Oracle® Fusion Middleware

WebCenter Forms Recognition ALE Learnset Manager Installation Guide

14c (14.1.1.0.0)

F76454-02

October 2025

Describes how to install and upgrade ALE Learnset Manager

ORACLE®

Oracle Fusion Middleware Oracle WebCenter Forms Recognition ALE Learnset Manager Installation Guide, 14c (14.1.1.0.0)

F76454-02

Table of Contents

About ALE Learnset Manager

The ALE Learnset Manager or ALM is a web-based administration client that enables you to create, modify, and delete projects and classes which will be used by the Automated Learning Engine (ALE) to learn how to classify or extract data from documents. You can also edit classes by adding and removing training documents to improve the performance of each learnset.

Prerequisites

Before you start installing ALM, you need to ensure that the following applications are installed:

- Java Runtime Environment 8. Please make sure to use a version that is appropriate for your operating system (64 bit). For information related to download, visit https://www.oracle.com/java/technologies/javase/javase8u211-later-archive-downloads.html
- Apache Tomcat 9. For information related to download, visit https://tomcat.apache.org/download-90.cgi

  Note: It is strongly recommended to use Apache Tomcat 9.0.86 or later versions to ensure the best security practices and address known vulnerabilities.

- Microsoft SQL Server 64-bit (versions 2012, 2014, 2016, or 2017) Or Oracle 19c.
- ImageMagick 7.1.1-41 or 6.9.13-19. For information related to download, visit https://imagemagick.org/script/download.php#windows

Note: ImageMagick is recommended for best performance in image conversion. If ImageMagick is not installed, an internal converter is used. Make sure the option Install legacy utilities (e.g. convert) is selected during installation.

- Microsoft Visual C++ 2015 Redistributables (x64). For information related to download, visit https://www.microsoft.com/en-us/download/details.aspx?id=48145
- You can install ALE Learnset Manager on the following operating systems:

  Note: Only 64-bit operating systems are supported

  Operating systems:

  - Windows 11 Enterprise
  - Windows 10
  - Windows Server 2022
  - Windows Server 2019
  - Windows Server 2016

When working with Microsoft SQL Server, verify the following settings are in place.

- Go to **SQL Server Configuration Manager > SQL Server Network Configuration** and enable TCP/IP connections.
- Go to **TCP/IP > Properties > IPAll** and enter the desired port for your SQL server instance. The value is usually 1433.
- Activate SQL Server authentication.
- Prepare a database account that ALM can use.

Create a database user for Oracle Environment

To create a database user in an Oracle environment, complete the following steps:

1. Create a user account that is identified or authenticated by a specific password using the system or a predefined administrative account that gets generated during Oracle installation.

Note: A default tablespace and a temporary tablespace are required to create the ALM user and the schema.

2. Assign **DBA** role to the user account created in Step 1.

For more information on creating users and tablespace, refer to the Oracle Database documentation.

Create a database user for SQL Environment

For SQL environment, before installing ALM, it is recommended that you create new database user credentials in order to prevent ALM from creating tables under the master database.

To create a database user in an SQL environment, complete the following steps:

1. Create a new empty database.
2. Create a new user credential for logging on to the database server. The default database of the newly created user credentials must be mapped with the database created in Step 1.

Note: Under **User Mapping**, assign **DB Owner** role to the database user account.

Installation Process

This section describes the installation and configuration procedure for ALE Learnset Manager.

ALM is installed with the **ALMSetup.exe** installer.

For a Windows installation without an active user account control, double-click **ALMSetup.exe**.

For a Windows installation with an active user account control, right-click **ALMSetup.exe** and click **Run as administrator**.

A sequence of panels is displayed. You need to enter the configuration details on each panel and click > to proceed to the next panel. In case of an error, you need to review the log messages that are displayed simultaneously during the installation process and fix the issue before you proceed. The following high-level steps need to be performed.

Configure Target Directories

Configure Database

Configure User Authentication

Note: It is recommended to stop the tomcat server during installation and start once the installation is complete.

Configure Target Directories

The REST services and the web client are deployed as a web application to an Apache Tomcat

installation. Some native libraries are installed to a directory outside of Tomcat’s directory structure. That directory is also used as the data directory for local files.

To configure the target directories, complete the following steps.

1. In the **Installation Directories** dialog box, enter the details, as required. For more information on specific fields, refer to the following table.

Field	Description
Directory	Enter the name of the directory for native libraries and local data.
Tomcat Directory	Enter the name of the directory where Tomcat is installed. **Note:** Setup tries to resolve this directory automatically and set the correct value as the default value.
Name of Web Application	Enter the name of the web application that is deployed. This name becomes part of the URL.
HTTPS Support	The Tomcat URL will automatically use the https protocol with a secure port (For example, 8443).You can uncheck this option to disable HTTPS, which will switch the Tomcat URL to use the http protocol with a non-secure port (For example, 8080).
Tomcat URL	Enter the base URL of your Tomcat installation. **Note**: Enter the base URL of your Tomcat installation. By default, this field will use the HTTPS protocol because HTTPS Support is selected. **Note**: The full URL of the web application is the Tomcat base URL and the name of the web application (for example: ALM). This URL is case-sensitive. • Example (HTTPS Enabled): https://< server-name>:8443/ALM • Example (HTTPS Disabled): http://< server-name>:8080/ALM

2. Click > to proceed.
3. If there is an existing installation, you will be prompted if the existing files should be overwritten. You can do any of the following:

   Click **Yes** to replace the files with the latest version, Or,

   Click **No** to update the configuration.

The necessary files are copied to the target locations, and the system's PATH environment variable is updated. If updating the PATH fails for some reason you will see an error message displayed in the log window. In that case, you need to edit the PATH manually and add the values that are displayed in the error message.

Configure Database

ALE Learnset Manager stores its configuration and the training sets in a database.

To configure access to the prepared database, complete the following steps.

1. In the **Database** dialog box, enter the details, as required. For more information on specific fields, refer to the following table.

Field	Description
Database Type	Select the type of the database – either Oracle or Microsoft SQL Server.

Host	Enter the name or IP address of the host on which Oracle SQL Server are running.
Port	Enter the port for TCP/IP connections - this is usually 1521 for Oracle and 1433 for SQL Server.
Oracle SID / Service Name	When connecting to an Oracle database, enter the SID or the Oracle Service Name.
Username	Enter the username that is used to connect to the database.
Password	Enter the password that is used to connect to the database.

2. Click **>** to proceed.

The installer connects to the database and prepares the configuration repository. If there is a problem, the log message displays an error message.

Configure User Authentication

ALE Learnset Manager supports two kinds of authentication:

An LDAP server can be used to authenticate users

An internal user account can be used

Note: Password complexity is not enforced with internal users.

Both approaches can be combined with support for Single Sign-On (SSO). When configuring SSO, it is expected that the SSO provider protects the access to the web application and puts the name of the authenticated user into an **HTTP header** field.

To support user authentication via an SSO provider, complete these steps:

1. In the **SSO Header** field, enter the name of the HTTP header field that the SSO provider uses to submit the name of the authenticated user.
2. Configure your SSO provider so that the following paths within the web application are protected:

   /packages/framework-core/sso

   /service/session/
3. Click **>** to proceed, setting up details for the selected authentication type.

LDAP Authentication

There are two options for configuring LDAP authentication:

All users are stored within a single node of the directory. In this case, only the server URL and a pattern is required that defines how the distinguished name (DN) of a user is constructed.

The users are stored in a tree structure. In this case, additional information is required including a user account that can log into the LDAP server and perform a search operation for a given username.

To configure LDAP authentication with all users in a single node, complete the following steps.

1. In the **LDAP Authentication** dialog box, enter the details, as required. For more information on specific fields, refer to the following table.

Field	**Description**

Users are	In the **Users are** list, click **In a single node**.
Server URL	Enter the LDAP URL of the server. **Note**: URLs should start with ldap or ldaps and contain the name or IP address of the server and the port. Optionally, the URL can also include a root path within the directory. Example: ldap://ad.mycompany.com:389/DC=ad,DC= DC=mycompany,DC=com
User DN Template	Enter a template for distinguished names for user. Use `{user}` as a placeholder for the username. Example: uid={user},ou=employee,o=mycompany
User	Enter a user account that will be used to test the configuration. **Note**: This is an optional step.
Password	Enter the password for the user account that will be used to test the configuration. **Note**: This is an optional step.

2. Click **>** to proceed.

To configure LDAP authentication with users in a directory structure, complete the following steps.

1. In the **LDAP Authentication** dialog box, enter the details, as required. For more information on specific fields, refer to the following table.

Field	Description
Users are	In the **Users are** list, click **In a tree structure**.
Server URL	Enter the LDAP URL of the server. **Note**: URLs should start with ldap or ldaps and contain the name or IP address of the server and the port. Optionally, the URL can also include a root path within the directory. Example: ldap://ad.mycompany.com:389/DC=ad,DC= DC=mycompany,DC=com
User DN	Enter the distinguished name of a user account. This account will be used to connect to the LDAP server and perform search operations.
Password	Enter the password associated with the DN user account.
Search Filter	Enter a pattern for the filter that is used for searching the user whose authentication is to be checked. Use `{user}` as a placeholder for the username. When connecting to Active Directory, use the following pattern: `sAMAccountName={user}`.
Search Paths	Enter one or more paths that contain the users. **Note**: This is an optional step. Only valid LDAP paths are accepted (for example: ou=users). Multiple paths can be separated by semicolons. If no path is provided at all, the entire directory is searched.
User	Enter a user account that will be used to test the configuration. **Note**: This is an optional step.
Password	Enter the password for the user account that will be used to test the configuration. **Note**: This is an optional step.

2. Click **>** to proceed. The installer connects to the LDAP server to verify the configuration and performs an authentication with the test account, if applicable.

Internal Authentication

Internal authentication is designed to be used for very simple use cases that do not require any real user management at all, or as a fallback authentication type in combination with SSO.

To use internal authentication, in the **User** and **Password** fields, enter an appropriate username and the password associated with the username, respectively.

Configuring the User Lockout

If you enter invalid credentials five times in a row, you will be locked out for the specified time. The time is in milliseconds.

To configure the user lockout, complete the following steps:

1. Stop the ALM server.
2. Navigate to the install location of the ALM server (Example: ../**ALM/WEB-INF/conf**) and open **alm-learnsetmanager.xml**.
3. Find the section with the ID=**lockoutManager**.
4. The first constructor-arg is the timeout in milliseconds. This value must be greater than 0. The default value is 60000.
5. The second constructor-arg is the amount of login attempts before the user is locked out. This value must be greater than 0. The default value is 5.
6. Save and close.
7. Start ALM server.

Authorized Users

When working with LDAP authentication or internal authentication with SSO, you can configure a list of users to use the application. Other users can log in but will not be able to access any functionality.

To set up a list of authenticated users, enter one username per line.

Apache Tomcat Troubleshooting and KB Topics

Troubleshoot Login Refresh Issue

If you experience an issue where the ALM login page continuously refreshes after entering user credentials in an HTTP-based ALM setup, follow these steps to resolve it:

1. Navigate to the config.json file located in the application directory under Tomcat (..\Apache Software Foundation\Tomcat 9.0\webapps\ALM).
2. Locate and remove the "cookieSameSitePolicy": "None" entry under the "framework-core" section.
3. Save the changes to the file.
4. Refresh the ALM Web URL in your browser.

Troubleshoot Access Denied Issue

Description of issue: After installation, ALM application does not run and the “Access is denied” or "nested exception is java.lang.UnsatisfiedLinkError" error message is thrown for any of the alm-libraries (for e.g. alews-core.jar) in the tomcat log files.

To resolve this issue, complete the following steps:

1. Go to Windows Service and select **Apache Tomcat Service**.
2. Right-click **Apache Tomcat Service** and select **Properties**.
3. In the **Properties** window, click **Log On**.
4. Select **This account**.
5. Enter the username of the specific user account in the format **DOMAIN\username** or **username** if it's a local account. Make sure the username contains the required administrator privilege.
6. Click **Apply** and then **OK** to save the changes.
7. Restart Apache Tomcat Service.

Troubleshoot SSL/TLS Handshake Failures

If you encounter issues connecting to ALM via HTTPS from a third-party application, such as BFI, follow these steps to resolve SSL/TLS handshake problems:

- Verify the validity and trust status of the ALM SSL certificate.
- Ensure compatible TLS versions are enabled in both applications.

For example- If ALM Server utilizes TLS 1.2, ensure that TLS 1.2 is enabled on the machine where the third-party application is running and verify that strong cryptography settings are configured appropriately.

Configure Tomcat URL Character Restrictions

In the more recent versions of Apache Tomcat, such as 9.0.8, 8.5.31 and above, the characters that can be present in a URL has been restricted. For ALM to work correctly, this restriction must be relaxed. To configure your Apache installation, complete the following steps.

1. Modify the server.xml file in the Apache Tomcat Conf directory.
2. Find the Connector element that defines the port on which Tomcat receives requests. Typically, it is port 8080 but this could have been modified at your site.

```
<Connector port="8080" protocol="HTTP/1.1"
      xpoweredby="false" server="Web"
      connectionTimeout="20000"
      redirectPort="8443"
/>
```

4. Add the underlined lines and restart Tomcat.

```
<Connector port="8080"
      protocol="HTTP/1.1"
      xpoweredby="false" server="Web"
      connectionTimeout="20000"
      redirectPort="8443"
      relaxedPathChars='[]|'
      relaxedQueryChars='[]|{}^&#x5c;&#x60;&quot;&lt;&gt;'
/>
```

Note: You must apply this configuration to all Tomcat servers which have the ALE Learnset Manager web application installed.

Configure Tomcat X-Frame-Options Header

Injecting HTTP Response with the secure header can mitigate most of the web security vulnerabilities.

To enable secure HTTP header in Apache Tomcat, configuring ‘X-Frame-Options Header’ is very essential to prevent ‘clickjacking attack’.

To configure X-Frame-Options Header, complete the following steps.

1. Modify the web.xml file in the Apache Tomcat Conf directory.
2. In the **Built In Filter Definitions** section in web.xml, add or uncomment the following filter configuration, in case it does not exist already.

```
<filter>
  <filter-name>httpHeaderSecurity</filter-name>
   <filter-class>org.apache.catalina.filters.HttpHeaderSecurityFilter</filter-
      class>
        <async-supported>true</async-supported>
   </filter>
```

3. In the **Built In Filter Mappings** section in web.xml, add the following configuration, in case it does not exist already.

```
<filter-mapping>
      <filter-name>httpHeaderSecurity</filter-name>
      <url-pattern>/*</url-pattern>
</filter-mapping>
```

4. Save the file and restart Tomcat.

Note: You must apply this configuration to all Tomcat servers which have the ALE Learnset Manager web application installed.

Configure Cross-Origin Resource Sharing

Cross-Origin Resource Sharing (CORS) is an HTTP-header based mechanism that allows a server to indicate any origins (domain or port) other than its own from which a browser should load resources. CORS also relies on a mechanism by which browsers make a preflight request to the server hosting the cross-origin resource, to check if the server permits the actual request. In that preflight, the browser sends headers that indicate the HTTP method and headers to be used in the actual request.

The following headers are used:

Access-Control-Allow-Origin: This specifies either a single origin which tells browsers to allow that origin to access the resource or else the (*) wildcard tells browsers to allow any origin to access the resource. By default, Access-Control-Allow-Origin is set as a * wildcard.

Access-Control-Allow-Headers: This header is used in response to a preflight request which includes the Access-Control-Request-Headers to indicate which HTTP headers can be used during the actual request. By default, Access-Control-Allow-Headers is set as origin,content-type,accept,authorization.

Access-Control-Allow-Methods: This response header specifies one or more methods allowed when accessing a resource in response to a preflight request. By default, Access-Control-Allow-Methods is set for GET, POST, PUT, DELETE, OPTIONS, HEAD.

Access-Control-Max-Age: This header indicates how long the results of a preflight request can be cached. By default, Access-Control-Max-Age is set as 600 and unit is considered as seconds.

To configure the CORS Access-Control features in ALM, complete the following steps:

1. Stop the Apache Tomcat service.
2. Navigate to the **<Server directory>\ALM\WEB-INF\conf** folder.
3. Open alm**-tomcat.xml**.
4. Search for **CorsFilter** as a class.
5. Verify the values for the following Access-Controls:
   - property name = ‘allowOrigin’ refers ‘Access-Control-Allow-Origin’ and its corresponding value.
   - property name = ‘allowHeaders’ refers ‘Access-Control-Allow-Headers’ and its corresponding value.
   - property name = ‘allowMethods’ refers ‘Access-Control-Allow-Methods’ and its corresponding value.
   - property name = ‘allowMaxAge’ refers ‘Access-Control-Max-Age’ and its corresponding value.
6. Change the **CORS Access-Control** value(s), as required.

7. Save **alm-tomcat.xml**.
8. Restart Tomcat server.

Restart Apache Tomcat

Sometimes when you start and stop ALM with the Tomcat Web Application Manager, Tomcat displays the message, “Application at context path/ALM could not be started”. To prevent this issue from occurring you can either restart or update the Apache Tomcat configuration.

To restart Apache Tomcat via the services management console, complete the following steps:

1. Open the services management console.
2. Right click on the Apache Tomcat service and select **Restart** or use the **Stop / Start** options as required.

   Note: The risk with this is that the Apache Tomcat Web Application Manager could still be used to stop and start ALM and result in failures using ALM.
3. Update the Apache Tomcat configuration so that it doesn't attempt to reload the Java Native Interface every time the web application is started.

   Note: You must perform the steps above to every Tomcat web server that is running ALE Learnset Manager.

Update the Apache Tomcat configuration

To update the Apache Tomcat configuration, complete the following steps.

Note: Before proceeding with the steps below, ensure that a backup is taken of the Tomcat installation area.

1. Ensure that all dependent systems are not being used. If required, stop the relevant services and/or websites.
2. Stop the Apache Tomcat service.
3. Create a new folder called shared under the **lib** folder within the Tomcat installation area.
4. Browse to the Apache Tomcat folder where ALE Learnset Manager is installed. Example: **...\WEB-INF\lib**.
5. Move the **columbusJNI.jar** file from the location in step 4 to the new folder created in step 3.
6. Browse to the conf folder within the Tomcat installation area and open the catalina.properties files in a text editor.
7. Search for the `shared.loader` entry and update this as follows:

```
shared.loader="${catalina.base}/lib/shared","${catalina.base}/lib/shared/*.jar","${catalina.home}/lib/shared","${catalina.home}/lib/shared/*.jar"
```

8. Restart the Apache Tomcat service.
9. Restart any services or websites stopped in step 1.

Configuring HTTPS Support for ALM

Previously, ALM supported HTTP protocol only. However, for enhanced security and data protection, ALM now supports HTTPS protocol as default behavior. This protocol encrypts communications between the client and the server. It is highly recommended to use HTTPS as the default protocol to ensure data privacy and integrity. This topic explains the steps to configure HTTPS in ALM hosted on Apache Tomcat.

Before configuring HTTPS, ensure you have the following:

- A valid .p12 certificate (also called a PKCS12 certificate) provided by a trusted Certificate Authority (CA). A .p12 certificate contains both the public and private keys necessary for secure communication.

If you do not have a .p12 certificate, you can also generate a self-signed certificate for testing purposes. You can create a self-signed certificate using the following command:

```
```bash
keytool -genkeypair -alias almssl -keyalg RSA -keysize 2048 -storetype PKCS12 -
keystore almcert.p12 -validity 365
- almssl: Alias for the certificate.
- almcert.p12: The name of the generated .p12 file.
- validity 365: Number of days the certificate will be valid for.
```

You will be prompted to provide a password and other details for the certificate.

After you have obtained a .p12 certificate, you can configure Apache Tomcat to use it for HTTPS. To configure HTTPS:

1. Locate the server.xml file. This file is usually located in the <TOMCAT_HOME>/conf/ directory.
2. Open the server.xml file in a text editor, and locate the <Connector> element for HTTP.
3. Add or modify the following HTTPS configuration:

```
<!-- Define an SSL/TLS HTTP/1.1 Connector on port 8443 -->
   <Connector
      protocol="org.apache.coyote.http11.Http11NioProtocol"
      port="8443"
      maxThreads="150"
      SSLEnabled="true"
       scheme="https"
        secure="true">

      <SSLHostConfig>
         <Certificate certificateKeystoreFile="conf/almcert.p12"
                      certificateKeystoreType="PKCS12"
                     certificateKeystorePassword "your_keystore_password"
                     certificateKeyAlias="<Alias for the certificate>"
```
```

```
                        sslProtocol="TLS"
 />
      </SSLHostConfig>
   </Connector>


   - port: Set to `8443`, which is the default HTTPS port.
   - certificateKeystoreFile: The path to your .p12 certificate.
   - certificateKeystorePassword: The password you set when generating the
certificate.
   - sslProtocol: This should be set to `TLS` (Transport Layer Security), the
successor to SSL.
```

Note: In SSLHostConfig, the "alias" which user had given while generating .p12 certificate should match the certificateKeyAlias in server.xml file. For example: If "alias" for the .p12 certificate is given as "almssl" , then certificateKeyAlias="almssl".

4. Save and close the server.xml file.
5. Ensure that the firewall is configured to allow traffic through port 8443.
6. Restart the Tomcat service for the changes to take effect. This can be done by executing the following command:

```
./bin/shutdown.bat
./bin/startup.bat
```

Or, you can also start and stop the Tomcat service from Services.

7. Test HTTPS Access.
    a. Open a browser and navigate to the ALM instance using the HTTPS URL.
    b. https://<servername>8443/ALM
        Note: A security warning is displayed in case of a self-signed certificate. This can be bypassed for testing purposes. However, for production, you must use a certificate from a trusted CA.
    c. Confirm that ALM is now accessible over HTTPS.
8. (Optional) Redirect HTTP to HTTPS to ensure that all users access ALM through the secure HTTPS protocol. To configure a redirection from HTTP to HTTPS:
    a. Open the server.xml file.
    b. Add a redirection rule in the HTTP <Connector> element:

```
<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000"
               redirectPort="8443" />
   ```
```

This will automatically redirect any HTTP traffic on port 8080 to HTTPS on port 8443.

**Note:** If there are other applications (HTTP) hosted on 8080 port make sure those are running fine after the tomcat configuration change.
```

Certificate Errors: If you encounter issues with your SSL certificate, ensure the certificate is correctly imported into the .p12 file and that the file path and password in server.xml are accurate.

Port Conflicts: If port 8443 is already in use, modify the port attribute in the HTTPS connector configuration to use an available port.

Upgrade Procedure

This section provides high level information on how to upgrade to the latest version of ALE Learnset Manager.

ALM 2.2.x to ALM 24.1

ALM 24.1 introduces support for the HTTPS protocol by default, with an option to use HTTP if preferred, during installation. Before upgrading, ensure that you take the backup of the ALM directories and ALM database and your environment meets all prerequisites and is configured correctly. If HTTPS is to be used, ensure the environment is HTTPS-enabled before starting the upgrade. For more information, refer to Configuring HTTPS Support for ALM.

Locate the current installation (2.2.x) directory of ALM on the server. Example: ...**\ALM\WEB-INF\conf**. Open **alm-config-db.xml** and make a note of the username. This is required during the installation of ALM 24.1.

To upgrade ALM, complete the following steps:

1. Ensure that all the existing ALM servers are stopped. There can be multiple servers located in different machines.
2. Remove ALE Learnset Manager by deleting the folder **C:\Program Files\Apache Software Foundation\Tomcat<installed version>\webapps\ALM** and **C:\ALM**.
3. Run the ALM 24.1 installer.
4. Select the HTTP or HTTPS preference.
   By default, the **HTTPS Support** check box is selected, and the Tomcat URL is set to the following format: https://<Host_Name_With_Fully_Qualified_Domain_Name>:<Port> (Example: https://<Server Name FQDN>:8443/).

   If HTTP is preferred, clear the **HTTPS Support** checkbox and update the Tomcat URL to the following format: http://<Host_Name_With_Fully_Qualified_Domain_Name>:<Port> (Example: http://<Server Name FQDN>:8080/)
5. Install ALE Learnset Manager on the same machine using the same database user credential that was used with the previous ALE Learnset Manager installation.
   Note: Do not delete the database.

6. Start the ALM servers.
7. Verify if **Starting: ALE Learnset Manager 24.1** is printed in the **alm-ws.log**.
8. Before logging on to ALE Learnset Manager, clear your browser history and cached files.
9. Ensure the entry <ALM data directory>\lib\win64 (e.g., C:\ALM\lib\win64) exists in the System Path variable within the Environment Variables section.
10. Log on to ALM Web GUI and verify that projects created in ALM 24.1 are available.
    Post Installation steps:
    a. Open the configuration for each project and confirm the following:
        i. The Use Classification Parameters checkbox is checked.
        ii. Threshold is set to 75 (default value).
        iii. Distance is set to 0 (default value).
    b. Open the ALM database and verify:
        i. The LEARNSET_DOC table contains all documents from ALM 23.1, with timestamps in the newly added creation_date column.
        ii. Two new tables, TMPALMDOCUMENT and TMPALMFIELDS, are present.
11. After upgrading it is recommended to learn the existing project(s).
12. Note: During upgrade, previous user credentials get deleted and new user credentials (provided by the user during installation) are stored in the database in an encrypted format. In case the new user credentials are same as previous version's (ALM 23.1) credentials, then the user account is recreated in an encrypted format using the dynamic key generated during installation.

Restore the database from a Backup.

You only need to restore the database tables if the existing database becomes corrupted during the installation process.

To restore the database from a backup, complete the following steps:

1. Create the database to which the data needs to be migrated.
2. Install ALE Learnset Manager and configure it to use the new database.
3. Start the tomcat server.
4. Execute the process to import the backup data into the new database.
5. Restart the tomcat server.
6. Login to ALE Learnset Manager using the default administrator account.

Note: For Oracle database backup and restoration, the LEARNSET_DOC, LEARNSET_DOC_IMAGE, LEARNSET_PROJECT_DATA tables contain BLOB/CLOB data. To migrate this data type, it is recommended to follow the Oracle Data Pump procedure. For more information on Oracle Data Pump table export and import, refer to the Oracle Database documentation.