## **Topic 1: Configuring dbt Cloud data warehouse connections**

1. What is the primary purpose of creating a data warehouse connection in dbt Cloud?  
   a. To run dbt models locally.  
   b. To allow dbt Cloud to execute queries against your data warehouse.  
   c. To connect to a git provider.  
   d. To manage user licenses.  
   Réponse : b
2. When configuring a connection to a data warehouse, what is the main advantage of using OAuth?  
   a. It's the only method supported by all warehouses.  
   b. It allows users to authenticate using their warehouse credentials without storing them directly in dbt Cloud.  
   c. It bypasses the need for IP whitelisting.  
   d. It provides faster query execution.  
   Réponse : b
3. What is the function of the "Test Connection" button in dbt Cloud?  
   a. It runs all your dbt models to ensure they are valid.  
   b. It validates that the provided credentials and settings are correct and that dbt Cloud can reach the warehouse.  
   c. It creates a new schema in your warehouse.  
   d. It checks your dbt code for syntax errors.  
   Réponse : b
4. For which of the following scenarios is IP whitelisting most critical?  
   a. When using a public git repository.  
   b. When your data warehouse is hosted within a private network or VPC.  
   c. When using OAuth for authentication.  
   d. When developing in the dbt Cloud IDE.  
   Réponse : b
5. To use OAuth with Snowflake in dbt Cloud, what must you typically provide during setup?  
   a. A username and password.  
   b. A public/private key pair.  
   c. A Client ID and Client Secret from your Snowflake OAuth integration.  
   d. An API token.  
   Réponse : c
6. Where do you configure the IP addresses to be whitelisted for your dbt Cloud connection?  
   a. In your git provider's settings.  
   b. In your data warehouse's network security settings.  
   c. In the dbt Cloud project settings.  
   d. In a dbt\_project.yml file.  
   Réponse : b
7. How does dbt Cloud use the credentials you provide for a data warehouse connection?  
   a. Only for the initial connection test.  
   b. For all queries run in development and deployment environments.  
   c. Only for jobs running on a schedule.  
   d. Only for accessing the dbt Cloud IDE.  
   Réponse : b
8. What information is generally required to test a new data warehouse connection in dbt Cloud?  
   a. Git repository URL.  
   b. A valid SQL query.  
   c. Hostname, port, database name, and credentials.  
   d. The number of threads for the project.  
   Réponse : c
9. When a developer uses the dbt Cloud IDE, how are they typically authenticated to the data warehouse if OAuth is configured?  
   a. They must re-enter their warehouse password every time.  
   b. Authentication is handled seamlessly through a browser-based login flow to the data warehouse provider.  
   c. They use a shared service account token.  
   d. Authentication is not required for the IDE.  
   Réponse : b
10. What is a key benefit of separating development and deployment credentials?  
    a. It simplifies connection setup.  
    b. It allows developers to have more permissive access in their own schemas while production jobs run with more restricted, specific permissions.  
    c. It is a requirement for all git integrations.  
    d. It reduces the cost of using the data warehouse.  
    Réponse : b

## **Topic 2: Configuring dbt Cloud git connections**

1. What is the role of a git repository in a dbt Cloud project?  
   a. It stores the query results.  
   b. It serves as the version control system for all dbt project code.  
   c. It authenticates users to the data warehouse.  
   d. It stores the dbt Cloud license keys.  
   Réponse : b
2. Which of the following is NOT a supported git provider for dbt Cloud integration?  
   a. GitHub  
   b. GitLab  
   c. Azure DevOps  
   d. SVN  
   Réponse : d
3. How do you connect a dbt Cloud project to a git repository?  
   a. By emailing the repository URL to dbt support.  
   b. By installing the dbt Cloud application or setting up an SSH key within your git provider account.  
   c. By adding the git URL to your dbt\_project.yml.  
   d. By using the dbt Cloud API exclusively.  
   Réponse : b
4. What is a "deploy key" in the context of a git connection?  
   a. A key used to trigger dbt Cloud jobs.  
   b. An SSH key that grants read-only access to a single repository.  
   c. A license key for dbt Cloud.  
   d. A key for decrypting data in the warehouse.  
   Réponse : b
5. When setting up a GitHub integration, what is typically required?  
   a. Granting the dbt Cloud application access to your GitHub repositories.  
   b. Manually copying every dbt file into dbt Cloud.  
   c. Configuring a webhook in your data warehouse.  
   d. Whitelisting GitHub's IP addresses.  
   Réponse : a
6. Why is a git connection fundamental to the dbt workflow?  
   a. It enforces a specific coding style.  
   b. It enables collaboration, versioning, and CI/CD practices for analytics code.  
   c. It is the only way to schedule jobs.  
   d. It automatically generates documentation.  
   Réponse : b
7. If you have multiple dbt projects, how should you manage them in git?  
   a. All projects must be in a single, monolithic repository.  
   b. Each dbt project should typically reside in its own git repository.  
   c. Git is not recommended for managing multiple projects.  
   d. Store them in different branches of the same repository.  
   Réponse : b
8. After committing a change to your git repository, how is it reflected in the dbt Cloud IDE in your development environment?  
   a. It is reflected automatically and instantly.  
   b. You need to manually pull the changes using the git actions in the IDE.  
   c. You must restart the dbt Cloud instance.  
   d. The changes only appear after a scheduled job runs.  
   Réponse : b
9. What advantage does a native git provider integration (like the GitHub App) have over using a manual SSH key setup?  
   a. It's less secure.  
   b. It's more complex to set up.  
   c. It simplifies the setup process and often provides a better user experience for linking repositories.  
   d. It allows write access to the repository from dbt Cloud.  
   Réponse : c
10. Can you change the git repository linked to a dbt Cloud project after the initial setup?  
    a. No, you must create a new project.  
    b. Yes, this can be changed in the project's repository settings.  
    c. Yes, but it requires contacting dbt Cloud support.  
    d. No, it is locked for security reasons.  
    Réponse : b

## **Topic 3: Creating and maintaining dbt Cloud environments**

1. What is the primary purpose of having multiple environments in dbt Cloud (e.g., Development, Staging, Production)?  
   a. To increase the number of users who can access the project.  
   b. To isolate development work from production runs and allow for different configurations (e.g., data sources, credentials).  
   c. To comply with git provider requirements.  
   d. To run different versions of dbt in parallel.  
   Réponse : b
2. When is it most appropriate to use a service account for a dbt Cloud environment?  
   a. For a developer's personal development environment.  
   b. For the dbt Cloud IDE.  
   c. For a deployment environment (e.g., production) to ensure runs are not tied to a specific user's credentials.  
   d. For connecting to the git repository.  
   Réponse : c
3. What is an environment variable in dbt Cloud?  
   a. A variable used only for styling the dbt Docs site.  
   b. A key-value pair that can be used to inject configuration into your dbt project at runtime.  
   c. A variable that controls the dbt version.  
   d. A setting for the git integration.  
   Réponse : b
4. How can you ensure a production job always builds models in the analytics schema, while developers build in their own dbt\_firstname\_lastname schemas?  
   a. By using different git branches.  
   b. By setting a default schema/dataset in the configuration for the production environment.  
   c. By hardcoding the schema in every model's SQL file.  
   d. This is not possible in dbt Cloud.  
   Réponse : b
5. What does "deferral" in dbt allow you to do?  
   a. To postpone a job run to a later time.  
   b. To refer to models in another dbt project without re-building them.  
   c. To allow a CI run to build only modified models by referencing a production manifest.  
   d. To delay user access to dbt Cloud.  
   Réponse : c
6. You have a main branch for production and feature branches for development. Which branch should the Production environment be configured to use?  
   a. Any feature branch.  
   b. The main branch.  
   c. A custom branch named dbt\_production.  
   d. It does not need a specific branch.  
   Réponse : b
7. How would you rotate the credentials (e.g., a private key) for a service account used in a deployment environment via the API?  
   a. By emailing the new key to dbt support.  
   b. By using the appropriate dbt Cloud API endpoint to update the environment's connection details.  
   c. By deleting and recreating the environment.  
   d. By updating the profiles.yml file in the git repository.  
   Réponse : b
8. What is a "Custom Branch" setting in an environment useful for?  
   a. It forces all developers to use a single branch.  
   b. It allows a deployment environment to run dbt commands against a branch other than the default main branch.  
   c. It automatically creates new branches.  
   d. It deletes old branches.  
   Réponse : b
9. When creating a new deployment environment, what is a critical configuration step?  
   a. Setting the number of threads to 1.  
   b. Defining the dbt version to be used.  
   c. Choosing a name for the environment.  
   d. All of the above are important configuration steps.  
   Réponse : d
10. Access control to different environments in dbt Cloud is typically managed through:  
    a. IP whitelisting.  
    b. Git provider settings.  
    c. dbt Cloud permission sets and Role-Based Access Control (RBAC).  
    d. Environment variables.  
    Réponse : c

## **Topic 4: Creating and maintaining job definitions**

1. What is the primary goal of a Continuous Integration (CI) job in dbt Cloud?  
   a. To deploy all models to production.  
   b. To run tests and build only the modified models in a pull request to ensure changes don't break the project.  
   c. To generate documentation for the main branch.  
   d. To update user permissions.  
   Réponse : b
2. To set up a CI job that only builds modified models, which feature must be enabled?  
   a. Job chaining.  
   b. Deferral to a production environment.  
   c. Environment variable overrides.  
   d. SSO authentication.  
   Réponse : b
3. What is the correct order for commands in a typical production deployment job?  
   a. dbt run, dbt test, dbt source freshness  
   b. dbt test, dbt run  
   c. dbt source freshness, dbt run, dbt test  
   d. dbt run, dbt docs generate  
   Réponse : c (Typically you check source freshness first, then run models, then test the created models).
4. How do you schedule a dbt job to run every day at 9:00 AM?  
   a. By using a cron expression in your dbt\_project.yml.  
   b. In the job settings, by enabling the "Schedule" trigger and configuring the time and frequency.  
   c. By sending an API call every day at 9:00 AM.  
   d. This can only be done through the git provider.  
   Réponse : b
5. What does the "Generate docs on run" setting in a job do?  
   a. It sends the documentation as an email.  
   b. It automatically runs dbt docs generate as the final step of the job, making the documentation available in dbt Cloud.  
   c. It requires you to add dbt docs generate to the run commands manually.  
   d. It generates documentation in PDF format.  
   Réponse : b
6. What is "job chaining" in dbt Cloud?  
   a. Linking a dbt job to a git repository.  
   b. Configuring a job to automatically trigger upon the successful completion of another job.  
   c. A method for running multiple dbt commands in a single step.  
   d. A type of deferral.  
   Réponse : b
7. When would you use an "environment variable override" in a job's settings?  
   a. To change the dbt version for a single run.  
   b. To provide a specific value for an environment variable that is different from the one defined in the environment settings, for that job only.  
   c. To override the git branch for a job.  
   d. To change user permissions for a single run.  
   Réponse : b
8. What does the target.name setting in a job's configuration allow you to do?  
   a. It specifies which output profile from your profiles.yml to use for that job, enabling different connection settings (e.g., users, warehouses) for different jobs within the same environment.  
   b. It renames the job.  
   c. It specifies the target schema for the job.  
   d. It is a required setting for all CI jobs.  
   Réponse : a
9. What is the purpose of the "threads" setting in a job?  
   a. It determines how many jobs can run concurrently.  
   b. It sets the number of parallel models dbt can build at once within a single run.  
   c. It configures the number of users who can view the job run.  
   d. It is related to job chaining.  
   Réponse : b
10. "Self deferral" is a configuration where a job:  
    a. Defers to a completely different project.  
    b. Defers to a previous successful run of itself.  
    c. Defers to the main branch.  
    d. Cannot be used in production.  
    Réponse : b
11. Advanced CI in dbt Cloud allows for:  
    a. Automatically merging pull requests.  
    b. Running CI jobs on forked repositories.  
    c. More granular control over when CI jobs are triggered, such as on every commit to a pull request.  
    d. Sending CI results to a different system via webhooks.  
    Réponse : c & d (Advanced CI offers more configuration, including for forks and webhook triggers for more detailed reporting). Choosing c as the primary user-facing feature.
12. You want to create a job that runs the entire DAG every night. What type of job would you create?  
    a. A Continuous Integration (CI) Job.  
    b. A Scheduled Deployment Job.  
    c. An API Triggered Job.  
    d. A Source Freshness Job.  
    Réponse : b
13. Which command step should be included in a job to populate the project's documentation site?  
    a. dbt run  
    b. dbt test  
    c. dbt docs generate  
    d. dbt seed  
    Réponse : c
14. When should you choose job deferral over self deferral?  
    a. When you want a CI job to compare against a stable, trusted production environment.  
    b. When you don't have a production environment.  
    c. When you want faster CI runs regardless of accuracy.  
    d. Always; self deferral is an anti-pattern.  
    Réponse : a
15. What is a key benefit of using the "dbt Version" override in a job setting?  
    a. It forces all users to upgrade their local dbt version.  
    b. It allows you to test a new dbt version on a specific job before migrating the entire environment.  
    c. It's the only way to set the dbt version.  
    d. It improves job performance.  
    Réponse : b

## **Topic 5: Configuring dbt Cloud security and licenses**

1. What is a Service Token (or API key) used for in dbt Cloud?  
   a. To log in to the dbt Cloud user interface.  
   b. To authenticate programmatic requests to the dbt Cloud API (e.g., triggering jobs from an external orchestrator).  
   c. To connect to the data warehouse.  
   d. To connect to the git provider.  
   Réponse : b
2. What does a "Permission Set" define in dbt Cloud?  
   a. The dbt version for a project.  
   b. A collection of specific permissions (e.g., Job Admin, Account Viewer) that can be assigned to a user or group.  
   c. The IP whitelist for the account.  
   d. The number of licenses available.  
   Réponse : b
3. Role-Based Access Control (RBAC) in dbt Cloud allows you to:  
   a. Assign users to groups and then assign permission sets to those groups for more scalable and manageable permissions.  
   b. Control access based on a user's role in the git repository.  
   c. Restrict access to specific dbt models.  
   d. Change the role of the service account in the warehouse.  
   Réponse : a
4. How do you add a new developer to your dbt Cloud project?  
   a. They can sign up themselves with a link.  
   b. An account administrator invites them by email, and then assigns them a license and appropriate permissions.  
   c. By adding their username to the dbt\_project.yml file.  
   d. They must be added to the git repository first.  
   Réponse : b
5. What is the purpose of setting up a Single Sign-On (SSO) application for a dbt Cloud Enterprise plan?  
   a. To allow users to log in to dbt Cloud using their corporate credentials (e.g., Okta, Azure AD).  
   b. To connect to the data warehouse without a password.  
   c. To automatically trigger CI jobs.  
   d. To bypass the need for a git provider.  
   Réponse : a
6. A "read-only" license in dbt Cloud typically allows a user to do what?  
   a. Trigger jobs and commit code.  
   b. View job statuses, documentation, and project definitions, but not make changes or run jobs.  
   c. Access the dbt Cloud IDE in write mode.  
   d. Manage users and licenses.  
   Réponse : b
7. What are "License Mappings" used for in an Enterprise plan with SSO?  
   a. To map warehouse roles to dbt Cloud roles.  
   b. To automatically assign dbt Cloud licenses (Developer or Read-only) to users based on their SSO group memberships.  
   c. To map git groups to dbt Cloud groups.  
   d. To purchase new licenses via the API.  
   Réponse : b
8. How do you remove a user's access to dbt Cloud entirely?  
   a. Remove them from the git repository.  
   b. An administrator must delete their user account from the dbt Cloud settings.  
   c. The user must delete their own account.  
   d. Their access expires automatically after 90 days.  
   Réponse : b
9. To grant a service token the ability to only trigger a specific job (e.g., Job ID 123), what level of permissions should it have?  
   a. Account Admin.  
   b. Job Admin.  
   c. It should be assigned to a custom permission set that only grants access to run that specific job.  
   d. A read-only license.  
   Réponse : c (This level of granularity is key for security).
10. Which of the following is a security best practice for managing Service Tokens?  
    a. Share the same token among all external services.  
    b. Create distinct tokens with the minimum required permissions for each specific use case.  
    c. Hardcode tokens directly in your scripts and commit them to git.  
    d. Use a token with Account Admin permissions for all API interactions.  
    Réponse : b

## **Topic 6: Setting up monitoring and alerting for jobs**

1. How can you be notified by email if a scheduled production job fails?  
   a. By default, all users are emailed for all job failures.  
   b. You must configure email notifications in the job's "Notifications" settings, specifying recipients for failure events.  
   c. This requires a custom API integration.  
   d. By monitoring the git repository for failed checks.  
   Réponse : b
2. What is a Webhook in the context of dbt Cloud job notifications?  
   a. A link to the dbt documentation site.  
   b. A mechanism to send an automated HTTP POST request to a specified URL when a dbt Cloud event (like job success or failure) occurs.  
   c. A way to embed dbt job results in a website.  
   d. A type of email notification.  
   Réponse : b
3. For which of the following use cases would a webhook be most appropriate?  
   a. Sending a simple success/failure notification to a single person's email.  
   b. Sending a notification to a Slack channel or triggering an action in a tool like PagerDuty when a job fails.  
   c. Generating documentation.  
   d. Running dbt models.  
   Réponse : b
4. Can you configure notifications to be sent only when a job succeeds?  
   a. No, notifications are only for failures.  
   b. No, notifications are sent for both success and failure together.  
   c. Yes, you can configure notifications for success, failure, or both.  
   d. Yes, but only for CI jobs.  
   Réponse : c
5. Where in the dbt Cloud interface do you configure a webhook URL for job notifications?  
   a. In the Account Settings, under a global "Webhooks" section.  
   b. In the dbt\_project.yml file.  
   c. In the notification settings of the specific job you want to monitor.  
   d. In your git provider's settings.  
   Réponse : a (You set up the endpoint in Account Settings, and then subscribe jobs to it).
6. The payload sent by a dbt Cloud webhook typically contains what information?  
   a. The full dbt project code.  
   b. A JSON object with details about the job and the run, such as the job ID, status, and links to the run results.  
   c. The dbt Cloud license key.  
   d. The query results from the run.  
   Réponse : b
7. You want to alert your on-call data engineering team via PagerDuty immediately when the main production pipeline fails. What is the best way to achieve this?  
   a. Send an email notification and have the team monitor that inbox.  
   b. Set up a webhook to send a notification to a PagerDuty endpoint URL configured for this purpose.  
   c. Manually check the dbt Cloud UI every morning.  
   d. Write a script that polls the dbt Cloud API for the job status every minute.  
   Réponse : b
8. Can you configure different email recipients for different jobs?  
   a. No, all notifications go to the account administrator.  
   b. Yes, email notifications are configured on a per-job basis, allowing for different recipients for each job.  
   c. No, all notifications go to the person who created the job.  
   d. Yes, but only if you use webhooks.  
   Réponse : b
9. To receive notifications for a job, a user must:  
   a. Be an Account Admin.  
   b. Be the person who last committed to the git branch.  
   c. Be added to the notification list in the job's settings.  
   d. Have a developer license.  
   Réponse : c
10. Besides job success and failure, what other event can trigger a notification?  
    a. When a new user is added.  
    b. When a pull request is opened.  
    c. When a job is cancelled.  
    d. When the data warehouse connection test is successful.  
    Réponse : c

## **Topic 7: Setting up a dbt mesh and leveraging cross-project references**

1. What is the core concept of dbt Mesh?  
   a. A specific type of dbt model.  
   b. A development methodology that allows different dbt projects to reference models from each other, treating them as a cohesive whole.  
   c. A tool for visualizing the dbt DAG.  
   d. A security feature for encrypting models.  
   Réponse : b
2. To reference a model from project\_A inside project\_B, what must be true?  
   a. project\_A and project\_B must be in the same git repository.  
   b. The model in project\_A must be configured as public, and project\_B must have a dependency defined on project\_A.  
   c. Both projects must use the same data warehouse connection.  
   d. The user running the job in project\_B must have admin access to project\_A.  
   Réponse : b & c (Both are generally required). Choosing b as the most direct dbt configuration.
3. How do you make a dbt model available for other projects to reference?  
   a. By adding a special tag to the model.  
   b. By defining its access level as public in its configuration (dbt\_project.yml or a .yml properties file).  
   c. By placing it in a folder named public.  
   d. All models are public by default.  
   Réponse : b
4. In a cross-project reference, what does dbt use to resolve the location of the upstream public model?  
   a. A git submodule.  
   b. The dbt Cloud metadata API, which knows about the artifacts from other projects' production runs.  
   c. A hardcoded database reference.  
   d. An environment variable.  
   Réponse : b
5. What is a major benefit of using dbt Mesh for model governance?  
   a. It forces all projects to use the same dbt version.  
   b. It establishes clear contracts between teams by formalizing which models are intended for consumption (public) and which are internal (private).  
   c. It removes the need for data testing.  
   d. It makes dbt runs faster.  
   Réponse : b
6. When setting up a new project that will consume models from an existing project, what is a key configuration step?  
   a. Adding a dependencies.yml file to the new project to declare the dependency on the existing project.  
   b. Copying the public models from the existing project into the new one.  
   c. Merging the two git repositories.  
   d. Granting access at the warehouse level.  
   Réponse : a
7. How do environment types (e.g., Development, Deployment) relate to cross-project references?  
   a. They are unrelated.  
   b. dbt can only resolve cross-project refs by using the production artifacts of the upstream project, meaning the upstream job must run in a Deployment environment.  
   c. You can only reference models from projects in the same environment type.  
   d. Development environments cannot use cross-project references.  
   Réponse : b
8. If project\_B references a model from project\_A, does a job run in project\_B also run the referenced model from project\_A?  
   a. Yes, it runs the entire upstream DAG.  
   b. No, it does not run the upstream model. It trusts that it has already been built and references the resulting table in the warehouse directly.  
   c. Only if they are in the same git repository.  
   d. It depends on the threads setting.  
   Réponse : b
9. What is the syntax for referencing a public model named dim\_customers from another project named jaffle\_shop?  
   a. {{ ref('dim\_customers') }}  
   b. {{ ref('jaffle\_shop', 'dim\_customers') }}  
   c. {{ source('jaffle\_shop', 'dim\_customers') }}  
   d. {{ var('jaffle\_shop.dim\_customers') }}  
   Réponse : b
10. What problem does dbt Mesh solve for large organizations?  
    a. It allows many teams to work on separate, autonomous dbt projects while still being able to build on each other's work in a scalable and governed way.  
    b. It reduces data warehouse costs by compressing data.  
    c. It simplifies git branching strategies.  
    d. It provides a single user interface for all data tools.  
    Réponse : a

## **Topic 8: Configuring and using dbt Explorer**

1. What is the primary function of dbt Explorer?  
   a. To execute dbt commands.  
   b. To provide a comprehensive, interactive interface for exploring and understanding your dbt projects, including lineage, documentation, and run history.  
   c. To manage user licenses and permissions.  
   d. To write and edit model SQL code.  
   Réponse : b
2. How can dbt Explorer help you troubleshoot an issue?  
   a. By automatically fixing bugs in your SQL code.  
   b. By allowing you to visually trace the lineage of a model or column upstream to its sources, helping to identify where an error or bad data might have originated.  
   c. By connecting you with dbt Labs support staff.  
   d. By reverting your project to a previous git commit.  
   Réponse : b
3. What does the "Lineage Graph" in dbt Explorer show?  
   a. The history of job runs.  
   b. The relationships and dependencies between all the resources in your project (models, sources, tests, etc.).  
   c. The cost of running each model.  
   d. The permissions structure of your project.  
   Réponse : b
4. How can dbt Explorer help with cost and performance optimization?  
   a. By automatically rewriting queries to be more efficient.  
   b. By providing insights into model run times, identifying bottlenecks in your DAG, and showing model complexity, which can guide refactoring efforts.  
   c. By recommending a cheaper data warehouse.  
   d. By deleting unused models automatically.  
   Réponse : b
5. In dbt Explorer, how would you find models from another project that you can reference (i.e., public models)?  
   a. You must search for them in the git repository.  
   b. Explorer's interface allows you to view cross-project references and discover public models from projects your current project depends on.  
   c. This is not possible in Explorer.  
   d. You have to look at the generated documentation site.  
   Réponse : b
6. What information about a model can you typically find in dbt Explorer?  
   a. The raw SQL code.  
   b. Its documentation (description, columns, tests).  
   c. Its recent run history and status.  
   d. All of the above.  
   Réponse : d
7. If you see a failing test on a model in dbt Explorer, what can you typically do?  
   a. Ignore it, as Explorer tests are not real.  
   b. Click on the test to see its definition, the failing query, and its run history to begin debugging.  
   c. Delete the model.  
   d. Rerun the entire project.  
   Réponse : b
8. How does dbt Explorer get the information it displays?  
   a. By directly querying the git repository.  
   b. It is populated by the metadata generated from your dbt job runs (i.e., manifest.json, catalog.json, run\_results.json).  
   c. By scanning your data warehouse schemas.  
   d. From user-submitted forms.  
   Réponse : b
9. What is the difference between dbt Explorer and the classic dbt Docs site?  
   a. There is no difference; they are the same thing.  
   b. dbt Docs is a static documentation site, while dbt Explorer is a more dynamic, feature-rich platform that includes lineage, model performance, and cross-project exploration capabilities.  
   c. dbt Explorer is for writing code, while dbt Docs is for viewing it.  
   d. dbt Explorer is a paid feature, while dbt Docs is free.  
   Réponse : b
10. You need to understand the full upstream impact of changing a column in a source table. How can Explorer help?  
    a. It can't; you must manually search the codebase.  
    b. You can use its column-level lineage feature to trace that specific column through every downstream model that depends on it.  
    c. It will send you an email with the impact analysis.  
    d. It will prevent you from making the change.  
    Réponse : b

## **General Review Questions**

1. What file is the heart of a dbt project, defining its configuration?  
   a. profiles.yml  
   b. dbt\_project.yml  
   c. packages.yml  
   d. run.sh  
   Réponse : b
2. In a dbt Cloud environment, where are development credentials typically stored?  
   a. In the git repository.  
   b. dbt Cloud securely manages them per-user or per-environment, separate from the project code.  
   c. In a local profiles.yml file on your machine.  
   d. They are not needed for development.  
   Réponse : b
3. A CI job in dbt Cloud is typically triggered by what event?  
   a. A schedule (e.g., every hour).  
   b. A user clicking a "run" button in the UI.  
   c. The opening of a pull request in the connected git repository.  
   d. An email notification.  
   Réponse : c
4. To ensure production data remains clean, which dbt command should be a standard part of your production job?  
   a. dbt seed  
   b. dbt test  
   c. dbt compile  
   d. dbt init  
   Réponse : b
5. The ref function is used to:  
   a. Reference another model, creating a dependency in the DAG.  
   b. Reference a source table.  
   c. Refer to an environment variable.  
   d. Refresh the dbt Cloud IDE.  
   Réponse : a
6. What is the main advantage of using SSO for your dbt Cloud account?  
   a. It makes dbt runs faster.  
   b. It centralizes user authentication and improves security by enforcing corporate identity policies.  
   c. It is required for connecting to Snowflake.  
   d. It gives you more dbt licenses.  
   Réponse : b
7. If a job needs to run after daily\_pipeline\_job\_1 completes successfully, you should configure:  
   a. A webhook.  
   b. Deferral.  
   c. Job chaining.  
   d. A custom branch.  
   Réponse : c
8. What does the dbt Cloud API allow you to do?  
   a. Programmatically manage users, jobs, and environments.  
   b. Write SQL for your models.  
   c. Connect directly to the data warehouse.  
   d. Edit the dbt Explorer interface.  
   Réponse : a
9. RBAC is a feature primarily for:  
   a. Improving model performance.  
   b. Managing user permissions at scale in an organized way.  
   c. Defining data tests.  
   d. Connecting to git.  
   Réponse : b
10. Column-level lineage in dbt Explorer is useful for:  
    a. Tracking the cost of a specific column.  
    b. Understanding data provenance and conducting impact analysis for a specific data point.  
    c. Assigning ownership of a column to a user.  
    d. Hiding a column from read-only users.  
    Réponse : b
11. Which environment should be configured for "deferral" in a CI job?  
    a. The development environment of the user who opened the pull request.  
    b. A staging environment.  
    c. The production environment.  
    d. A new, empty environment.  
    Réponse : c
12. To allow one dbt project to build upon the models of another, you would implement:  
    a. A monolithic repository.  
    b. dbt Mesh / cross-project references.  
    c. Job chaining.  
    d. A custom macro.  
    Réponse : b
13. A "service token" is essential for what type of integration?  
    a. Allowing users to log in with SSO.  
    b. Integrating dbt Cloud with external orchestrators like Airflow.  
    c. Connecting to a GitHub repository.  
    d. Letting developers use the dbt Cloud IDE.  
    Réponse : b
14. Which feature is most critical for ensuring that code changes in a pull request do not introduce errors before being merged?  
    a. A scheduled production job.  
    b. A well-configured CI job with deferral.  
    c. Email notifications.  
    d. The dbt Explorer.  
    Réponse : b
15. The primary role of a dbt Cloud Architect is to:  
    a. Write all the dbt models for the company.  
    b. Design, configure, and maintain the dbt Cloud platform to be secure, efficient, and scalable for the organization's analytics teams.  
    c. Manage the data warehouse user accounts.  
    d. Approve every pull request.  
    Réponse : b