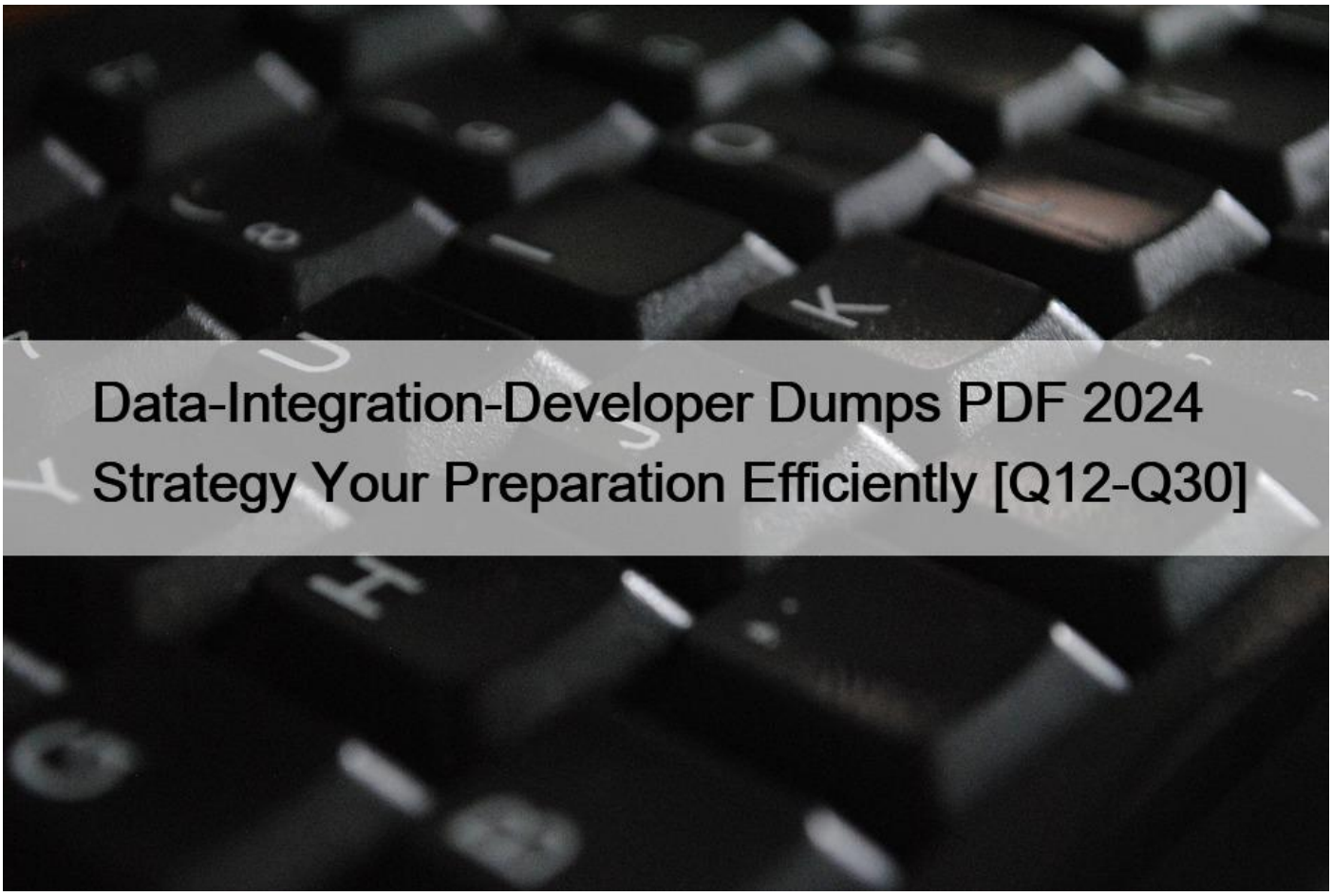
Data-Integration-Developer Dumps PDF 2024 Strategy Your Preparation Efficiently [Q12-Q30



Data-Integration-Developer Dumps PDF 2024 Strategy Your Preparation Efficiently
Latest Verified & Correct Talend Data-Integration-Developer Questions

NO.12 Which method allows you to execute the compiled Job Talend Studio?
* Click the Export button in the Edit Properties window.
* Right-click the Job in the Repository and select Build Job
* Right-click the Job in the Repository and select Export items.
* Select Save As on the main File menu

Explanation

To execute the compiled job outside Talend Studio, you need to right-click on the job in the Repository tree view and select Build Job option. This option allows you to export your job as an executable file that can be run independently from Talend Studio on any platform that supports Java. You can access this option by right-clicking on your job in the Repository tree view and selecting Build Job. This will open a dialog box where you can configure the build settings, such as destination folder, archive name, context, etc.

You do not need to click the Export button in the Edit Properties window, right-click on the job in the Repository and select Export items, or select Save As on the main File menu. These options are not used to execute the compiled job outside Talend Studio. The

Export button in the Edit Properties window is used to export your job properties as an XML file. The Export items option is used to export your items (such as jobs, metadata, routines, etc.) as an archive file that can be imported into another project or workspace. The Save As option is used to save a copy of your job with a different name or location. References: Talend Open Studio:

Open-source ETL and Free Data Integration | Talend, [Build Job – 7.3]

NO.13 In some instances, after applying changes to a component schema, you are asked if you would like to propagate the changes.

What is the significance of this prompt?
* Confirm Out you want to apply the schema, changes to the previous component in the Job.
* Confirm That you want to apply the schema changes to the selected component.
* Confirm that you want to apply the schema changes to both the previous and next components in the Job.
* Confirm that you want to apply the schema changes to the next component in the Job.

Explanation

In some instances, after applying changes to a component schema, you are asked if you would like to propagate the changes. This prompt is significant because it allows you to confirm that you want to apply the schema changes to the next component in the Job. This can save you time and effort by automatically updating the schema of the downstream component, instead of manually editing it. However, you should be careful when propagating changes, as it may overwrite existing schemas or cause errors in the Job logic. You can also choose to cancel the propagation and edit the schema manually.

NO.14 Which statements ate true about a tWarn component?

Choose 2 answers
* Used alongside a tLogCatcher in order for the log data collected to be encapsulated and passed on to the output defined.
* Used only in conjunction with the log4j feature.
* Used to trigger a warning often caught by the tLogCatcher component for an exhaustive log.
* Used only in conjunction with a iLogCatcher component.

Explanation

The tWarn component is used to trigger a warning message that can be caught by the tLogCatcher component for an exhaustive log. The tLogCatcher component is a component that collects the log data generated by the tWarn, tDie, and Java Exception components and passes it on to the output defined. The tWarn component can be used alongside a tLogCatcher component in order for the log data collected to be encapsulated and passed on to the output defined. The tWarn component can also be used independently of the tLogCatcher component, but in that case, the warning message will only be displayed in the console. The tWarn component is not used only in conjunction with the log4j feature, nor only in conjunction with a tLogCatcher component.

The log4j feature is a logging framework that allows you to customize the logging level, format, and destination of your log messages. You can enable or disable the log4j feature in Talend Studio by checking or unchecking the Use log4j option in the Preferences menu. The tLogCatcher component can catch log data from other components besides the tWarn component, such as the tDie and Java Exception components. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend,

NO.15 You have a tMap component configured with a single input row1and three outputs: CA, NY, Rejects. The input row1 consists of two columns. Name and State, containing the following data:

```
Thomas Coolidge,CA
Andrew Taylor,Ca
Calvin Adams,NY
```

All outputs map only the Name column from row1, but CA employs a filter expression, CA.equals(row1.state), while Ny employs a filter expression, NY. Equals(row1,State). All Rejects have no such configuration.

Under these conditions, how does data flow to the outputs?
* CA:Thomas Coolidge Andrew TyalorlNY;Calvin Adams;rejects;none
* CA:Thomas Coodge;Calvin Adams; rejects:Andrew Taylor
* CA:Thomas Coolidge;Ny:Calvin Adams;rejects:Thomas Coolidge, adrew taylor, Calvin Adame
* CA:Thomas Coolidge, Andrew Taylor;Ny:Calvin Adams; rejects:Thomas Coolidge, Andrew Taylor,Calvin Adams

Explanation

The tMap component allows you to configure multiple outputs for your data transformation and mapping. You can apply filter expressions on each output to filter out the rows that meet certain criteria. For example, if you want to send only the rows that have CA as the state value to one output, you can use CA.equals(row1.state) as the filter expression for that output. The filter expression evaluates to true or false for each row, and only the rows that evaluate to true are sent to the output. In this case, only Thomas Coolidge and Andrew Taylor have CA as their state value, so they are sent to the CA output. Similarly, only Calvin Adams has NY as his state value, so he is sent to the NY output. The Rejects output has no filter expression, so it receives all the rows from the input by default. However, since there is a conflict between the CA and NY outputs and the Rejects output, only the rows that are not sent to any other output are sent to the Rejects output. In this case, there are no such rows, so the Rejects output receives no data. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend,

NO.16 Which file should you edit to enable SSL for a JobServer?
* Conf/TalendJobServer.properties
* Conf/TalendServer.properties
* Conf/server.xml
* Jsl_stati64.ini

Explanation

To enable SSL for a JobServer, you need to edit the conf/TalendJobServer.properties file. A JobServer is a server application that allows you to execute jobs remotely from Talend Studio or Talend Administration Center. The conf/TalendJobServer.properties file contains various configuration parameters for your JobServer, such as port number, log level, security options, etc. To enable SSL for your JobServer, you need to set the ssl parameter to true and provide the path and password of your keystore file that contains your SSL certificate.

You do not need to edit conf/TalendServer.properties, conf/server.xml, or jsl_static64.ini files. These files are not related to JobServer configuration or SSL settings. The conf/TalendServer.properties file is used to configure Talend Administration Centersettings, such as database connection, LDAP authentication, email notification, etc. The conf/server.xml file is used to configure Tomcat server settings, such as connectors, realms, valves, etc. The jsl_static64.ini file is used to configure Java Service Launcher settings, such as service name, description, startup type, etc. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend, [Configuring Talend JobServer – 7.3], [Configuring Talend Administration Center – 7.3],

[Configuring Tomcat – 7.3], [Installing Java Service Launcher – 7.3]

NO.17 Where do you access the option to create a Joblet from an existing Job?
* Selecting the File menu.
* Selecting the desired components and right-clicking.
* Adding a Joblet component to the Designer.
* Right-clicking the I oh in the Repository

Explanation

To create a Joblet from an existing Job, you can follow these steps:

Open the Job that you want to create a Joblet from.

Select the components that you want to include in the Joblet.

Right-click on the selected components and select Create Joblet from the context menu.

In the Create Joblet dialog box, enter a name and description for the Joblet.

Click Finish to create the Joblet.

References:

Talend Open Studio User Guide: Creating a Joblet from a Job:

https://help.talend.com/r/en-US/8.0/studio-user-guide/creating-joblet-from-job

NO.18 Which statements are true about using the Implicit Context Load feature in Talend Studio?

Choose? answers
* Loads context parameters from a database table only
* Loads context parameters dynamically at the time of Job execution
* Loads context parameters from both a delimited file and a database table
* Loads context parameters from a delimited file only
Explanation

The Implicit Context Load feature in Talend Studio is a way of loading context parameters dynamically at the time of job execution from an external source, such as a delimited file or a database table. This feature can be enabled by checking the Implicit Context Load option in the Advanced Settings tab of Run Job section in Talend Studio. This feature allows the developer to avoid hard-coding context parameters in the job design and to change them easily without modifying the job. The Implicit Context Load feature can load context parameters from both a delimited file and a database table, depending on the configuration of the Implicit Context Load option. The developer can specify the file path or the database connection details, as well as the mapping between the source columns and the context variables. References: Talend Data Integration – Software to Connect, Access, and Transform Data | Talend, [Implicit Context Load – 7.3]

NO.19 Which capabilities does Traces Debugprovide?

Choose 2 answers
* Filtering which data is propagated by a flow
* Advancing one column at a time
* Breaking on a selected line of code
* Breaking when an input column fulfills a condition
* Advancing one row at a time
Explanation

Trace Debug mode is a feature that allows you to trace each row processed by your job components and see the values of each column in each row. You can access this mode by opening the Debug Run tab of the Run view and clicking on Trace Debug button.

The capabilities that Trace Debug mode provides are:

Breaking when an input column fulfills a condition. You can set breakpoints on traces based on a condition or an expression that involves input data columns. For example, you can set a breakpoint to pause your job when a customer name contains a certain string or when a product price exceeds a certain value. To set a breakpoint based on a condition, you need to right-click on a trace on your jobdesign workspace and select Show Breakpoint Setup option. This will open a dialog box where you can enter a condition or an expression for your breakpoint.

Advancing one row at a time. You can advance the execution of your job one row at a time by using the Step Over button in the toolbar of the Run view. This will allow you to see how each row is processed by your job components and how it affects the output data.

The capabilities that Trace Debug mode does not provide are:

Filtering which data is propagated by a flow. You cannot filter which data is propagated by a flow in Trace Debug mode. A flow is a link that shows the data transfer between components in your job design workspace. In Trace Debug mode, you can see the data flow on each trace and inspect the values of each column for each row processed by your job. However, you cannot change which rows are propagated by a flow based on a condition or an expression.

Advancing one column at a time. You cannot advance the execution of your job one column at a time in Trace Debug mode. You can only advance the execution of your job one row at a time by using the Step Over button in the toolbar of the Run view. This will allow you to see how each row is processed by your job components and how it affects the output data. However, you cannot see how each column is processed by your job components and how it affects the output data.

Breaking on a selected line of code. You cannot break on a selected line of code in Trace Debug mode.

You can only break on a trace based on a condition or an expression that involves input data columns.

To break on a selected line of code, you need to use Java Debug mode. Java Debug mode allows you to debug your job code in Java or Perl and see the values of each variable or expression in your code. You can access this mode by switching to the Java perspective and clicking on the Debug button in the toolbar of the Code view. References: Talend Open Studio: Open-source ETL and Free Data Integration

| Talend, [Trace Debug mode – 7.3], [Java Debug mode – 7.3]

NO.20 A colleague has exported a Data Integration Job to run outside Talend Studio.

How do you run the Job?
* Extract the contents of the archive and run the batch file and shell script.
* Extract the content of the archive and run the batch file or the shell script.
* Install the Job and start the resulting service.
* Extract the files from the archive and run the JAR file.

Explanation

To run a job that has been exported by a colleague to run outside Talend Studio, you need to extract the content of the archive and run the batch file or the shell script. The archive file contains all the files and libraries required to run the job independently from Talend Studio on any platform that supports Java. The archive file also contains two executable files: a batch file (.bat) for Windows platforms and a shell script (.sh) for Linux platforms. You need to run the appropriate file for your platform by double-clicking on it or using a command line tool. This will launch the job and display its output in a console window.

You do not need to extract the contents of the archive and run both the batch file and shell script, install the job and start the

resulting service, or extract the files from the archive and run the JAR file. These methods are not correct or available in Talend Studio and may cause errors or unexpected results. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend, [Build Job – 7.3]

NO.21 In Talend Studio, you have access to a references project named project_ref. You need to reuse a Job named testJob from the project_ref in your main Job.
* Right-click Job Designs and click import items.
* Navigate to Metadata, @ project ref, Job Designs, Standard, then select testJob.
* Right-click Job Designs, click Crete Standard Job, and enter testJob as the Job name.
* Navigate to @ Referenced project, @ project_ref, Job Designs, Standard, then select testJob.

Explanation

To reuse a job named testJob from a referenced project named project_ref in your main job, you need to navigate to @Referenced project, @project_ref, Job Designs, Standard, then select testJob. A referenced project is a project that can be accessed from another project in read-only mode. You can add a referenced project to your main project by using the Project Settings window in Talend Studio. You can then browse the items (such as jobs, metadata, routines, etc.) of the referenced project in the Repository tree view under the

@Referenced project node. You can drag and drop any item from the referenced project to your main job design workspace.

You do not need to use import items, create standard job, or navigate to Metadata options. These options are not used to reuse jobs from referenced projects. The import items option is used to import items (such as jobs, metadata, routines, etc.) from an archive file that can be exported from another project or workspace. The create standard job option is used to create a new job with a name and a description. The Metadata node in the Repository tree view is used to store and manage metadata definitions for various data sources (such as databases, files, web services, etc.). References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend, [Referenced projects – 7.3]

NO.22 You are using the tMap component to configure a mapping.

What do the tables on the left side of the Map Editor window represent?
* Explosions to apply to the input data
* Schemas of the output rows
* Schemas of the input rows
* Expression to apply to the output data

Explanation

The tables on the left side of the Map Editor window represent the schemas of the input rows. The schemas define the structure and type of the data that is coming from the input components connected to the tMap component. You can drag and drop columns from the input tables to the output tables on the right side of the Map Editor window to create mappings and transformations.

NO.23 Which statement is true about the Sync columns button on the Basic settings tab of the Component view?

Choose 2 answers
* Retrieves and synchronizes the output file schema with the input file schema for tFileOutputed limited.
* Retrieves the schema of the current component to match the Next component
* Retrieves the schema from the input component connected in the lob.
* Retrieves and synchronizes the output schema with the input schema for tJavaRow.

Explanation

The Sync columns button on the Basic settings tab of the Component view retrieves the schema from the input component connected

in the Job and synchronizes it with the current component’s schema. This button is available for most of the components that have an input and an output schema, such as tJavaRow, tMap, tFileOutputDelimited, etc. However, this button does not retrieve and synchronize the output file schema with the input file schema for tFileOutputDelimited, as this component only has one output schema.

NO.24 You are collaborating on a project with other developers using talend best practices.

When should you use a tag?
* Capturing snapshot of a codebase that requires ongoing maintenance.
* Fixing hugs or enhancements.
* Denoting major or minor releases.
* Denoting a milestone in the lifecycle of a project

Explanation

Tags are used to mark specific points in the development lifecycle of a project. This can be done for a variety of reasons, such as:

To mark the completion of a major feature or release

To mark the start of a new development phase (e.g., QA, production)

To mark a known good state of the codebase (e.g., before a risky change) Tags can be used to create branches, which can then be used to develop and test new features or bug fixes.

They can also be used to deploy the codebase to different environments (e.g., development, staging, production).

Examples of when to use a tag:

When you are ready to release a new version of your application to production.

When you have completed a major feature or milestone in the development of your application.

Before making a risky change to your codebase, such as a database migration or refactoring.

When you need to deploy your application to a different environment, such as staging or production.

Benefits of using tags:

Tags can help you to keep track of the different versions of your codebase.

Tags can help you to roll back to a previous version of your codebase if necessary.

Tags can help you to deploy your codebase to different environments.

Tags can help you to collaborate with other developers more effectively.

References:

Talend Best Practices Guide:

https://help.talend.com/r/en-US/8.0/software-dev-lifecycle-best-practices-guide/branching-merging-and-ta Git documentation on

tags: https://git-scm.com/docs/git-tag

NO.25 You need to calculate the total number of rows in an input file using a tMlelnputDelimited component.

Which code should you use in a tJava component to write a nb-line variable?
* Int nb_line-(integer)globalMap.put(”tFileInputDelimited_1_NB_LINE”)
* Int nb_line=(integer)globalMap.put(”tFileInputDelimited_1_NB_LINE”)
* Int nb.line-(integer)globalMap.put(”tFileInputDelimited_1_NB_LINE”)
* Int nb_line-(String)globalMap.put(”tFileInputDelimited_1_NB_LINE”)
Explanation

You need to use the following code in a tJava component to write a nb_line variable:

int nb_line = (Integer)globalMap.get(“tFileInputDelimited_1_NB_LINE”);

This code retrieves the value of the global variable tFileInputDelimited_1_NB_LINE, which stores the number of rows processed by the tFileInputDelimited component, and assigns it to an integer variable named nb_line.

You can then use this variable to print or manipulate the number of rows in your Job. Note that you need to use globalMap.get, not globalMap.put, to access the value of a global variable.

NO.26 Which options can you use to add a Joblet to your talend Job?

Choose 3 answers
* Use a tRunJob component and select the Joblet from the drop-down menu.
* Type the Joblet name on the Studio canvas, then select if from the Palette drop-down menu.
* Right-click the Joblet from Palette and select the Add option.
* Drag theJobket from the Repository tree view to designer canvas.
* Drag the Joblet from Palette to the design workspace.
Explanation

To add a Joblet to your Talend Job, you can use one of these options:

Type the Joblet name on the Studio canvas, then select it from the Palette drop-down menu. This will create a Joblet container on your canvas that contains all the components and links of your Joblet.

Drag the Joblet from Repository tree view to designer canvas. This will also create a Joblet container on your canvas that contains all components and links of your Joblet.

Drag Joblet from Palette to design workspace. This will open a dialog box where you can select an existing Joblet from Repository or create a new one.

You cannot use a tRunJob component and select Joblet from drop-down menu, nor right-click Joblet from Palette and select Add option. These methods are not available in Talend Studio and may cause errors or unexpected results. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend,

[Joblets – 7.3]

NO.27 A data professional would like to combine multiple data sources into a relational database to improve their business efficiency.

Which data integration initiative can help them achieve this?
* Manually collect the data from various source interfaces,thenclean and combine the data into one warehouse.
* leverage data lakes to manage complex and massive volumes of structured and unstructured data.
* Use a middleware application that acts as a mediator to normalize data and bring it into a master data pool.
* Create a data warehouse to run queries, reports, and analyses to retrieve data in a consistent format.

Explanation

A middleware application is a software that acts as a mediator between different data sources and data consumers. It can normalize data and bring it into a master data pool, which is a centralized repository of consistent and quality data. This can help a data professional to combine multiple data sources into a relational database to improve their business efficiency. A middleware application can also provide data integration services such as data transformation, validation, cleansing, and enrichment. References: What is Data Integration?, What is Middleware?

NO.28 Which statements are true about Use or register ashared DB Connection option in Talend Studio?

Choose 3 answers
* Compatible with the Use an independent process to run subJob option of tKunJob component
* Compatible tobesnared between parent and childJobs
* Compatiblewith The Use dynamicjob option of TRunJob component
* Incompatible with the Use an independent process to run subJob option of tRunJob component
* Incompatible with the Use dynamic job option of tRunJob component

Explanation

The Use or register a shared DB Connection option in Talend Studio is an option that allows you to share a single database connection between multiple components in a job or between parent and child jobs. This option can improve the performance and efficiency of your job by avoiding opening and closing multiple connections to the same database. The Use or register a shared DB Connection option is compatible with the following scenarios:

Sharing a connection between parent and child jobs. You can use this option to pass a database connection from a parent job to a child job that is called by a tRunJob component. This way, you do not need to create a new connection in the child job, but reuse the existing one from the parent job.

Sharing a connection between components in the same subjob. You can use this option to register a database connection in one component (such as tMysqlConnection) and use it in other components (such as tMysqlInput or tMysqlOutput) in the same subjob. This way, you do not need to configure each component with the same connection properties, but use a shared one.

The Use or register a shared DB Connection option is incompatible with the following scenarios:

Using an independent process to run subjob option of tRunJob component. This option allows you to run a child job in a separate JVM process from the parent job. This option is not compatible with sharing a database connection, because each JVM process has its own memory space and cannot access the objects of another process.

Using dynamic job option of tRunJob component. This option allows you to call a child job dynamically based on a context variable that contains the name of the child job. This option is not compatible with sharing a database connection, because dynamic jobs are loaded at runtime and cannot access the metadata of other jobs. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend, [Use or register a shared DB Connection – 7.3], [tRunJob properties – 7.3]

NO.29 How do you create an empty Joblet?
* Create an empty Job, then export it as a Joblet by right-clicking it in the Repository.

* Create an empty Jon and use the file menu to save it as Joblet.
* Right-click Joblet Designs the Repository and select Create Joblet.
* Right-clicking a space in the Designer and select refactor to Joblect.

Explanation

To create an empty Joblet, you need to right-click on the Joblet Designs node in the Repository tree view and select Create Joblet option. This will open a dialog box whereyou can enter the name and description of your Joblet and select an icon for it. You can then design your Joblet by adding components and links to it in the Designer workspace.

You cannot create an empty Joblet by creating an empty job, then exporting it as a Joblet by right-clicking it in the Repository, creating an empty job and using the file menu to save it as Joblet, or right-clicking a space in the Designer and selecting refactor to Joblet. These methods are not available in Talend Studio and may cause errors or unexpected results. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend, [Joblets – 7.3]

NO.30 You need to set up a server in your Talend environment that enables several members of your team to collaborate on a project.

Which application must you enable before working on a remote project?

* Git or SVN server instance must be running.
* Talent Administration Center instance must be running with the team members and project configured.
* Git or SVN client must be installed on each developed workstation.
* Talend admission Center service must be running on each development workstation.

Explanation

To set up a server in your Talend environment that enables several members of your team to collaborate on a project, you need to enable two applications before working on a remote project: Git or SVN server instance and Talend Administration Center instance. These applications are used to manage version control and collaboration for your project.

Git or SVN server instance is a server application that allows you to store and track changes of your project files using a version control system. Git and SVN are two popular version control systems that are supported by Talend Studio. You need to have a Git or SVN server instance running and accessible by your team members before working on a remote project. You also need to have a Git or SVN client installed on each development workstation to connect to the server and perform version control operations.

Talend Administration Center is a web-based application that allows you to manage users, projects, tasks, execution servers, and licenses for your Talend environment. You need to have a Talend Administration Center instance running and accessible by your team members before working on a remote project. You also need to configure the team members and project settings in Talend Administration Center to grant access and permissions for your project. References: Talend Open Studio: Open-source ETL and Free Data Integration | Talend, [Version control – 7.3], [Talend Administration Center – 7.3]

Talend Data-Integration-Developer Exam Syllabus Topics:

TopicDetailsTopic 1- Create ?if' triggers based on component variables- Use metadata, generic schemas, and context variables in

database componentsTopic 2- Configure a remote connection in Talend Studio- Send dynamic parameters to a child Job by overriding context variablesTopic 3- Customize SQL queries in database components- Use triggers to create a sequence of subJobsTopic 4- Profile the execution of a Job with and without parallelism applied- Define a standard context variable use caseTopic 5- Explain key differences between a local and remote connection- Use the tMap component and configure a simple mapping

Data-Integration-Developer PDF Dumps Are Helpful To produce Your Dreams Correct QA's:
https://www.actualtests4sure.com/Data-Integration-Developer-test-questions.html]