

erwin Data Modeler

Git Support

Release 12.5

Legal Notices

This Documentation, which includes embedded help systems and electronically distributed materials (hereinafter referred to as the "Documentation"), is for your informational purposes only and is subject to change or withdrawal by Quest Software, Inc and/or its affiliates at any time. This Documentation is proprietary information of Quest Software, Inc and/or its affiliates and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of Quest Software, Inc and/or its affiliates

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all Quest Software, Inc and/or its affiliates copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to Quest Software, Inc and/or its affiliates that all copies and partial copies of the Documentation have been returned to Quest Software, Inc and/or its affiliates.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, QUEST SOFTWARE, INC. PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL QUEST SOFTWARE, INC. BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF QUEST SOFTWARE, INC. IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is Quest Software, Inc and/or its affiliates.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2024 Quest Software, Inc and/or its affiliates All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Contact erwin

Understanding your Support

Review support maintenance programs and offerings.

Registering for Support

Access the erwin support site and register for product support.

Accessing Technical Support

For your convenience, erwin provides easy access to "One Stop" support for all editions of <u>erwin Data Modeler</u>, and includes the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- erwin Support policies and guidelines
- Other helpful resources appropriate for your product

For information about other erwin products, visit http://erwin.com/products.

Provide Feedback

If you have comments or questions, or feedback about erwin product documentation, you can send a message to <u>techpubs@erwin.com</u>.

News and Events

Visit <u>News and Events</u> to get up-to-date news, announcements, and events. View video demos and read up on customer success stories and articles by industry experts.

Contents

Git Support	6
Connecting to Git Repositories	
Troubleshooting	
Committing Forward Engineering Scripts	15
Scenario 1: Committing New or Full FE Scripts	
Scenario 2: Committing Alter Scripts	

Git Support

Git Support

This guide walks you through the configuration and usage of Git via erwin Data Modeler.



A Git repository may be hosted on GitLab, GitHub, or Bitbucket. For a successful connection to these repositories, following are the prerequisites:

GitHub Scope: Ensure that the following minimum scope is configured.

Expiration *	
90 days 🗘 The token	will expire on Wed, Mar 15 2023
Select scopes	
Scopes define the access for pe	ersonal tokens. Read more about OAuth scopes.
🗹 геро	Full control of private repositories
repo:status	Access commit status
repo_deployment	Access deployment status
public_repo	Access public repositories
repo:invite	Access repository invitations
security_events	Read and write security events
workflow	Update GitHub Action workflows
□ write:packages	Upload packages to GitHub Package Registry
read:packages	Download packages from GitHub Package Registry
□ delete:packages	Delete packages from GitHub Package Registry
admin:org	Full control of orgs and teams, read and write org projects
write:org	Read and write org and team membership, read and write org projects
read:org	Read org and team membership, read org projects
manage_runners:org	Manage org runners and runner groups

• **GitLab Scope**: Ensure that the following minimum scope is configured.

Select scopes Scopes set the permission levels granted to the token. Learn more.	
Grants complete read and write access to the scoped project API, including the Package Registry.	
Grants read access to the scoped project API, including the Package Registry.	
Grants read access (pull) to the repository.	
Grants read and write access (pull and push) to the repository.	
 read_registry Grants read access (pull) to the Container Registry images if a project is private and authorization is required. 	
Grants write access (push) to the Container Registry.	

.

Bitbucket Sco	e: Ensure	that the	following	minimum	scope is	configured.
		cina cine	1011011110		000000.0	connigar car

-	
Scopes	
Learn more about scopes.	
Repositories	Pipelines
✓ Read	Read
🖌 Write	Write
Admin	Edit variables
Delete	Runners
Pull requests	Read
Read	Write
Write	
Webhooks	
Webhooks Read and write	

- erwin Mart: Ensure that,
 - erwin DM is connected to erwin Mart Portal. For more information on connecting erwin DM to Mart Portal, refer to the <u>Connect to Mart</u> topic.

Permissions	Ē	ē
Catalog Management		~
DI Connect Management		~
Library		~
Mart		~
Model		~
NSM File Management		~
Permission Management		~
Reports		~
Session Management		~
User Management		~
DM Connect for Git		^
Enable 🔕		•

This permission is not available for other out-of-box profiles.

Personal Access Token: Ensure that you have created the required personal access token. To know how to create personal access tokens for GitLab, refer to the GitLab documentation. To know how to create personal access tokens for GitHub, refer to the GitHub documentation.

Once, these prerequisites are in place, to connect Git repositories to erwin DM, follow these steps:

1. On the ribbon, click Mart.

	- II E	• * ~	▶ ₹									erwin DM	
File	Home	View	Dia	agram	Mode	I Mart	t A	ctions	Tools	Help			
	E2	Ċ		ব	8		\rightarrow		9		6	\diamond	
Connect	Disconnect	Open	Save	Save As	Lock	Catalog Manager Mart	_	Review	Refresh	Session Manager	Password	Git Connections	

2. Click Git Connections.

The Git Connection Manager page appears.

Git Connection	Manager			×
User Credentials				
Connection Name*:				
Git Hosting Service*:	GitLab			~
User Name:				
Password:				
Personal Access Token*:				
Git Repository*:				
Git Branch*:				
ecent Connections:				
		Save	Cancel	Help

3. Enter appropriate values in the fields. Refer to the following table for field descriptions.

Field Name	Description	Additional Information
Connection Name	Specifies a user defined con- nection name	For example, ConnectGit. You can create multiple connections one for each Git repository.
Git Hosting Service	Specifies the Git hosting service to which erwin DM	GitLab : Indicates that erwin DM connects to GitLab

Field Name	Description	Additional Information
	connects	GitHub: Indicates that erwin DM connects to GitHub Bitbucket: Indicates that erwin DM connects
	Specifies the user-	to Bitbucket
User Name	name to log on to the Git hosting service	This field is not mandatory.
Password	Specifies the pass- word to log on to the Git hosting service	This field is not mandatory.
Personal Access Token	Specifies the per- sonal access token to connect to the Git hosting service	
Accoss	Specifies the per- sonal access	This option is available only when the Git Hosting Service option is set to Bitbucket.
Access Token	token to connect to the Bitbucket hosting service	Additionally, click the Access Token field label to learn about creating tokens and their types.
Git Repos- itory	Specifies the URL of a Git repos- itory where you want to push the forward engin- eering script	For example, https://- gitlab.com/d4215/GitLabIntegration or https://github.com/poly-inc/poly-main-MCL

Field Name	Description	Additional Information
Git Branch	Specifies the branch that is used to push the forward engin- eering script	For example, main.

4. Click Save.

On successful connection, the connection name appears under Recent Connections.

E Git Connection N	lanager	\times
User Credentials		
Connection Name:	ConnectGit]
Git Hosting Service:	GitLab]
User Name:]
Password:]
Personal Access Token:	glpam-A6BFrmClh7o6witzBh_l]
Git Repository:	https://gitlab.com/d4215/GitLabIntegration]
Git Branch:	main	
Recent Connections:		
ConnectGit		
		_
		_
<		>
	Save Cancel Help	

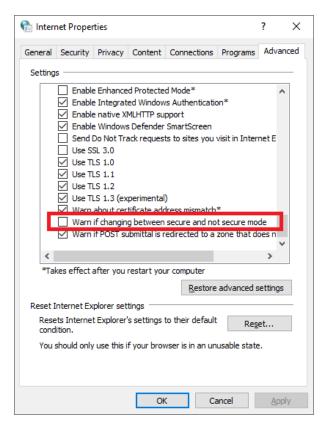
Troubleshooting

While setting up your connection, you may encounter the following error:



To resolve this error, follow these steps:

- 1. On your system, go to **Control Panel > Internet Options > Advance Tab**.
- 2. Clear the Warn if changing between secure and not secure mode check box.



- 3. Click **OK**.
- 4. Close and reopen erwin DM.

- 5. Connect erwin DM to Mart Portal.
- 6. Launch the Git Connection Manager page and configure the Git connection.

Once you are connected to a Git repository, you can <u>commit FE scripts</u>.

There are two scenarios in which you commit Forward Engineering (FE) scripts to a Git repository:

Scenario 1: Committing new or full FE scripts:

Use the Forward Engineer Schema Generation Wizard to commit a physical database schema or FE script from a Mart Model.

To avoid script files from being overwritten, ensure that you use unique file names.

Scenario 2: Committing alter scripts:

Use the Forward Engineer Alter Script Schema Generation Wizard to commit an alter script after you make changes to a Mart Model. You can commit an alter script in two ways:

- Commit and append an alter script to an existing script file
- Commit and create a new alter script file in the Git repository

For more information, refer to the Scenario 2: Committing Alter Scripts topic.

Scenario 1: Committing New or Full FE Scripts

The Forward Engineer Schema Generation Wizard generates a physical database schema or Forward Engineering (FE) script. For a Mart Model, you can push the FE script to a Git repository.

To commit new or full FE scripts to Git repositories, follow these steps:

1. On the ribbon, go to **Mart** > **Open**.

The Open page appears.

Open							×
Catalog:						≒∣ ≕	۱
Mart Nart	Name	Creator	Last Modifier	Modified On	Entity/	Attribu	Rel
	🗅 erwin			10-02-2022 12:40:00	8	95	7
	Model_2			16-12-2021 12:23:00	1	1	D
	<						>
Model Name: erwin					Sho	w Templates	only
Lock Type: Unlocked	Open Version:	(Current Version)	Version 1 by :	on 10-02-2022 12:40:	00 V	Open	
Model Description						Cancel	
% 🛼 🗞 🖂 🛷							
					^		
					~		

2. Select a model, and then click **Open**.

The Mart Model opens.

Mart://Mart/erwin : v1 🛛 🗙					Objects Count += >	Y Pro
Mart://Mart/erwin: v1 : ER_Diagram_	mo_rent_rec rental_date due_date rental_status overdue_charge rental_rate rental_record_date	cust CUST_address	credit_card credit_card_exp status_code	movie	Name: MODEL_2 Type: Logical/Physical View Mode: Physical Subject Areas: 0 Indexes: Tables: 8 Relationships: Columns: 61 Sub-Categories: Style: Pie Style (2D) Pallete: Illu	Properties Objects Count
CUST_number CUST_address email CUST_city CUST_first_name CUST_state CUST_state CUST_state CUST_state CUST_state CUST_pp_code rental_record_date rental_date ER_Disgram_117		CUST_city CUST_last_name CUST_last_name CUST_state CUST_zip_code email CUST_number	CUST_number CUST_address email CUST_city CUST_first_name CUST_state CUST_state CUST_zip_code rental_record_date due_date	movie descr star_1 star_2 movie genre rental movie		
	Onlin	ne Mart Model	100	1	Databricks 100% – –	

3. Go to Actions > Schema.

The Forward Engineer Schema Generation Wizard appears.

Forward Engineer Scl	hema Generation Wizard — 🗌 🔿
nema Generation Ov his page provides an over	erview erview of the Forward Engineer Schema Generation.
Overview	Overview
Option Selection	Welcome to the Forward Engineer Wizard. This wizard lets you forward engineer a model.
	Forward engineering is the process of generating the physical database schema from a physical model. You use the
Owner Override	Schema Generation wizard to forward engineer a model and generate the schema. The schema that you generate includes all options that are supported in your target server. You can customize the generated schema for your target
Table Filter	server using features in the Schema Generation wizard.
	The Option Selection page lets you customize the Schema Generation option set and the database template.
	The Summary page lets you review the selected Schema Generation options in a hierarchical tree structure. The page also lets you enter a comment for the current Schema Generation option set.
	The Owner Override page lets you enter the owner override name of an object group. The object group and name you indicate override the owner name assignment for that object group in the model.
	The Table Filter page lets you select a subset of the tables in a model for forward engineering.
	The Preview page lets you preview the schema DDL script generated for the current model.
	×
	< <u>Back</u> <u>Next</u> > Generate OK Cancel Help

4. On the Forward Engineer Schema Generation Wizard, click the Preview section.

The FE script appears. For example, in the following image the Preview section displays FE script of a Databricks database. For more information on generating FE scripts, refer to the <u>Forward Engineering/Schema Generation for Databases</u> topic.

, page provides a pre	view of the Forward Engineer Schema Generation.		
	🗎 🗦 萬 🖣 💩 📎		
Option Selection	1		_
Summary	2 CREATE TABLE cust		
Owner Override	3 (4 CUST_address string,		
Table Filter	5 CUST_city_string,		
Preview	6 CUST first name string, 7 CUST last name string,		
Preview	7 CUST last name string, 8 CUST state string,		
	9 CUST zip code int,		
	10 email string,		
	11 CUST number int		
	12)		
	13 USING delta		
	14 LOCATION 'dbfs:/user/hive/warehouse/erwin.db/cust'		
	15 TBLPROPERTIES ('delta.minReaderVersion'='1', 'delta.minWs	terVers	31
	16		

5. Click 🔷.

The Commit to Git screen appears.

Commit to Git	
Connected To*:	ConnectGit 🗸
Git Repository*:	https://gitlab.com/gitintigration/shubhamgitlab
Git Branch*:	shubham_gitlab
File Name*:	
Hie Name .	
Git Path:	
Commit Summary*:	
Author Name:	
Autior Name.	
Author Email ID:	
Local Path:	Browse
	Commit Cancel Help

6. Enter appropriate values in the fields. Fields marked with an asterisk (*) are mandatory. Refer to the following table for field descriptions.

Field Name	Description	Additional Information
Connected To	Specifies the con- nection that con- nects erwin DM to a Git repository	For example, ConnectGit.
Git Repos- itory	Specifies the Git repository con- figured for the connection	For example, https://- gitlab.com/d4215/GitLabIntegration is set for the ConnectGit connection. This field autopopulates based on the repos- itory configured in the Git Connection Man- ager.
Git Branch	Specifies the Git branch con-	For example, main is set for the ConnectGit connection.

Field Name	Description	Additional Information
	figured for the connection	This field autopopulates based on the repos- itory configured in the Git Connection Man- ager.
File Name	Specifies the user- defined name of the FE script file being committed to a Git repository	For example, Databricks-Sales-Data.sql To avoid script files from being overwritten, ensure that you use unique file names.
Git Path	Specifies the loc- ation in the Git repository where the FE script is committed	For example, FY2022/ The FE script is committed to the FY2022 folder inside the root folder of your Git repos- itory.
Commit Summary	Specifies the sum- mary of the push commit	For example, Sales Rectification.
Author Name	Specifies the name of the author pushes the commit	
Author Email ID	Specifies the email address of the author pushes the commit	
Local Path	Specifies the loc- ation on your local machine where the FE script is saved	C:\Users\SO\Documents\Databricks

7. Click **Commit**.

The FE script file is saved on the local path and committed to the Git repository.

For example, in the following image, FE script is committed to a GitLab repository in a file, Databricks-Sales-Data, with a commit summary, Sales Rectification using the main branch.

G GitLabIntegration Project information C Learn GitLab Repository	main	+ • History Find file Web IDE •	✓ d × Clone × c6aa3fd2 C
Files			
Commits	Name	Last commit	Last update
Branches			
Tags			
Contributors	♦ .gitkeep	Add new directory	4 days ago
Graph	🕒 Databricks	Sales Data	4 days ago
Compare	🕒 Databricks-Sales-Data	Sales Rectification	just now
D Issues 0	🕒 HumanResourceData	HR Data	3 days ago
33 Merge requests	(D. 5)	C-Clarke	2 dama and
🕼 CI/CD	🕒 PII	Confidential	3 days ago
Φ Security & Compliance	🕒 WebSecurityIssues	PII	3 days ago

You can click the file to review its content. For example, in the following image, Databricks-Sales-Data's content is visible.

G GitLabIntegration	main	 GitLabIntegration / FY2022 / Databricks-Sales-Data 	Find file Blame History Permalink
Project information			
😵 Learn GitLab 🛛 😢		ales Rectification authored 12 minutes ago	c6aa3fd2
E Repository	- 1	authored 12 minutes ago	
Files			
Commits	🖻 Data	pricks-Sales-Data [្រុំ 6.62 KB	Edit in Web IDE 💙 Replace Delete 🛱 🖄 🕹
Branches	1		
Tags	2	CREATE TABLE cust	
Contributors	4	CUST_address string,	
Graph	5	CUST_city string, CUST_first_name string,	
Compare	7	CUST_last_name string, CUST state string,	
D Issues	9	CUST_zip_code int,	
	10	email string,	
11 Merge requests	11 12	CUST_number int	
🕼 CI/CD	13	USING delta	
A	14	LOCATION 'dbfs:/user/hive/warehouse/erwin.db/cust'	
Security & Compliance	15	TBLPROPERTIES ('delta.minReaderVersion'='1','delta.minWriterVersion':	='2');
− ≪ Collapse sidebar	16 17 18	CREATE TABLE cust_credit (

You can use FE Schema Generation Wizard to commit FE script using the same connection again. The Commit to Git screen autopopulates the previously set values in File Name and Git Path.

For example, in the following image File Name is set to Databricks-Sales-Data and Git Path is set to FY2022/.

Commit to Git	
Connected To*:	ConnectGit ~
Git Repository*:	https://gitlab.com/d4215/GitLabIntegration
Git Branch*:	main
File Name*:	Databricks-Sales-Data
Git Path:	FY2022/
Commit Summary*:	
Author Name:	
Author Email ID:	
Local Path:	Browse
	Commit Cancel Help

Committing the FE script again with the same File Name and Git Path overwrites the previous file in the Git repository.

Once the FE script is committed, you can run it on your database to generate and verify the physical schema.

For example, in the following Databricks database, the FE script copied from the Git repository is run.



After running the FE script, the required database objects are created. You can access these objects from the database. For example, the following tables can be accessed in a Databricks database.

D Data Science &… ▼	Databases 📀	•	Tables	
O Creata	Q Filter Databases		Q Filter Tables	
⊕ Create	Se customer_sc		⊞ cust	-
Workspace	₿ db1_sravan		⊞ cust_credit	-
😱 Repos	efault		⊞ emp	-
() Recents	Se demo		⊞ mo_rent_rec	-
	🛢 demo_db2		⊞ movie	-
Q Search	🛢 demo_fe		⊞ movie_copy	-
👌 Data	emors		⊞ payment	-
	Semployees		⊞ store	-
์ Compute	semployees2			
ž⊟ Jobs	erwin5			
	erwinsales			
S Partner Connect	new_sravan			

Scenario 2: Committing Alter Scripts

The Forward Engineer Alter Schema Generation Wizard generates an alter script for a database after you make changes to a model. For a Mart Model, you can push the alter script to a Git repository.

To commit alter scripts to Git repositories, follow these steps:

1. On the ribbon, go to Mart > Open

The Open page appears.

Open					_		×
Catalog:						≣ ≡	۱
Mart Nart	Name	Creator	Last Modifier	Modified On	Entity/	Attribu	Rel
	🗅 erwin			10-02-2022 12:40:00	8	95	7
	Model_2			16-12-2021 12:23:00	1	1	0
	<						>
Model Name: erwin					Sho	w Templates	only
Lock Type: Unlocked V	/ Open Version:	(Current Version)	Version 1 by :	on 10-02-2022 12:40:	00 ~	Open	
Model Description						Cancel	
🏸 🛼 🙈 🔂 😽							
					^		
					~		

2. Select a model, and then click **Open**.

The Mart Model opens.

Mart://Mart/erwin : v1 🛛 🗙					Objects Count	+×
Mart://Mart/erwin : v1 : ER_Diagrar	n_117 *			C	Name: MODEL_2	+ x
	mo_rent_rec				Type: Logical/Physical View Mode: Physical Subject Areas: 0 Indexes:	٤
credit_card credit_card_exp status_code	rental_status overdue_charge rental_rate rental_record_date	cust	customer_invoice credit_card credit_card_exp	movie	Tables: 8 Relationships Columns: 61 Sub-Categori Style: Pie Style (2D) * Palete:	es: 0
Overdue_Charge_Rate CUST_address email CUST_city CUST_first_name CUST_ist_name CUST_state CUST_state CUST_state CUST_stip_code rental_record_date rental_date		CUST_address CUST_city 	status_code CUST_number CUST_address email CUST_city CUST_first_name CUST_last_name CUST_state CUST_state CUST_zip_code rental_record_date due_date	D rating star_1 p rating star_2 movie genre rental movie	8	
ER_Diagram_117		ne Mart Model			Databricks 100% -	

3. Make the required changes in the model.

For example, in the following model, a new table, cust_dec with four columns is added.

Mart://Mart/erwin : v1 🗙						Objects Count	⇔ X
Mart://Mart/erwin : v1 : ER_Diag	ram_117 *				- • ×	Name: MODEL_	2
	mo_rent_rec				^	Type: Logical/P View Mode: Physical	hysical
overdue_notice	rental_date due_date rental_status overdue_charge rental_rate	cust_dec First_Week Second_Week	customer_invoice			Subject Areas: 0 Tables: 9 Columns: 65	Indexes: Relationships: Sub-Categories:
credit_card credit_card_exp status_code Overdue_Charge_Rate CUST_number	rental_record_date	Third_Week Fourth_Week	credit_card credit_card_exp status_code CUST_number	movie		Style: Pie Style (2D)	✓ Pallete: Illu
CUST_raidness email CUST_city CUST_first_name CUST_first_name CUST_state CUST_zip_code rental_record_date		CUST_address CUST_city CUST_first_name CUST_fast_name CUST_state CUST_sit_code email CUST_aumber	CUST_address email CUST_city CUST_first_name	movie_title movie_director description star_1_name rating star_2_name movie_number genre rental_rate		65	
<er_diagram_117< td=""><td></td><td></td><td></td><td></td><td>></td><td></td><td></td></er_diagram_117<>					>		

4. Go to Actions > Alter Script.

The Forward Engineer Alter Script Schema Generation Wizard appears.

Forward Engineer Al	ter Script Schema Generation Wizard — 🗌 🔿
	eneration Overview erview of the Forward Engineer Alter Script Schema Generation.
Overview	Overview
	Welcome to the Forward Engineer Wizard. This wizard lets you forward engineer a model.
	Forward engineering is the process of generating the physical database schema from a physical model. You use the Schema Generation wizard to forward engineer a model and generate the schema. The schema that you generate includes all options that are supported in your target server. You can customize the generated schema for your target server using features in the Schema Generation wizard. The Option Selection page lets you customize the Schema Generation option set and the database template. The Summary page lets you review the selected Schema Generation options in a hierarchical tree structure. The page also lets you enter a comment for the current Schema Generation option set. The Owner Override page lets you enter the owner override name of an object group. The object group and name
	The Table Filter page lets you select a subset of the tables in a model for forward engineering. The Table Filter page lets you select a subset of the tables in a model for forward engineering. The Preview page lets you preview the schema DDL script generated for the current model.
	< Back Next > Generate OK Cancel Help

5. On the Forward Engineer Alter Schema Generation Wizard, click the Preview section.

The alter script appears. For more information on generating alter scripts, refer to the <u>Generating Alter Script for Databases</u> topic.

For example, in the following image the Preview section displays an alter script of a Databricks database.

E Forward Engineer Alte	er Script Schema Generation Wizard			×
Alter Script Schema Ger This page provides a previ	neration Preview ew of the Forward Engineer Alter Script Schema Generation.			
Overview Option Selection Summary Owner Override Table Filter Preview	CREATE TABLE cust_dec CREATE TABLE cust_dec CREATE TABLE cust_dec First_Week BYTE, Second_Week BYTE, Fourth_Week BYTE); 9			>
	< <u>B</u> ack Next > Generate OK (Cancel	Не	lp

6. Click 🔷.

The Commit to Git screen appears. The File Name and Git Path values autopopulates with the values configured in the previous commit. You can update the File Name and Git Path as per the requirement.

Commit to Git	
Connected To*:	ConnectGit 🗸
Git Repository*:	https://gitlab.com/gitintigration/shubhamgitlab
Git Branch*:	shubham_gitlab
File Name*:	
Git Path:	
Commit Summary*:	
Author Name:	
Author Email ID:	
Local Path:	Browse
Auto Append	Commit Cancel Help

7. Enter appropriate values in the fields. Fields marked with an asterisk (*) are mandatory. Refer to the following table for field descriptions.

Field Name	Description	Additional Information
Connected To	Specifies the connection that connects erwin DM to a Git repository	For example, ConnectGit.
Git Repos- itory	Specifies the Git repository configured for Connection	For example, https://- gitlab.com/d4215/GitLabIntegration is set for the ConnectGit connection. This field autopopulates based on the repository configured in the Git Connection Manager.
Git Branch	Specifies the Git branch that was set for con-	For example, main is set for the ConnectGit con- nection.

Field Name	Description	Additional Information
	nection in the Git Connection Manager	This field autopopulates based on the repository configured in the Git Connection Manager.
File Name	Specifies the user-defined name of the FE script file being committed to a Git repository	For example, Databricks-Sales-Data.sql
Git Path	Specifies the location in the Git repository where the FE script is com- mitted	For example, FY2022/ The FE script is committed to the FY2022 folder inside the root folder of your Git repository.
Commit Summary	Specifies the summary of the push commit	For example, Append December Sales.
Author Name	Specifies the name of the author pushes the commit	
Author Email ID	Specifies the email address of the author pushes the commit	
Local Path	Specifies the	C:\Users\SO\Documents\Databricks

Field Name	Description	Additional Information
	location on your local machine where the Alter script is saved	
Auto Append	Specifies whether the alter script is appended to the file set in File Name and Git Path	By default, the Auto Append check box is selected. To create a new script file, clear the Auto Append check box and set the File Name and File Path belonging to an existing file. A new file with the fol- lowing naming convention: <file name="">_YYYY-MM- DD_HH-MM-SS is created. Ensure that you use this check box consistently every time you commit an alter script.</file>

8. Click Commit.

The alter script file is saved on the local path and committed to the Git repository.

For example, in the following image, an alter script file is committed to a GitLab repository and appended to an existing file, Databricks-Sales-Data, with a commit summary, Append December Sales using the main branch.

main	22 / + •	History Find file Web IDE 🔹 Clone 🗸
Append December Sales Data authored 8 minutes ago		af7a2c4f
Name	Last commit	Last update
🚸 .gitkeep	Add new directory	1 week ago
🕒 Databricks	Sales Data	1 week ago
Databricks-Sales-Data	Append December Sales Data	8 minutes ago
🕒 HumanResourceData	HR Data	1 week ago
🕒 PII	Confidential	1 week ago

You can click the file to review its content. For example, in the following image, Databricks-Sales-Data contains the alter script.

21	status_code string,
22	CUST_number int
23)
24	USING delta
25	LOCATION 'dbfs:/user/hive/warehouse/erwin.db/cust_credit'
26	<pre>TBLPROPERTIES ('delta.minReaderVersion'='1','delta.minWriterVersion'='2');</pre>
27	
28	CREATE TABLE cust_dec
29	(
30	First_Week BYTE,
31	Second_Week BYTE,
32	Third_Week BYTE,
33	Fourth_Week BYTE
34);
35	
36	CREATE TABLE emp
37	(
38	EMP_first_name string,
39	EMP_address string,
40	EMP_phone int,
41	EMP_address_2 string,
42	email string,
43	salary int,
44	hire_date timestamp,
45	soc_sec_number int,
46	EMP_number string

Clearing the Auto Append check box and setting the File Name and File Path belonging to an existing file creates a new file with the following naming convention: <File Name>_YYYY-MM-DD_HH-MM-SS.

For example, in the following image, a file is created with a time stamp in a Git repository.

nain ~ GitLabIntegration / FY2022 / +	 ✓ History Find file V 	Web IDE
Append Sales Data authored just now		a4005f10 [^A]
Name	Last commit	Last update
♦ .gitkeep	Add new directory	1 week ago
Databricks	Sales Data	1 week ago
		2 dava ana
Databricks-Sales-Data	Append December Sales Data	3 days ago

This file contains only the alter script.

main	GitLabIntegration / FY2022 / Databricks-Sales-Data_2022-02-21_16-3 Find file Blame History Permalink
*	Append Sales Data a4005f10
🗄 Da	tabricks-Sales-Data_2022-02-21_16-30-04 🛱 6.72 KB
:	CREATE TABLE cust_dec
1	2 (
3	First_Week BYTE,
4	Second_Week BYTE,
5	5 Third_Week BYTE,
6	Fourth_Week BYTE
	7);
8	3

Use the committed FE script to generate a physical schema in your database. To generate schema, copy the FE script from your Git repository and run the script in the database.

For example, in the following Databricks database, the FE script copied from the Git repository is run.

	e	rwii	nsales SQL
D		Cmd 1	AWS_erwin_cluster Image: Control of the second
Ŧ	l	1	Use erwinsales; CREATE TABLE cust_dec
	<u> </u>	2 3 4	(First_Week BYTE,
જિ	Ĥ	4 5 6	Second_Week BYTE, Third_Week BYTE,
U	L	7	Fourth_Week BYTE);

The cust_dec table is created in a Databricks database.

Se databrick:	Data	Create ⁻	Table
D Data Science &… ▼	Databases 📀 🗸	Tables	
⊕ Create	Q Filter Databases	Q Filter Tables	
Geleale	■ customer_sc	⊞ cust	•
Workspace	■ db1_sravan	⊞ cust_credit	•
୍ବିତ୍ତ Repos	Se default	⊞ cust_dec	•
() Recents	Se demo	⊞ emp	•
() Recents	🛢 demo_db2	⊞ mo_rent_rec	-
Q Search	🛢 demo_fe	III movie	•
👌 Data	emors emors	⊞ movie_copy	•
	🛢 employees		-
์ Compute	employees2	⊞ store	•
ž⊒ Jobs	erwin5		
	erwinsales		