It requires the most resources.
 - Process Data (for a table or partition only)
 - Loads data only.
 - Does not rebuild hierarchies, relationships.
 - Does not recalculate calculated columns or measures.

- Process Clear
 - Removes all data, either from the database, the partition, or from the table and table partitions.
- Process Recalc (for a semantic model only):
 - Updates/recalculates hierarchies, relationships, and calculated columns.
- Process Defrag (for a table only):
 - Defragments table indexes.
- Process Add (for a partition only):
 - Incrementally updates partition with new data

11a. Create and update reusable assets, including Power BI template (.pbix) files

- Templates contain:
 - Report pages and visuals,
 - The data model definition, which includes:
 - Schema,
 - Relationships, and
 - Measures.
 - Any query definitions, which includes:
 - Queries,
 - Query Parameters.
- Templates do not contain data.
- To create a template, go to File – Export – Power BI template.
 - You can enter a description for your template.
 - Select your file location and name for your .PBIX file.
- To use a template, either:
 - Double-click on a .PBIX file in Windows Explorer, or
 - Go to File – Import – Power BI template.
 - Go to File – Open, "Browse this device".
- You can then:
 - Enter values for any parameters,
 - Enter the file location for any data sources if necessary.
 - You can then connect the data based on your credentials.

11b. Create and update reusable assets, including Power BI data source (.pbids) files

- .pbids files contain a JSON structure which point towards a single Power BI data source.
- To create a PBIDS File in Power BI Desktop:
 - go to File – Options and settings – Data source settings.

- click on your Data source settings, and click on “Export PBIDS”.
- Give the file a name and location, and click Save.
- Note: if columns are encrypted in the data source, then it will generate an error.
- To open the .pbids file, double-click on it in Windows Explorer.
- When you open the .pbids file, you will be asked for any necessary credentials to open.
- You would then select any tables from that data source, and possibly the database and connection model if it isn’t part of the .pbids file.

11c. Create and update reusable assets, including shared semantic models

- To share a semantic model, either:
 - go to the semantic model and click Share (at the top) or “Share semantic model” (in the main section), or
 - go to the OneLake data hub, click on the ... next to the item, and click Share.
- Enter the names or email address that you want to share the semantic model with. You can select:
 - Allow recipient to modify/share this semantic model,
 - Allow recipient to build content with the data associated with this semantic model,
 - Send an email notification.
 - Add an optional message, and then click “Grant access”.
- You manage permissions by:
 - going to the Workspace, click on ... next to the semantic model and go to “Manage permissions”.
 - going to the semantic model and clicking on File – “Manage permissions”,
 - going to the semantic model, clicking on Share, then the ... and Manage permissions.
 - going to the OneLake data hub, click on “Manage permissions”.
- To use the semantic model:
 - in Power BI Desktop:
 - go to Home – OneLake data hub – Power BI Semantic Model, or go to Home – Get data – Power BI semantic models
 - select the semantic model, and click Connect.
 - in Power BI Service:
 - go to OneLake data hub, and click on the semantic model.
 - You can then create a report.

Copy data

12. Create a data connection

- Best ways to copy data:
 - If you are uploading a small file(s) from a local machine

- Use a Local file upload
 - You can right-hand click on the ... next to Files, and go to Upload – Upload files/folder.
 - Table names can contain alphanumeric characters and underscores up to 256 characters. No dashes or spaces are allowed.
 - Column names allow upper/lower cases, characters in other languages like Chinese, and underscores up to 32 characters.
- You can also use the OneLake file explorer app. It integrates OneLake with Windows File Explorer.
 - You can download it from <https://www.microsoft.com/en-us/download/details.aspx?id=105222>
- It adds this location into Windows Explorer, and includes a Sync column, showing the synchronization status, showing:
 - Blue cloud icon – online only,
 - Green tick – downloaded to your computer,
 - Sync pending arrows – in progress.
- If you are uploading a small amount of data, or using a specific connector (from over 200 connectors), or want to use Power Query transformations
 - Use a Dataflow
- If you have a large data source without using any data transformations
 - Use the Copy tool in a pipeline
- If you have got complex data transformations
 - Use Notebook code

13. Discover data by using OneLake data hub and real-time hub

- The OneLake data hub is where you can find Fabric items to which you have access.
 - It used to be called the Power BI Data Hub.
- You can select from:
 - All items (that you have access to),
 - My items (items you own),
 - Endorsed items (organization items – certified data items first, then promoted data items).
 - You can promote items by clicking on ..., go to Settings, then the Endorsement tab.
 - Favorites (items you have marked as favorite).
 - You can favorite items in the workspace by clicking on ..., and selecting Favorite.
- You can also filter by workspace that you have access to.

- Only items that you have access to and are discoverable will be displayed.
- You can also filter by type:
 - Data types, such as Eventhouse, KQL Database, Lakehouse, Semantic model, SQL analytics endpoint, SQL database and Warehouse.
 - Insights,
 - Processes,
 - Solutions, and
 - Configurations.
- You can either click on an item to show the Items Details view, or click on the ... next to it for options.
- The Item Details view include:
 - Overview tab, containing Location, last data update date/time, Owner, Sensitivity Label (if any), and tables if appropriate.
 - Lineage tab, containing items which are upstream (items which it relies on) or downstream (items which rely on it).
 - Monitor tab, containing the item's activities.
- You can also open it, and there are additional options in the menu.
- The Real-Time hub shows all data-in-motion.
- You can access:
 - All data Fabric eventstreams and KQL Database tables that you can access,
 - My data streams (that you brought into Fabric),
- For streams, you can click on the ... and select:
 - Preview data,
 - Open eventstream, and
 - Endorse.
- For KQL tables, you can click on the ... and select:
 - Explore data,
 - Create real-time dashboard,
 - Open KQL Database, and
 - Endorse.
- Clicking on "Connect to – Data sources" allows you access:
 - Google Cloud and Amazon Kinesis Data Streams,
 - Kafka Clusters, and
 - Database Change Data Capture (CDC) feeds:
 - Azure SQL Database CDC
 - PostgreSQL Database CDC

- MySQL Database CDC
- Azure Cosmos DB CDC
- Azure SQL Managed Instance CDC
- SQL Server on virtual machine (VM) DB CDC
- Microsoft sources:
 - Azure Event Hubs,
 - Azure IoT Hub, and
 - Azure Service Bus.
- Automatically generated Fabric workspace item events, and
- Azure Blob Storage events.
- The other options show the Microsoft sources, Fabric events, and Azure events.

14. Ingest or access data as needed

- To create an Eventhouse:
 - Either
 - Go to a Fabric-enabled workspace.
 - Click on "+New item", and click on "Eventhouse".
 - or
 - Go to the Real-Time Intelligence experience (bottom-left hand corner), and
 - In Home, in "Recommended items to create", click on "Eventhouse"
 - Enter a name of the Eventhouse.
 - After a few seconds, the Eventhouse will open, together with a KQL database with the same name as the database.
 - An Eventhouse stores multiple databases, maybe over the same project, and is used for real-time data streams and event-based data, such as log data (maybe security and compliance logs), telemetry data, time series data (maybe financial records), and IoT data.
 - To create a new database:
 - Go to the Eventhouse,
 - Click on the + next to "KQL databases".
 - Enter a database name, and keep the Type as "New database (default)".
- To add data:
 - In the Eventhouse, click on the ... next to the Database (not the queryset), and go to Get Data. You can then get the following data:
 - One time import:
 - Sample. These include:
 - Stock analytics,
 - Weather analytics,

DP-600: Fabric Analytics Engineer Associate

Copy data

- IoT analytics,
- Log analytics,
- Metrics analytics,
- Automotive (taxi) analytics, and
- Azure PlayFab game analytics.
- Local file,
- OneLake,
- Azure Storage
- Continuous import (stream):
 - Real-time hub,
 - Event Hubs,
 - Existing or New Eventstream,
 - Pipeline and
 - Amazon S3.

	Eventhouse	Lakehouse	Warehouse
Primary purpose	For event-based, time series or IoT data, especially in real-time, using KQL, and then create visualizations. Can also query in SQL.	Tables and Files, for data engineers to query/modify using PySpark (and Spark SQL) in notebooks. Additionally, querying data using SQL using an SQL analytics endpoint.	For data analysts to query and modify using SQL.
Type of data	Unstructured, semi-structured, structured		Structured
Primary developer persona	Data scientist, data engineer		Data warehouse developer, data architect, database developer
Primary dev skill	No code, KQL, SQL	Spark (Scala, PySpark, Spark SQL, R)	SQL
Data organized by	Databases, schemas, and tables	Folders and files, databases, and tables	Databases, schemas, and tables
Read operations	KQL, SQL, Spark	Spark, SQL	SQL (and Spark using shortcuts)
Write operations	KQL, Spark, connector ecosystem	Spark (Scala, PySpark, Spark SQL, R)	SQL
Multi-table transactions	Yes, for multi-table ingestion	No	Yes
Primary development interface	KQL Queryset, KQL Database	Spark notebooks, Spark job definitions	SQL scripts
Security	RLS	RLS, CLS**, table level (T-SQL), none for Spark	Object level, RLS, CLS, DDL/DML, dynamic data masking
Can be a source for shortcuts	Yes	Yes (files and tables)	Yes (tables)
Advanced analytics	Time Series native elements, full geo-spatial and query capabilities	Interface for large-scale data processing, built-in data parallelism, and fault tolerance	Interface for large-scale data processing, built-in data parallelism, and fault tolerance
Advanced formatting support	Full indexing for free text and semi-structured data like JSON	Tables defined using PARQUET, CSV, AVRO, JSON, and any Apache Hive compatible file format	Tables defined using PARQUET, CSV, AVRO, JSON, and any Apache Hive compatible file format
Ingestion latency	Queued ingestion, streaming ingestion has a couple of seconds latency	Available instantly for querying	Available instantly for querying

14a. Copy data by using a data pipeline

- You can use the Copy data assistant:
 - In the pipeline, click on "Copy data" or go to Home or Activities – Copy data – Use copy assistant
 - Source
 - Select a data source, including sample data
 - Enter your connection settings, either using an "Existing connection" or "Create new connection".
 - Choose the specific data to be transferred (for example, file/folder).

- Select a data destination source
 - Select a data source
 - Enter your connection settings, either using an "Existing connection" or "Create new connection".
 - Map your data to the destination.
- Review the details, and click OK to save.
 - It will then be added to your data pipeline canvas.
 - Advanced settings will be available in the tabs.
- You can also a copy activity.
 - Go to Home or Activities - Copy activity – Add to canvas
 - In the general tab, you can select:
 - Name and Description, and whether it is enabled (in the Activity state),
 - Timeout – how long the activity can run. The default is 12 hours. It shown in the format D.HH:MM:SS.
 - Maximum number of retry events,
 - Number of sections between each retry attempt,
 - "Secure output/input". When this is checked, details of the activity is not logged.
 - In the Source tab:
 - select an existing connection, or click on +New to create a new connection.
 - In a dialog box, you can select the data source and connection.
 - Back in the source tab, you can select more details, depending on the connection type – for example, the connection type, user query (table/query/stored procedure) or root folder, and table.
 - There are more settings in the Advanced section.
 - In the Destination tab:
 - select the connection, and more details.
 - In the advanced section, you can select more settings, such as:
 - Max rows per file,
 - Table action – Append or Overwrite, and
 - Max concurrent connections.
 - In the Mapping tab, you can select the mapping from the source table to the destination table.
 - This allows you to map between columns which are differently named in the two sources.
 - In the Type conversion settings, you can select:

DP-600: Fabric Analytics Engineer Associate
Copy data

- Allow data truncation (for example, from decimal to integer, or DateTimeOffset to Datetime),
- Treat Boolean as number (true = 1),
- Date and DateTime format (for example "yyyy-MM-dd HH:mm:ss.fff").
- DateTimeOffset format (for example "yyyy-MM-dd HH:mm:ss.fff zzz").
- TimeSpan format (for example "dd.hh:mm:ss")
- Culture (for example, "en-us", "fr-fr")
- In the Settings tab, you can select:
 - Intelligent throughput optimization. Choose from Auto (which is dynamic based on the sources and destinations), Standard, Balanced and Maximum.
 - Degree of copy parallelism,
 - Fault tolerance – what happens if there are errors while copying.
 - Enable logging – log copied files and skipped files and rows,
 - Enable staging and Staging account connection (advanced).
- To run the data pipeline, go to Home – Run.
 - You can see the results in the Output tab.
 - You can export the results to CSV.
 - You can filter for a particular "Activity status" (for example, succeeded), hide output columns and show columns for Activity type, Run end, Activity run ID, Source and Destination.
- To schedule it, see topic 24.

14a. Ingest data by using a data pipeline

- Once you have created a data flow, you can incorporate it into a data pipeline.
- In the workspace, go to New – Data pipeline.
- Provide a name for the data pipeline.
- Click on Dataflow activity.
- In the settings tab, select the Dataflow.
- Optionally, you can add an Office 365 Outlook activity to send an email notification.
 - Go to the Activities menu, click on the "Office 365 Outlook" icon, and connect to Office 365 Outlook.
 - Enter an email address, subject and body.
 - Additional properties can be set in the Advanced area.
 - You can then connect this new activity to the previous activity.
- To run the data pipeline:
 - Go to Home - Run or Run – Run in the data pipeline.

- You can see the output in the Output tab.
- To schedule the data pipeline, see topic 24.

14b. Copy data by using a dataflow

- See topic 16b.

14b. Ingest data by using a data dataflow

- To ingest data by using a data dataflow:
 - Go to Data Factory.
 - Go to your Fabric-enabled workspace.
 - Go to New – Dataflow Gen2.
 - Go to Home – Get data, and select a data source.
 - You can transform the data, using the Power Query interface.
 - Go to Home – Add data destination, and select either:
 - Azure SQL Database,
 - Lakehouse
 - Azure Data Explorer (Kusto) or
 - Warehouse.
 - Then select:
 - your data destination,
 - the table name,
 - The update method
 - Whether data is being appended, or replaced.
- You can optionally export this dataflow as a template:
 - Go to Share – Export template
 - Add a name and an optional description, and click OK.
 - To use it again, create a Dataflow Gen2
- To schedule the refresh:
 - Go to the workspace and click on the ... next to the Dataflow.
 - Go to settings.
 - Expand Refresh.
 - Change "Configure a refresh schedule" to On.
 - Select the refresh frequency - either Daily or Weekly.
 - If Weekly, select which day(s) of the week.
 - Click "Add another time" and select a time
 - This can be on the hour or on the half hour.
 - Add additional times if required.
 - Under "Send refresh failure notifications to", check or uncheck:

- Dataflow owner, and
- These contacts (and add contacts)
- Click "Apply".

14b. Implement Fast Copy when using dataflows

- If you need to copy a lot of data, then you can use the “Fast copy” option.
- This gives you the Graphical User Interface of a dataflow, with the speed of the “Copy Activity” pipeline.
- You can only use the following incoming connectors:
 - Azure Data Lake Storage Gen2 (using .csv or parquet files),
 - Azure Blob storage (using .csv or parquet files),
 - Azure SQL Database,
 - Lakehouse,
 - PostgreSQL, and
 - On premises SQL Server.
- You can only use a Lakehouse as the output destination.
 - If you want other destinations, then you can use the results of this query in another query.
- It only supports the following transformations:
 - Combine files,
 - Choose columns,
 - Change data types,
 - Rename columns,
 - Remove columns.
- To enable fast copy in a dataflow Gen2:
 - Go to Home – Options,
 - In the “Scale” tab, check “Allow use of fast copy connectors”.
- It will only be used when the data size goes above:
 - 100 Mb in Azure Data Lake Storage Gen2 or Azure Blob storage,
 - 5 million rows in Azure SQL Database or PostgreSQL.
- However, you can override this threshold by right-hand clicking on the query and checking “Require fast copy”.
 - This can only be done if “Allow use of fast copy connectors” has already been checked.
- You can see if it is used by hovering over the first applied step.
 - It would say “This step is going to be evaluated with fast copy”.

14c. Copy data by using a notebook

- Continuing from topic 16c.

- To save the dataframe as a delta lake, parquet table to Tables section of the default Lakehouse
 - `df.write.mode("overwrite or append").format("delta").saveAsTable([DeltaTableName])`
 - Use overwrite to save it as new table, or overwrite an existing table.
 - You can also use "ignore" which ignores the write operation if the file already exists.
 - Instead of using (for example) "overwrite", you can also use (without quotation marks) `SaveMode.Overwrite`.
 - If you don't use anything, the default is "errorifexists" or "error", which returns an error if the file already exists.
 - Use append to add the table to an existing table.
- After saving to a table, you can then create a semantic model by going to "New semantic model" in the Lakehouse (not the notebook).
- In a warehouse, you can create a new semantic model by going to Reporting – New semantic model.
 - New objects are automatically added to its default Power BI semantic model.
 - You may wish to have another model with a more focused list of tables.

14c. Ingest data by using a notebook

- To load data from a file into a dataframe using a notebook:
 - Right-hand click the file (or left-click on the ... next to the table) and select:
 - Load data – Spark or Pandas
 - This would generate the data in your notebook.
 - Copy relative path for Spark
 - This should be used if you are using this data source in this lakehouse.
 - Example: `Files/MtoMAActual.csv`
 - Copy ABFS (Azure Blob Filesystem) path for Spark
 - This should be used if you are using this data source in another lakehouse.
 - Example: `abfss://06e284ed-4b3f-4882-b35c-e96f16f2479f@onelake.dfs.fabric.microsoft.com/655ad85b-0764-4156-a93c-06f825e09a8d/Files/MtoMAActual.csv`
- If the last two options, create a PySpark cell and then enter:
- `df = spark.read.parquet("[Location]")`
- To load CSV, JSON, ORC or Parquet files, use:
 - `df = spark.read.csv("[Path]")` or `.json` or `.orc` or `.parquet`
or
 - `df = spark.read.load("[Path]", format='csv', header=True)`

- To load data from a table into a dataframe, right-hand click the file (or left-click on the ... next to the table) and select Load data – Spark.
- To read it using an explicit data structure, then you can add (for example, after `.read`):
 - `.schema(schemaTarget)`
- The schemaTarget is defined using StructType and StructField. You can generate this for an existing table by using:
 - `df.schema`
- An example of the schemaTarget is:
 - `StructType([StructField('Country', StringType(), True),
StructField('Location', StringType(), True),
StructField('Actual', StringType(), True)])`
- To save a dataframe as CSV or Parquet files to Files section of the default Lakehouse
 - `df.write.mode("overwrite").format("csv or parquet or json").save("Files/[FileName]")`
 - You can also use `(f"Files/{FileName}")`
 - You can use mode first, or format first – it doesn't matter what the order is.
- To read a table in Spark, use
 - `df = spark.read.table("TableName")`
- To read the parquet file with Pandas from the default lakehouse mount point and use:
 - `import pandas as pd`
 - `df = pd.read_parquet("/lakehouse/default/Files/[NameOfFile].parquet")`
- To use load the data using Pandas API:
 - right-hand click on the file, and select “Copy File API path”.
 - Example: `/lakehouse/default/Files/MtoMAActual.csv`
 - Use the following code:
 - `import pandas as pd`
 - `df = pd.read_parquet("[APIPath]/[LakehouseName]/Files/[NameOfFile].parquet")`
- Continued in topic 22c.

16. Implement OneLake integration for eventhouse and semantic models

- You can turn on OneLake integration for eventhouse by:
 - clicking on a KQL database, the word "Tables" or KQL table in an eventhouse, and
 - in the Database details pane, click on OneLake – Availability: On.
 - If switching on for the database or all tables, you get the option for it to be switched on to existing tables, or only for new data.
 - Note: I have had problems switching it on for empty KQL Databases. However, I can switch it when a table has been added.

- This creates a copy of your data in OneLake, allowing you to query your KQL Database in the Delta Lake format in Power BI, Lakehouse, Warehouse, Notebooks etc.
- It may take hours for the data to appear in OneLake.
 - Only new data will be added to OneLake, not existing data.
- There is no cost apart from OneLake storage cost.
- When switched on:
 - Data cannot be removed.
 - The table cannot be renamed.
 - RLS (Row Level Security) cannot be applied.
- To enable OneLake integration for a semantic model, you need a Fabric or Premium capacity:
 - Go to the Settings for the semantic model.
 - Expand "OneLake Integration" and turn it On.
- Administrators may need to enable semantic models, by:
 - going to Settings – Admin portal, and
 - in Tenant settings (under Integration settings) enable "Semantic models can export data to OneLake".
- You can now see it in OneLake File Explorer, and you can create shortcuts from OneLake.

Transform data

17a. Create views

- A view is a statement which has been saved, and can be retrieved.
- To create a view:
 - Go to the SQL analytics endpoint,
 - Write a query.
 - For example, `SELECT * FROM NameOfTable`
 - And then either:
 - Select the query statement, and click on "Save as view", or
 - Prefix the statement with
 - `CREATE VIEW Schema.TableName AS`
 - You can define the column names at the end of the Schema.TableName in brackets
 - You can use a WITH in the query.
- To use the view:
- `SELECT * FROM Schema.TableName`

17b. Create functions

- In Fabric, a CREATE FUNCTION can return a table, which you can use in the FROM clause.

- In Fabric, you cannot return a single value (scalar function).
- The syntax is:
 - *CREATE FUNCTION Schema.FunctionName*
 - *(@Parameter AS ParameterDataType...)*
 - *RETURNS TABLE*
 - *[AS]*
 - *RETURN Select_Statement*
- Parameters use a @ at the beginning, and contain a single value.
 - You cannot pass tables as a parameter.
 - "= default" is a default value for the parameter.
- You can use DECLARE statements creating a local data variable.
- Example function:
 - *CREATE FUNCTION dbo.func_AddressData (@Country AS varchar(20))*
 - *RETURNS TABLE*
 - *AS*
 - *RETURN*
 - *SELECT **
 - *FROM AddressData*
 - *WHERE CountryRegion = @Country*
- Calling the function:
 - *SELECT AddressID, City*
 - *FROM func_AddressData('Canada')*

17c. Create procedures

- A procedure is a sequence of code that you can run outside of a SELECT statement.
 - You can have a single or multiple SELECT statements in a procedure.
 - It allows for input parameters.
 - You can run other procedures from the procedure.
- The syntax is
- *CREATE [OR ALTER]*
- *[PROC or PROCEDURE]*
- *SchemaName.ProcedureName*
- *[@parameter datatype - optional]*
- *AS*
- *[BEGIN – optional]*
- *SQL Statement(s)*

- [END - optional]
- To run the procedure, use
 - *ProcedureName*
 - optional followed by the parameters
 - preceded optionally by *EXEC* or *EXECUTE*.
- Example procedure:
- *CREATE PROC dataproc @passengerCount INT*
- *AS*
- *SELECT **
- *FROM [LakehouseTrial].[dbo].[datatable]*
- *WHERE passengerCount = @passengerCount*
- Example running of the procedure:
- *exec dataproc 2*

19. Implement a star schema for a lakehouse or warehouse

- Included in the PL-300 exam.

21. Aggregate data

KQL

- summarize by GroupCol1, Col3 = GroupCol2, ...
 - This groups by the GroupCols.
- summarize Aggregation[, Col2 = Aggregation, ...] by GroupCol1, Col3 = GroupCol2, ...
 - This groups by the GroupCols, using the Aggregations (which could be renamed).
- avg(expression) – Mean Average of expression.
- avgif(expression, condition) - Mean Average of expression when condition is true.
 - Use == to compare in a condition
 - = is for assigning a value
- count – Number of rows
 - | count
 - | summarize CountRows=count() by Category
- countif(condition) – Number of rows when expression is true
- dcount(expression) and count_distinct(expression)
 - Counts the number of distinct values.
 - dcount gives an approximation, but is quicker than count_distinct.
 - dcount can also use an optional second "accuracy" argument from 0 (less accurate) to 4 (more accurate). The default is 1.
 - Recommend using dcount, as count_distinct is not recognized by Azure Data Explorer.

- dcountif and count_distinctif filters on the second argument.
- max(expression) and min(expression) give the maximum and minimum value.
 - maxif and minif filters on the second argument.
- sum calculates the total.
 - sumif also filters on the second argument.

Aggregation	Description
avg, avgif	Mean average
count, countif	Counting the number of rows
dcount, dcountif	Distinct counting, eliminating duplications
max, maxif	The biggest value of numbers, text, datetime and bool
min, minif	The smallest value of numbers, text, datetime and bool
sum, sumif	The total

22a. Merge data

SQL

- To merge data in SQL, you can use:
 - *SELECT **
 - *FROM firstTable*
 - *UNION [ALL]*
 - *SELECT **
 - *FROM secondTable*
 - The names do not need the same – it merges by position.
- You can also merge data using a Dataflow Gen2 (in the Power Query environment) by using Home – Append Queries, just as in Power BI.

KQL

- Unioning two tables together:
union [kind = JoinType] [withsource = SourceTable] Table1, Table2...
 - Returns a table which is a combination of 2 or more tables.
 - JoinType is either inner or outer:
 - inner returns only those columns which are in all tables.
 - outer returns all columns in all tables.
 - withsource adds an extra column with the relevant table name.
 - If the row was taken from Table1, then this "SourceTable" extra column will contain "Table1".
 - The tables can include wildcards (for example TableName* would be all tables in the database which start with "TableName").
- You can use a calculated table by enclosing it in brackets/parentheses.

- Table | union (Table | where...)

22b. Join data

SQL

- To join data in SQL, you can use:
- SELECT *
- FROM firstTable
- INNER/LEFT/RIGHT JOIN secondTable
- ON firstTable.column = secondTable.column
- You can also join data using a Dataflow Gen2 (in the Power Query environment) by using Home – Merge Queries, just as in Power BI.

KQL

- Joining two tables together:
LeftTable | join [kind = JoinType] RightTable on Conditions
 - Note that there is only one equal sign after "kind". kind includes:
 - innerunique – the default Join – deduplicated LeftTable which matches with RightTable.
 - inner - LeftTable which matches with RightTable. You can also leftsemi and rightsemi to return just one table's columns.
 - leftouter and rightouter – all rows in one table, plus those that match in the other table. Unmatched rows returns "null" from the other table.
 - You can also use leftanti (or anti), leftantisemi, rightanti and rightantisemi to find the rows in one table which do not match in the second table. It will return just that one table's columns.
 - fullouter – all rows in both tables.
 - Conditions show the columns which are in both tables.
 - If the column names are the same in both tables, then you can just use "ON" followed by the column name – for example, "ON Col1".
 - If the column names are different, then you should use: ON \$left.Col1 == \$right.Col2, where \$left is LeftTable and \$right is RightTable.
 - You can specify multiple conditions using AND or separate by commas.
 - For best performance using "join", use the smaller table as LeftTable.
- lookup
 - If the relationship is straightforward, and you are using leftouter or inner, then "lookup" is quicker than "join".
 - When using "lookup", "leftouter" is the default kind.
 - "lookup" does not support many-to-many; only one-to-one and one-to-many are supported.
 - The RightTable cannot be big – not more than tens of Megabytes.
 - For best performance using "lookup", the smaller table should be RightTable (the opposite of "join").

23a. Identify and resolve duplicate data

SQL

- To identify duplicate data in SQL, you can GROUP BY the data, then use a HAVING COUNT(*)>1

KQL

- You can identify duplicate data by using a summarise, then a count() > 1. This is equivalent of Having.
 - MyTable
 - | summarize Count = count() by ID, Name, Value
 - | where Count > 1
- To remove duplicate data, then either use:
 - | summarize, or
 - | distinct.

23b, c. Identify and resolve missing data and null values

KQL

- To identify null values, you can use:
 - | where isnull(column_name).
- To identify missing data, you can join between two (or more table), and use:
 - | where isnull(column_name) for one of the tables, or
 - join use leftsemi or rightsemi to find data in one table which is not in the other table.

24. Convert column data types

SQL

- In Spark SQL and the SQL Analytics endpoint, you can use:
 - cast(column_name as Date)
- In the SQL Analytics endpoint only (not in Spark SQL), you can also use:
 - convert(Date, column_name)
- "format" uses:
 - yy or yyyy (not capitalized) – year
 - Q – quarter of year
 - d – day of month
 - E – day of week ("Tue" or "Tuesday")
 - D – day of year
 - M (capital – otherwise, it would be minute) or L – month of year
 - M is the "standard" form and L the "stand-alone" form, which may be different in some languages (for example, Russian)
 - M or L = 1 or 12

DP-600: Fabric Analytics Engineer Associate
Transform data

- MM or LL = 01 or 12
- MMM or LLL = Jan
- MMMM or LLLL = January
- h – hour of day (1 to 12)
- H – hour of day (0 to 23)
- K – hour of day (0 to 11)
- k – hour of day (1 to 24)
- m – minute of hour
- s – second of minute
- S (1 to 9 characters) – fractional second
- a – am or pm
- VV – time-zone ID (America/Los_Angeles; Z; -08:30)
- z (1 to 3) – time-zone name (Pacific Standard Time; PST)
- O (1, 2 or 4) – offset ("GMT+8" or "GMT+08:00" or "UTC-08:00")
- X (1 to 5) – zone-offset (Z; -08; -0830; -08:30; -083015; -08:30:15)
- x (1 to 5) – zone-offset (+0000; -08; -0830; -08:30; -083015; -08:30:15;)
- Z (1 to 5) – zone-offset (+0000; -0800; -08:00;)
- ' – escape for text
- " – string literal
- The data types are:
 - tinyint: -128 to 127
 - smallint: -32,768 to +32,767
 - int: -2,147,483,648 to 2,147,483,647
 - bigint: -9223372036854775808, 9223372036854775807
 - decimal or decimal(p, s) (or numeric for SQL)
 - p = precision – number of digits
 - s = scale – number of decimal places
 - The default is (10, 0)
 - float – single precision floats
 - double for PySpark (or float for SQL) – double precision floats
 - string (for PySpark), char(n) and varchar(n)
 - strings, with a maximum length of "n".
 - varchar allows for a variable length.
 - bool (for PySpark)
 - timestamp (for PySpark) and datetime2 (for SQL) – date and time
 - date (and time for SQL)

- The SQL data types are:

Description	SQL
0 to 255	tinyint
-32,768 to +32,767	smallint
-2,147,483,648 to 2,147,483,647	int
-922337203685477580 to 9223372036854775807	bigint
decimal or decimal(p, s)	or numeric
floating numbers	float and real
strings	char(n) and varchar(n)
boolean	bit
date and time	datetime2
date	date
time	time

KQL

- In KQL, you can use:
 - tobool(value): Convert inputs to boolean (signed 8-bit) representation.
 - todatetime(value): Converts input to datetime scalar.
 - toreal(Expr): Converts the input to a value of type real.
 - toString(value): Converts input to a string representation.
 - totimespan(value): Converts input to timespan scalar.
 - format_datetime(datetime(2017-01-29 09:00:05), 'yy-MM-dd HH:mm:ss')
- The formats for format_datetime are:

DP-600: Fabric Analytics Engineer Associate
Transform data

Format specifier	Data Type	Description
y or <u>yy</u>	datetime	The year, from 0 (or 00) to 99.
yyyy	datetime	The year as a four-digit number.
M and MM	datetime	The month, from 1 (or 01) through 12.
d and dd	datetime	The day of the month, from 1 (or 01) through 31.
d to dddddddd	timespan	Number of days, with extra zeros if needed.
h and hh	datetime	The hour, using a 12-hour clock from 1 (or 01) to 12.
H and HH	Both	The hour, using a 24-hour clock from 0 (or 00) to 23.
m and mm	Both	The minute, from 0 (or 00) through 59.
s or ss	Both	The second, from 0 (or 00) through 59.
f to ffffffff	Both	Fractions of a second in a date and time value.
F to FFFFFFFF	Both	If non-zero, fractions of a second in a date and time value.
tt	datetime	AM / PM hours

- Data types
 - bool
 - true, false, bool(true) and bool(false)
 - datetime
 - This is a date from 1 AD to 9999 AD. Can include a time
 - Examples:
 - datetime(2030-03-02),
 - datetime(2030-03-02 01:23:45.6)
 - You can add/subtract timespans to dates – for example, + 1d
 - You can also use:
 - now([offset]) – the current UTC time. You can use an optional "offset" timespan to add to the current time.
 - ago(timespan) – A datavalue time, which is the difference between now() and the "timespan" argument.
 - For example: ago(1h)
 - decimal
 - A 128-bit decimal number.
 - Example: decimal(4.5).
 - It is a lot slower than the "real" data type.
 - dynamic
 - Contains a value or an array of values.
 - Examples:
 - dynamic([3, 4, "hi"])
 - dynamic(6)
 - int
 - A signed 32-bit integer.
 - Examples: int(5), int(-7)
 - long
 - A signed 64-bit integer. The default type for integers.
 - Examples: long(5), long(-7), 9
 - real
 - A 64-bit, double-precision, floating-point number. The default type for decimal numbers and numbers expressed in scientific notation.
 - Examples: 1.23, 2e6, real(9.87)
 - string
 - Unicode characters, surrounded by single or double quotes (speech marks).
 - If you use single quote marks, you can use double quotes in the string (and vice versa).
 - For example: 'She said "hello".'
 - Can include:
 - \t – tabs,
 - \n – new line,

DP-600: Fabric Analytics Engineer Associate
Transform data

- \u – a Unicode character, specified by the next 4 hexadecimal characters – for example, \u0020 for a space (character number 32).
- \\ - a backslash
- If you don't want \s to be treated like that, you can use @ outside of the quote marks.
 - Example: @'hello\hi', @"C:\folder\file.txt"
- If you want to use a multi-line string literal, enclose it with three ` s - ```String```
- timespan (also known as time)
 - A time interval, which can be represented by d, h, m, s, ms, microsecond and tick for days, hours, minutes, seconds, milliseconds, microseconds and nanoseconds.
 - For example: 3d, 2.5h, 15m, 40s
 - You can also use timespan(3) for 3 days, timespan(40 seconds), and timespan(1.23:45:17.8) for days to seconds.
 - You can add or subtract two timespans to create a timespan data type.
 - You can multiply a timespan by a number to create a timespan data type.
 - You can also divide two timespans to return a real data type.
- null
 - A missing value.
 - Any data type can be used except string to represent this – for example, datetime(null).
- Comment
 - Use // to add a comment line.

Data type	Description	Example
bool	True/False	true, false, bool(true) and bool(false)
datetime	A date from 1 AD to 9999 AD. Can include a time.	datetime(2030-03-02), datetime(2030-03-02 01:23:45.6)
decimal	A 128-bit decimal number. A lot slower than "real".	decimal(4.5)
dynamic	A value or an array of values.	dynamic([3, 4, "hi"]), dynamic(6)
int	A signed 32-bit integer.	int(5), int(-7)
long	A signed 64-bit integer. The default type for integers.	long(5), long(-7), 9
real	A 64-bit, double-precision, floating-point number. The default for decimal/scientific numbers.	1.23, 2e6, real(9.87)
string	Text surrounded by single or double quotes, or ``` for strings over multiple lines.	'She said "hello".' ```Multi-line string```
timespan	A time interval expressed in d, h, m, s, <u>ms</u>	3d, 2.5h, 15m, 40s, timespan(3) [days], timespan(40 seconds), timespan(1.23:45:17.8)

25. Filter data

KQL

- Comparison using | where
 - You can also use "| filter" – they are the same.

- Comparisons:
 - == and != for equality and not equal (case-sensitive)
 - =~ and !~ for equality and not equal (case-insensitive). This is slower than case-sensitive versions (== and !=).
 - < and <= for less than, and less than or equal to
 - and >= for greater than, and greater than or equal to
 - "contains", "has", "startswith" and "endswith":
 - Example which equates to true: "long string" contains "Str"
 - contains looks for smaller strings within a larger string.
 - startswith and endswith looks for strings at the start/end of a string.
 - has looks for a complete word within a larger string.
 - hasprefix and hassuffix looks for strings at the beginning/end of a word.
 - An "!" at the beginning of a function looks for strings which do not match. For example, !has looks for one small string which is NOT in a large string.
 - All of the above are case insensitive functions. Add "_cs" for a case sensitive function (such as "has_cs").
 - has_any and has_all looks for multiple small strings within a larger string.
 - "long string" has_all ("long", "string", "long")
 - has_any will return true if any of the smaller strings are in the larger string.
 - has_all will return true if all of the smaller strings are in the larger string.
 - "expression1 and expression2" looks for both expressions to be true.
 - "expression1 or expression2" looks for at least one expression to be true.
 - not (expression) looks for "expression" to be not true.
 - You can also use not(expression) with brackets – they are optional.
- between (leftRange .. rightRange)
 - Examples
 - | where x between (50 .. 55)
 - | where dateTime between (datetime(2031-01-01) .. datetime(2031-03-31))
 - leftRange and rightRange can be int, long, real or datetime
- !between (leftRange .. rightRange) – not in the range
- in, in~
 - For example: "yellow" in ("red", "yellow", "blue")
 - Whether a value is equal to one of the elements.

DP-600: Fabric Analytics Engineer Associate
Query and analyze data

- in~ is case-insensitive. in is case-sensitive.
- !in and !in~
 - Whether a value is not equal to any of the elements.
 - !in~ is case-insensitive. !in is case-sensitive.
- Limiting the number of rows:
 - | take 10 – retrieves the top 10 rows.
 - Can also use "limit" instead.
 - | sample 10 – retrieves a random 10 rows.
 - It can retrieve different rows each time.
 - | top 10 by expression [asc/desc] [nulls first/last]
 - retrieves the top 10 rows ordered by "expression".
 - asc and desc order by Ascending/Descending.
 - nulls first/last show where the nulls should be – before or after non-null values.
 - | distinct Col1, Col2, Col3...
 - Returns distinct combination of values based on the specified columns.
 - | sample-distinct 10 of Col1
 - Combines sample and distinct for one column.

Condition	Description
== and =~	equals (case-sensitive and case-insensitive)
!= or <>, and !~	does not equal (case-sensitive and case-insensitive)
< and <=	less than, and less than or equal to
> and >=	greater than, and greater than or equal to

Query and analyze data

26. Select, filter, and aggregate data by using the Visual Query Editor

- To use the visual editor, click on "New visual query".
 - You can drag objects (for example) tables into the SQL query window.
 - You can use the transformations in this cut-down Power Query window, either in the menu or clicking the + in a data source.
 - Manage columns
 - Choose columns
 - Go to column
 - Remove column
 - Remove other columns

DP-600: Fabric Analytics Engineer Associate
Query and analyze data

- Reduce rows
 - Keep top/bottom/range of rows
 - Keep duplicates
 - Keep errors
 - Filter rows
- Sort ascending/descending
- Transform
 - Group by
 - Replace values
- Combine
 - Merge queries/Merge queries as new
 - Append queries/Append queries as new
- You can also add the following additional transformations by clicking on the + near the data source:
 - Transform any column
 - Change type
 - Rename
 - Move columns after/to end
 - Add column/Transform text column – Format
 - lowercase/UPPERCASE
 - Trim
 - Add prefix/suffix (not Add column)
 - Add column/Transform text column – Extract
 - Length
 - First/Last characters
 - Range
 - Add column/Transform number column – Standard
 - Add/Multiply/Subtract/Divide
 - Modulo
 - Percentage
 - Percent of
 - Add column/Transform number column – Scientific
 - Absolute value
 - Square/Cube/Power/Square root
 - Base-10 logarithm

DP-600: Fabric Analytics Engineer Associate
Query and analyze data

- Add column/Transform number column – Trigonometry
 - Sine/Cosine/Tangent
 - Arcsine/Arc cosine/Arctangent
- Add column/Transform number column – Rounding
 - Round up/Round down/Round
- Add column/Transform number column – Information
 - Sign
- Add column
 - Add column from examples for all columns/selection
 - Add conditional column
 - Duplicate column
- For a full Power Query window, click on Expand (top-right hand corner).
- In the menu, you can:
 - Save as view,
 - View SQL, or
 - Refresh.
- In the bottom to the view, you can:
 - Reset,
 - Fit to view,
 - Full screen,
 - Show mini-map,
 - change the zoom, and
 - collapse/expand all queries.
- In the Data preview, you can:
 - Download Excel file,
 - Visualize results, and
 - Change the size of the Data preview (including expanding/collapsing it).
- For the individual columns, you can:
 - Sort ascending/descending,
 - Remove empty,
 - Filter

27. Select, filter, and aggregate data by using SQL

- To create SQL queries for a lakehouse, you can use the SQL Analytics Endpoint.
 - This is read-only only – no DML commands (UPDATE, INSERT, DELETE or MERGE).
 - It only allows you to read delta Tables, and not Files.
- You can access the SQL analytics endpoint:

- as an object in your Workspace.
- If you are in the Lakehouse, then you can change the connection in the top-right hand corner, and change it from Lakehouse to SQL analysis endpoint.
- To create a new SQL query, in the lakehouse:
 - click on "New SQL query", and enter your SELECT statement,
 - You can drag objects (for example) tables into the SQL query window.
 - You can click on "Run" to run the query.
- In the Data Preview, you can also:
 - See messages/results
 - Open in Excel,
 - Explore this data (but you need to highlight a query first),
 - Expand/Collapse the Data Preview

28. Select, filter and aggregate data by using KQL

- Specifying which columns are to be used
 - | project Col1, NewCol = expression ...
 - Shows the columns you want to be included.
 - Any columns not included are dropped.
 - It is not recommended to create a new column with the same name as an existing column in the input table.
 - Expressions can include:
 - +- * / for plus, minus, times and divide.
 - An integer/long divided by an integer/long results in an integer/long, so $3 / 2 = 1$. Other divisions result in a real data type.
 - % for modulo (remainder about division).
 - | project-away Col1, Col2, ...
 - Removes existing column(s) from the output.
 - Can use wildcards – for example, Column* will remove all existing columns starting with "Column".
 - | project-rename NewCol = OldCol
 - Renames OldCol to NewCol. Retains all existing columns.
 - | project-reorder Col1, Col2...
 - Reorders the columns.
 - Can also use wildcards with an optional asc or desc – for example, Column* desc.
 - Any columns not mentioned will still be retained, and will be put after the named columns.
 - | extend Col1, NewCol = expression ...
 - Adds an additional column. Any other columns are retained.
 - If NewCol already exists, the existing column is dropped.

- For aggregating data, see topic 21.
- For filtering data, see topic 25.
- Sorting
 - | sort by Col1 [asc/desc] [nulls first/nulls last], Col2...
 - Identical to "Order by".
 - Unlike SQL, the default order is descending, not ascending.
- Functions
 - See separate document.

Design and build semantic models

29. Choose a storage mode, including Direct Lake

- Import caches the data, so is often used for smaller amounts of data.
 - It requires time to import, but is swift once imported to generate results.
- DirectQuery retrieves data as and when needed, and will always have the latest data.
 - It requires no time to import, as no importing is required. However, it may take time to retrieve data.
- Composite mode (using Dual mode) is a bridge between Import and DirectQuery modes.
- Direct Lake uses the advantage of data being stored in the OneLake:
 - to give fast performance (similar to import mode),
 - but with the freshest data (like DirectQuery mode).
- It needs a lakehouse or warehouse on a Microsoft Fabric capacity.
 - This lakehouse/warehouse then uses OneLake to store the data.
 - Only tables in the semantic models derived from tables (not views) in the Lakehouse/Warehouse can use Direct Lake mode.
 - You cannot use both Direct Lake tables and other table modes (Import, DirectQuery, Dual).
 - Calculated columns and tables are not supported.
 - It supports write operations using the XMLA endpoint in the latest versions of SSMS, Tabular Editor and DAX Studio.
- See also topic 38.

30. Implement a star schema for a semantic model

- Covered in the PL-300 exam.

31. Implement relationships, such as bridge tables and many-to-many relationships

- Bridge tables can resolve many-to-many relationships, by making them one-to-many relationships.

32a. Write calculations that use DAX variables

- Example:
 - SalesAmountCalculation =
(CALCULATE(SUM(FactInternetSales[SalesAmount]), DATESMTD(FactInternetSales[DueDate])) - SUM(FactInternetSales[SalesAmount])) /

`CALCULATE(SUM(FactInternetSales[SalesAmount]),DATESMTD(FactInternetSales[DueDate]))`

- This formula calculates the MTD SalesAmount, and calculates the percentage for the month already gone excluding the current day over the current month.
- To use a DAX variable, enter, after the name of the measure:
 - `VAR NameOfVariable =`
 - and then at the end, the answer is
 - `RETURN Calculation`
- DAX variables allow you to:
 - Improve performance.
 - If you are referring to the same expression twice, it needs to calculate (evaluate) it twice.
 - If you put the expression in a variable, it only evaluates it once.
 - `SalesAmountCalculation =`
 - `VAR MonthToDate =`
`CALCULATE(SUM(FactInternetSales[SalesAmount]),DATESMTD(FactInternetSales[DueDate]))`
 - `RETURN`
 - `(MonthToDate - SUM(FactInternetSales[SalesAmount])) / MonthToDate`
 - Improve readability.
 - Variable names can be shorter than the calculation.
 - Allow for debugging.
 - You can RETURN the variable instead of the actual answer to debug.
 - You do this by commenting out ('--') the answer, and return the variable.
 - `SalesAmountCalculation =`
 - `VAR MonthToDate =`
`CALCULATE(SUM(FactInternetSales[SalesAmount]),`
`DATESMTD(FactInternetSales[DueDate]))`
 - `VAR CurrentSales = SUM(FactInternetSales[SalesAmount])`
 - `RETURN`
 - `(MonthToDate - CurrentSales) / MonthToDate`
 - Reduce complexity.
 - Variables are calculated outside of the filter contexts.
 - This may mean that you don't need to use the EARLIER or EARLIEST DAX functions.

32d. Write calculations that use DAX windowing functions

- ROWNUMBER

- You can use this as a measure. This goes through the entire table.
- RowNumberColumn =
ROWNUMBER(ORDERBY(DimProduct[EnglishProductSubcategoryName]))
 - You can order it ASCending or DESCending, and you can also order by multiple columns.
- RowNumberColumn =
ROWNUMBER(ORDERBY(DimProduct[EnglishProductSubcategoryName],
ASC,DimProduct[EnglishProductCategoryName],0))
 - You can also use it at a particular granularity. The below formula uses only the DimProductSubcategoryName to create the row numbers.
- RowNumberColumn = ROWNUMBER(ALLSELECTED(DimProductSubcategory),
ORDERBY(DimProductSubcategory[EnglishProductSubcategoryName], ASC))
 - The blanks are shown at the beginning, because ASC is being used. (This is DEFAULT.) You can change it to the end by using:
 - ASC BLANKS LAST
 - DESC BLANKS FIRST
 - or DESC.
 - You can also use DEFAULT, FIRST or LAST in the blanks argument:
- RowNumberColumn =
ROWNUMBER(ALLSELECTED(DimProductSubcategory),ORDERBY(DimProductSubcategory[EnglishProductSubcategoryName], ASC),LAST)
 - To reset every ProductCategoryKey, you can use the partitionBy argument.
 - RowNumberColumn =
ROWNUMBER(ALLSELECTED(DimProductSubcategory),
ORDERBY(DimProductSubcategory[EnglishProductSubcategoryName],
ASC),LAST,
PARTITIONBY(DimProductSubcategory[ProductCategoryKey]))
- RANK
 - This needs an additional argument at the front – what to do for ties. Suppose there is a 3-way tie for rows 2-4. The results would then be:
 - ROWNUMBER – 1, 2, 3, 4, 5
 - SKIP – 1, 2, 2, 2, 5
 - DENSE – 1, 2, 2, 2, 3
 - Examples:
- RankNumberColumn = RANK(SKIP,
ALLSELECTED(DimProduct[EnglishProductSubcategoryName],
DimProduct[EnglishProductCategoryName]),
ORDERBY(DimProduct[EnglishProductCategoryName],ASC))
- For the examples:

DP-600: Fabric Analytics Engineer Associate
Design and build semantic models

- Create table: SummaryTable = SUMMARIZECOLUMNS(DimDate[CalendarYear], DimProduct[EnglishProductCategoryName], "SalesAmount", SUM(FactInternetSales[SalesAmount]))
- INDEX
 - Returns the item in the 1st/2nd etc. position.
 - Examples:
- IndexColumn = INDEX(1, ALL(SummaryTable[CalendarYear]))
- IndexCalculation = INDEX(2, ALL(SummaryTable[CalendarYear]), ORDERBY(SummaryTable[CalendarYear], DESC))
- OFFSET
 - Returns the number of rows before/after the current row.
 - Example - Create measure:
- PreviousYearSales = CALCULATE(SUM(SummaryTable[SalesAmount]), OFFSET(-1, , ORDERBY(SummaryTable[CalendarYear])))
- WINDOW
 - Creates a series of rows which is based on an absolute or relative position.
 - Example:
- RunningSum = SUMX(
 - WINDOW (-1, REL, 0, REL, ALLSELECTED(SummaryTable[CalendarYear], SummaryTable[EnglishProductCategoryName])), SUM([SalesAmount]))
- REL = relative, ABS = absolute

33a. Implement calculation groups

- Calculation groups allow you to create similar calculations for multiple measures.
 - Often used for time intelligence calculations – MTD, QTD, YTD, PY
- The calculation groups contains calculation items – the individual calculations on a generic measures.
 - In place of a measure, you would use SELECTEDMEASURE() instead.
 - If you need the name of the measure, you can use SELECTEDMEASURENAME().
- If you are using calculation groups, implicit measures are disabled.
 - You can no longer drag “SalesMeasure” onto a visual to automatically get SumOfSalesMeasure – it needs to be separated created.
 - You can’t use OLS/RLS on calculation group tables.
- Creating calculation groups in Power BI Desktop is currently in preview. To switch it on:
 - Go to File – Options and Settings – Options.
 - Then go to Preview features and check “Model explorer and Calculation group authoring”.
- To add a calculation group:

DP-600: Fabric Analytics Engineer Associate
Design and build semantic models

- Go to the Model view.
- In the Home tab, go to the new “Calculation group” button.
 - It will then warn you that implicit measures will be disabled.
- It will then add into the formula pane:
 - Calculation item = SELECTEDMEASURE()
- It will create a table called “calculation group” – you can rename it.
- It will also create a column called “Calculation group column” – you can rename it, and an Ordinal column, which is ineffective in the Power BI Desktop.
- To view the calculations within a calculation group, in the Data pane of the Model view, click on the new Model tab.
 - You can then expand the relevant calculation groups and see the calculation items.
- To add a new calculation group, click on the “Calculation items” in the Data pane of the Model view, then click on “+ New calculation item” in the Properties pane.
 - Or right-hand click on the calculation group and select “+ New calculation item”.
- You can reorder the calculation groups items by clicking on the calculation group, and then reordering them in the Properties pane.
 - There is also an Ordinal column in the Calculation Group table.
- You can also change Precedence of a calculation group.
 - This determine which calculation group is to calculation first, if you have multiple groups.
 - Lower precedence calculation groups are applied first.
 - However, this is very advanced.
- You can create a Matrix with:
 - Dates (Year, Quarter and Month) in the Rows,
 - A Measure in the Values, and
 - The calculation group column in the columns.
- You can also add a slicer with the calculation group column, to allow the user the choice of calculations.
- Examples of calculation items:
 - MTD = CALCULATE(SELECTEDMEASURE(), DATESMTD(DimDate[Date]))
 - QTD = CALCULATE(SELECTEDMEASURE(), DATESQTD(DimDate[Date]))
 - YTD = CALCULATE(SELECTEDMEASURE(), DATESYTD(DimDate[Date]))
 - PY = CALCULATE(SELECTEDMEASURE(), SAMEPERIODLASTYEAR(DimDate[Date]))
- Example of using a Time Calculation function in another measure:
- SalesAmountPYMTD = CALCULATE(SUM(FactInternetSales[SalesAmount]), SAMEPERIODLASTYEAR(DimDate[Date]),Time Intelligence'[Time Calculation] = "MTD")

- Other examples of calculation items:
 - PY MTD = CALCULATE(SELECTEDMEASURE(), SAMEPERIODLASTYEAR(DimDate[Date]), 'Time Intelligence'[Time Calculation] = "MTD")
 - PY QTD = CALCULATE(SELECTEDMEASURE(), SAMEPERIODLASTYEAR(DimDate[Date]), 'Time Intelligence'[Time Calculation] = "QTD")
 - PY YTD = CALCULATE(SELECTEDMEASURE(), SAMEPERIODLASTYEAR(DimDate[Date]), 'Time Intelligence'[Time Calculation] = "YTD")
 - YOY = SELECTEDMEASURE() - CALCULATE(SELECTEDMEASURE(), 'Time Intelligence'[Time Calculation] = "PY")
- This last example uses “Sideways recursion”.
 - YOY% = DIVIDE(CALCULATE(SELECTEDMEASURE(), 'Time Intelligence'[Time Calculation] = "YOY"), CALCULATE(SELECTEDMEASURE(), 'Time Intelligence'[Time Calculation] = "PY"))
- Other functions:
 - SELECTEDMEASURENAME
 - Returns the name of the selected measure
 - ISSELECTEDMEASURE(Measure1, Measure2...)
 - Whether the selected measure is in the list in brackets/parentheses.
 - Used if a measure needs to be differently calculated for different measures.
 - SELECTEDMEASUREFORMATSTRING
 - The format of the SELECTEDMEASURE

33b. Implement dynamic strings

- Dynamic strings in Calculation items
 - The YOY% calculation is showing as 0.04, not 4%.
 - If you click on the YOY% calculation in the Model view, in the Properties pane – Formatting:
 - you can switch the “Dynamic format string” to Yes.
 - Click on Edit.
 - Enter: “0.00%”
- Dynamic strings for measures
 - You can change the formatting for explicit measures, such as Sum(Transactions), based on other fields.
 - You cannot use it for fields themselves (implicit measures).
 - This is a preview feature. To switch it on:
 - Go to File – Options and Settings – Options.

- Then go to Preview features and check “Model explorer and Calculation group authoring”.
- Setup:
 - MtoMPeople and MtoMTransactions
 - Create new measure:
SumOfTransaction = SUM(MtoMTransactions[Transaction])
 - Create a Table with Owner, BankAccount, SumOfTransactions and Currency.
- Click on the measure, and go to the Measure Tools menu.
- Change the Format to the new “Dynamic” option.
- There is a new drop-down next to the formula bar, containing Measure and Format.
- Enter a formula for the Format – for example:
 - `if(MIN(MtoMTransactions[Currency]) <> MAX(MtoMTransactions[Currency]),"Multiple", if(MIN(MtoMTransactions[Currency])="USD","$#,##0", if(MIN(MtoMTransactions[Currency])="EUR","€#,##0", if(MIN(MtoMTransactions[Currency])="GBP","£#,##0", MIN(MtoMTransactions[Currency]) & " #,##0"))))`

33c. Implement field parameters

- Field parameters allow the end user to change which measure or dimension is being used in your visuals.
- Field parameters are currently in preview. To switch it on:
 - Go to File – Options and Settings – Options.
 - Then go to Preview features and check “Field parameters”.
- To create a field parameter, go to Modeling – New parameter – Fields.
- Select a name for the Parameter, and add and reorder fields for that parameter.
 - You can also check “Add slicer to this page”.
 - There is no option to use no fields. This is equivalent to selecting all the fields.
 - You can include measures or dimensions.
- You can then go to your data, and go to the new table, and drag the new parameter into your visual.
- When added, you can right-hand click on the parameter and check “Show selected field”.
- To edit a parameter, you need to edit the DAX formula by clicking on the parameter in the Data pane.
 - The last argument of the NAMEOF function shows the order of the parameters.
- Note:
 - You cannot use AI visuals or Q&A.

- You cannot use this using DirectQuery unless you are using a composite model, with a local model for field parameters.
- You cannot use implicit measures – they must be explicit.
- You cannot use field parameters as the link for a drill-through or tooltip page.

34. Identify use cases for and configure large semantic model storage format

- If you have a premium capacity, you can use Large semantic models.
 - This is only for semantic models in the Power BI Service.
 - It does not affect Power BI Desktop, which is limited to 10 Gb.
- You will need to have:
 - a Premium P SKU,
 - Embedded A SKUs, or
 - Premium Per User (PPU).
 - It is not available for US Government Department of Defence customers.
- It allows for:
 - Semantic models to grow beyond 10 Gb,
 - Improves performance for XMLA write operations.
 - The difference can be significant for semantic models over 1 Gb (after compression)
 - You can check the estimated size in SSMS using the XMLS endpoint by right-hand clicking on the database and going to Properties. You will see the “Estimated size”.
 - Sets the default segment size to 8 million rows.
 - This creates a good balance for large tables between memory requirements and query performance.
 - On-demand load. If your semantic model has been “evicted” so that other models can use the memory, it:
 - retrieves relevant data pages on-demand, and
 - allows the evicted semantic model to be quickly made available for queries.
- To enable it when creating a workspace:
 - Go to the Advanced section, and click on “Large dataset storage format”.
- To enable it for all future semantic models for an existing workspace:
 - Go to the workspace.
 - Go to ... - Workspace Settings – License info – Edit.
 - In the Default storage format, change “Small dataset storage format” to “Large dataset storage format”.
- To enable it for a particular semantic model:
 - In the workspace, click on the ... next to the Semantic Model and go to Settings.

- In "Large semantic model storage format", switch to "On".
- Please note:
 - You can download a large format semantic model to Power BI Desktop, as long as it is less than 50 Gigabytes.
 - It is not available for Pro workspaces.
 - If you have a Premium workspace with a large format semantic model, and downgrade it to Pro, it will not load.
 - Be careful when refreshing large semantic models if the model size is near half of the capacity size, as it may exceed the capacity memory during refreshes.
 - You may need to use fine grained data refreshes using the XMLA endpoint instead.
 - Large semantic models must stay in the region it was first created in.
 - You can change its workspace, but the new workspace must have the same region as the old workspace.

35. Design and build composite models that include aggregations

- A composite model is a model which has:
 - multiple (two or more) data connections from different source groups, and
 - least one DirectQuery data connection (as opposed to import).
 - This can be useful if you have a huge fact table which should not be imported, but you have smaller dimension tables which can be imported.
- Previously, if you used DirectQuery, you could only have the one data connection.
- If you have both DirectQuery and Import tables, the status bar displays "Mixed" storage mode.
- Using composite models, you can more easily establish many-to-many relationships between tables.
- To use composite models on Power BI Service, you would need to have the following in Settings – Admin portal – Tenant settings:
 - Allow XMLA Endpoints and Analyze in Excel with on-premises semantic models
 - Otherwise you can't use a DirectQuery connection.
 - Users can work with Power BI semantic models in Excel using a live connection
 - Otherwise you can't make live connections.
 - Allow DirectQuery connection to Power BI semantic models
 - For Premium capacities or Premium Per User, you also need "XMLA endpoint" setting enabled, and set to "Read Only" or "Read/Write".
- Note about composite models:
 - relationships between different sources are called "limited" relationships (as opposed to "regular" relationships). For "limited" relationships:
 - You can only use INNER JOIN, not LEFT or RIGHT joins,

- You cannot use the RELATED DAX function to retrieve the “one” side of the relationship.
- It is marked on the model relationship with ().
- queries to one data source may contain potentially sensitive information given from the other data source.
- queries to one data source may include a lot of literal values, which may slow execution.
- if the query is too large, it may need to be split into multiple queries, which again may slow execution.
- Because composite models include DirectQuery sources, which may be very large, you can add aggregations.
 - These cache data for the aggregations, improving performance.
- You need to set up a separate aggregation table, as it will be hidden.
 - You can use your DirectQuery table, and then create a GroupBy version of that in Power Query.
- To set up the aggregations:
 - right-hand click on the aggregated table (not the field or the fact table) in the Data pane, and select “manage aggregations”.
 - You can also click on the ... next to the aggregated table.
 - Select what each of the aggregated fields maps to.
 - The GroupBy fields are optional if they are part of a relationship. However, they can be set if you would like.
 - If you want to create an Average, then set up two separate columns for Sum and Count.
- Note:
 - The Detail Table must use DirectQuery storage mode.
 - For best performance, the aggregation table should use the Import storage mode.
 - This means that tables connecting to the two tables on the one-side probably would use the Dual storage mode.
 - The Aggregation and Detail Columns must have the same datatype, unless you are using Count or “Count table rows”.
 - You cannot use multiple aggregations which use the same Summarization function, Detail Table and Detail Column.
 - It is not used if you only have read-only access. That’s because you might be using RLS. Instead, the detail table is used instead.
 - It needs “regular” relationships, not “limited” relationships.

Optimize enterprise-scale semantic models

37. Improve DAX performance

- To install DAX Studio, go to <https://daxstudio.org/>

- You can launch it from Power BI Desktop by going to External tools – DAX Studio.
 - You can also connect to it using a Power BI Premium XMLA Endpoint by going to File – New, and entering the connection next to “Tabular Server”.
- You can enter DAX queries:
 - You can enter “EVALUATE TableName” to query all rows in the TableName.
 - To reduce to a single column, you can use: `EVALUATE VALUES(TableName[ColumnName])`
- You can click on “Cache: Clear on Run” to remove the previously-loaded cache before running the query.
- You can copy a DAX query from Performance Analyzer in Power BI Desktop to investigate it.
- You can enable the following traces.
 - “All Queries” trace captures all query events, showing the number of milliseconds the query takes.
 - “Query Plan” displays the plan in its raw form.
 - Use only if necessary!
 - “Server Timings” display the timings:
 - Total – this is the total time, taken from the Query End profiler event. It is made up of Storage Engine (SE) time and Formula Engine (FE) time.
 - Storage Engine retrieves data. It is multi-threaded, and so should be used over FE.
 - Formula Engine computes data. It is single-threaded. This time should be minimized.
 - SE CPU – the approximate of CPU time spent on Storage Engine queries.
 - FE – the time spent in the Formula Engine.
 - SE – the amount of time spent in the Storage Engine.
 - SE Queries – a count of the number of Storage Engine queries that were performed.
 - SE Cache – the number of times the SE cache was used.
 - It also shows:
 - The estimated number of rows to be used in the query. This is useful to understand its cardinality.
 - KB – the size of the SE query. Also known as “data cache”.
- If something is running slowly, try a different DAX statement and run it having cleared the cache, and see the difference the results.
- The following might generate multiple SE queries:
 - DISTINCTCOUNT,
 - Complex filters.
- IF statements are hard for the engine to optimize.

- This includes SWITCH Statements, which are essentially nested IF statements.
- To get statistics, you can use the VertiPaq Analyzer, which is part of DAX Studio. To access it, go to Advanced – View Metrics.
 - The cardinality column contains the number of distinct values (after removing all duplicates).
 - The “Total Size” is the combined size of Data, Dictionary and Hierarchies Size.
 - The “% Table” column shows the percentage of the Total Size over the entire table.
 - The “Data Type” shows DateTime, Int64, String, Double etc.
- Best practices:
 - Reduce number of columns where possible
 - Reduce column cardinality (number of distinct values).
 - Especially do this on the 1-side of a relationship,
 - Don’t use date/time unless you need to – split it into date and time to improve cardinality.
 - If you have floating point values, apply a specific precision unless you need it. For example, don’t use 12.345 – use 12.3 if that is good enough.

38. Configure Direct Lake, including default fallback and refresh behavior

- There are limitations for Direct Lake, based on your Fabric capacity license.

SKU	Direct Lake rows per table	Maximum Direct Lake model size	Max memory
F2, F4, F8	300 million	10 Gb	3 Gb
F16	300 million	20 Gb	5 Gb
F32	300 million	40 Gb	10 Gb
F64	1.5 billion	Unlimited	25 Gb
F128	3 billion	Unlimited	50 Gb
F256	6 billion	Unlimited	100 Gb
F512	12 billion	Unlimited	200 Gb
F1024 and above	24 billion	Unlimited	400 Gb

- If the Max model size or Rows per table are exceeded, then any queries will fallback to DirectQuery mode.
- If the Max memory is exceeded, then performance will be impacted (but Direct Lake can still be used).
- To change these limits, you would need to change your Fabric capacity license.
- You cannot use Direct Lake with the SQL Analytics Endpoint view – it will automatically fallback to DirectQuery.
- You can change the default by:

- editing the Semantic Model,
 - clicking on Model in the Data pane,
 - clicking on "Semantic model", and
 - changing the
- Columns can be removed from the Direct Lake tables when:
 - The model/table has been refreshed,
 - No query has used the column for a while, or
- When the memory in the Fabric capacity should be used elsewhere.
- Data is copied into the Direct Lake tables from the underlying Parquet files, based on:
 - Queries – any new columns in new queries will be added into the Direct Lake
 - Note – this means that the first queries after a refresh could be delayed as new columns are loaded.
 - Any new columns would only be added based on the last refresh date/time.
 - When the data changes, based on the refresh settings.
- To check the Refresh settings:
 - In the Workspace, click on the ... next to the Semantic Model and go to Settings. Then expand the Refresh node.
 - Or open the Semantic Model, then go to File – Settings. Then expand the Refresh node.
- You can:
 - enable "Keep your Direct Lake data up to date".
 - Any changes to the OneLake data will automatically be reflected in the Direct Lake tables. This generally will take seconds.
 - Enabling this is recommended for small to medium-sized semantic models, or if you need up-to-date data when data is regularly updated.
 - disable "Keep your Direct Lake data up to date".
 - This might be useful if a lot of data is going to be changed – for example, before data preparation or ETL (Extract-Transform-Load) processes.
 - You can then refresh the data:
 - using a Power BI REST API (programming), or
 - in the semantic model, going to Refresh – Refresh now.
- Configure a refresh schedule.
 - You can have a refresh frequency of "Daily" (7 days a week) or "Weekly" (1 to 7 days a week – you can choose which days).
 - Note – there is no monthly refresh schedule.
 - You can select one or multiple time of days.
 - You can select on the hour, or on the half hour.

DP-600: Fabric Analytics Engineer Associate
Optimize enterprise-scale semantic models

- You can configure up to 8 time slots, unless it is on Power BI Premium, when you can configure up to 48 time slots.
- You can send refresh failure notifications to the Semantic model owner, and/or other contacts.

External Programs

Implement performance improvements in queries and report visuals

ALM Toolkit

- ALM Toolkit is used for Application Lifecycle Management (ALM).
 - You can download it from <http://alm-toolkit.com/>
- You can compare a source and target model and find changes.
 - You can launch it from Power BI Desktop and go to External tools – ALM Toolkit.
- To view the differences, go to Home – Select Actions – Hide Skip Objects.
 - These are objects which don't need any updating.
- You can also select multiple actions, and use the context menu to:
 - Skip any updates to the selected objects,
 - Create any objects which are missing in the Target,
 - Delete any objects which are missing in the Source, and
 - Update any objects which have different definitions.
- You can also do that for all objects in the Home – Select Actions dropdown.
- Once you have selected all of your actions, you should then click on Home - Validate selection, and check any warnings.
- You can select comparison options in the Home - Options dialog.
 - One of the options is “Retain only refresh-policy based partitions”. This only keeps partitions generated by incremental-refresh policies.
- You can choose from the following process modes:
 - Process Recalc (for a semantic model only):
 - Updates/recalculates hierarchies, relationships, and calculated columns.
 - Process Default
 - Performs the processing which is necessary to get it to a fully processed data.
 - For example:
 - data for empty tables/partitions are loaded,
 - Hierarchies, calculated columns, and relationships are recalculated/build/re-built.
 - Do Not Process
 - Process Full
 - Drops all existing data, and then processes the object.

DP-600: Fabric Analytics Engineer Associate
Optimize enterprise-scale semantic models

- This is required when a structural change has been made.
- It requires the most resources.
- You can also check “Process only affected tables”, to restrict the amount of processing required.
- You can create an Excel spreadsheet containing the comparison by clicking on Home – Report Differences.
- You can then click on Home – Update, which will do the actions.

Metadata Translator

- Metadata Translator
 - allows you to translate captions, descriptions and display folder names of tables, columns, measures and hierarchies.
 - allows you to translate in over 300 cultures in all languages that you can use in Power BI.
 - You can either translate it manually or you can use Azure Cognitive Services.
 - You can also export and import translations using .csv files.
- You can also translate using Tabular Editor.
 - 49. Improve DAX performance by using DAX Studio
- To install DAX Studio, go to <https://daxstudio.org/>
- You can launch it from Power BI Desktop by going to External tools – DAX Studio.
 - You can also connect to it using a Power BI Premium XMLA Endpoint by going to File – New, and entering the connection next to “Tabular Server”.
- You can enter DAX queries:
 - You can enter “EVALUATE TableName” to query all rows in the TableName.
 - To reduce to a single column, you can use: EVALUATE VALUES(TableName[ColumnName])
- You can click on “Cache: Clear on Run” to remove the previously-loaded cache before running the query.
- You can copy a DAX query from Performance Analyzer in Power BI Desktop to investigate it.
- You can enable the following traces.
 - “All Queries” trace captures all query events, showing the number of milliseconds the query takes.
 - “Query Plan” displays the plan in its raw form.
 - Use only if necessary!
 - “Server Timings” display the timings:
 - Total – this is the total time, taken from the Query End profiler event. It is made up of Storage Engine (SE) time and Formula Engine (FE) time.
 - Storage Engine retrieves data. It is multi-threaded, and so should be used over FE.

- Formula Engine computes data. It is single-threaded. This time should be minimized.
- SE CPU – the approximate of CPU time spent on Storage Engine queries.
- FE – the time spent in the Formula Engine.
- SE – the amount of time spent in the Storage Engine.
- SE Queries – a count of the number of Storage Engine queries that were performed.
- SE Cache – the number of times the SE cache was used.

Improve DAX performance by using DAX Studio

- It also shows:
 - The estimated number of rows to be used in the query. This is useful to understand its cardinality.
 - KB – the size of the SE query. Also known as “data cache”.
- If something is running slowly, try a different DAX statement and run it having cleared the cache, and see the difference the results.
- The following might generate multiple SE queries:
 - DISTINCTCOUNT,
 - Complex filters.
- IF statements are hard for the engine to optimize.
 - This includes SWITCH Statements, which are essentially nested IF statements.
- To get statistics, you can use the VertiPaq Analyzer, which is part of DAX Studio. To access it, go to Advanced – View Metrics.
 - The cardinality column contains the number of distinct values (after removing all duplicates).
 - The “Total Size” is the combined size of Data, Dictionary and Hierarchies Size.
 - The “% Table” column shows the percentage of the Total Size over the entire table.
 - The “Data Type” shows DateTime, Int64, String, Double etc.
- Best practices:
 - Reduce number of columns where possible
 - Reduce column cardinality (number of distinct values).
 - Especially do this on the 1-side of a relationship,
 - Don’t use date/time unless you need to – split it into date and time to improve cardinality.
 - If you have floating point values, apply a specific precision unless you need it. For example, don’t use 12.345 – use 12.3 if that is good enough.
- Traditional database storage
- VertiPaq database storage
- VertiPaq database storage

- VertiPaq database storage

Optimize a semantic model by using Tabular Editor 2

- Installing Tabular Editor 2
 - TE2 is the free version. TE3 has more features, but requires a subscription.
 - See <https://docs.tabulareditor.com/?tabs=TE3> for more details.
 - You can create calculated columns, measures, hierarchies, perspectives, translations, display folders etc.
 - Both features use the Tabular Object Model (TOM) to show objects and properties, and loads/saves metadata to/from Model.bim files and existing databases.
 - To download it, go to <https://github.com/TabularEditor/TabularEditor/releases/latest>
- You can open it from Power BI Desktop by going to External Tools – Tabular Editor.
- The left-hand side shows all of the tables. Within the tables, there are columns, measures and hierarchies.
- Right-hand clicking the tree will show a context menu, including adding new measures, hiding/duplicating/deleting objects.
 - You can rename objects by clicking on F2.
 - You can also select multiple objects and rename them all using the context menu.
- In the top-right hand corner, you can use the DAX Editor to edit measures or calculated columns.
 - You can click the “Dax Formatter” button to format the code.
- In the bottom-right hand corner there is a property grid.
 - You can get/set properties such as Format String and Description.
 - You can do this for multiple objects – for example, change the format string for multiple dates.
- The Best Practice Analyzer (BPA) within Tabular Editor 2 gives you recommendations for improving your model.
- To use it:
 - Go to Tools – Manage BPA Rules.
 - Next to “Rule collections”, click Add.
 - Click “Include Rule File from URL”.
 - Enter the following URL:
- <https://raw.githubusercontent.com/microsoft/Analysis-Services/master/BestPracticeRules/BPARules.json>
 - Note: The “Severity” is only for information in Tabular Editor.
 - For the other rule collections, you can create a new/clone (duplicate)/edit/delete rule.
 - You cannot do this for the BPA, because it is read-only.
 - The other rule collections are:

DP-600: Fabric Analytics Engineer Associate
Optimize enterprise-scale semantic models

- Rules within the current model,
 - Rules for the local user, and
 - Rules on the local machine.
 - Combined, they become the “(effective rules)”.
- The BPA gives suggestions for:
 - DAX Expressions,
 - Error Prevention,
 - Formatting,
 - Maintenance,
 - Naming Conventions, and
 - Performance.
- To check your model, either:
 - click the “X BP issues” at the bottom, or
 - go to Tools – Best Practice Analyse, or
 - Press F10.
- You can open an “issue” and double click on an object to go to it (or right-hand click and “Go to object...”).
- You can also:
 - Ignore item,
 - Generate fix script – you can then try it in the “C# script” (previously called the “Advanced Scripting”).
 - Apply fix (don’t forget to “Save” the changes to apply them), or
 - Copy to Clipboard.
- There is also an Undo button, just in case!