Quest

Software Installation Guide

Quest Data Intelligence (DI)

Version 16.0

Linux Installation Guide

This document provides the instructions to install the new 16.0 version of Quest Data Intelligence on a Linux OS.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.

Attn: LEGAL Dept

4 Polaris Way Aliso Viejo, CA 92656

Refer to our Web site (https://www.quest.com) for regional and international office information.

Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at https://www.quest.com/legal.

Trademarks

Quest, the Quest logo, Quest Data Intelligence, erwin by Quest are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit https://www.quest.com/legal/trademark-information.aspx. All other trademarks and registered trademarks are the property of their respective owners.

Table of Contents

About this Guide....1
Software Solution Architecture....1
System Specifications and Software Requirements....3
Getting the Quest DI 16.0 software ready....6
Installing the Quest Data Intelligence software....6
Detailed Walkthroughs for Installation of Quest Data Intelligence on Linux OS....7
1 - Red Hat Enterprise Linux 8 & 9....8
2 - Ubuntu Server, and Debian....24

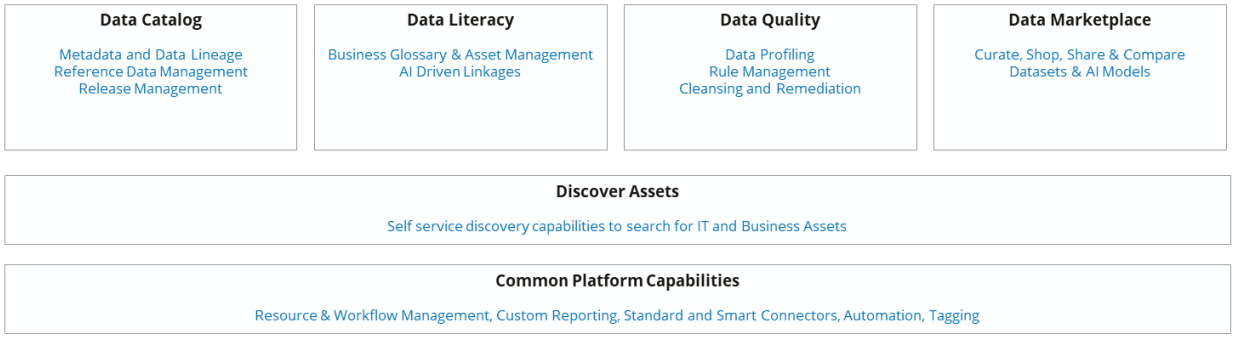
About this Guide

This document describes the installation process of the Quest Data Intelligence application on a dedicated on-premises physical or virtual server, as well as cloud based virtual machines. It provides the software installation procedure for a basic HTTP installation of Quest Data Intelligence, configuration tasks, and troubleshooting information. This document also describes the technical specifications, and the pre-requisites required for the successful installation of Data Intelligence software on a supported Linux Distribution.

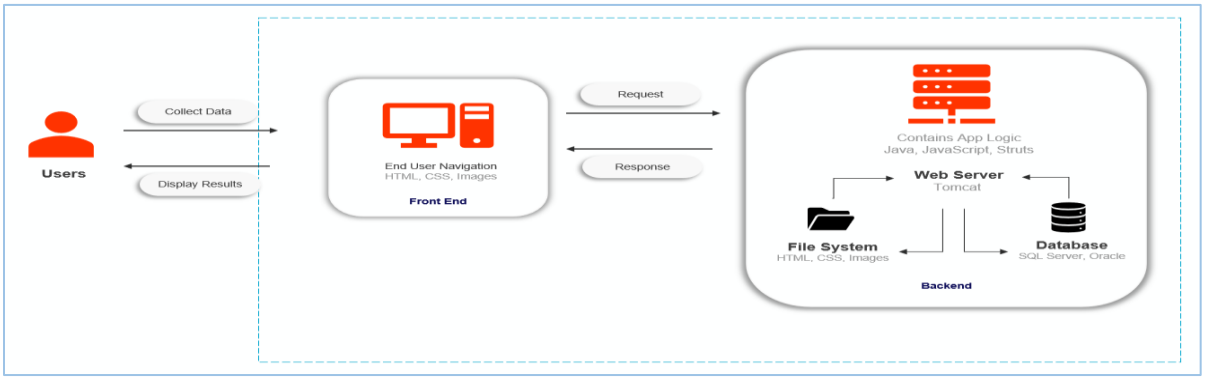
Software Solution Architecture

Key Components

The following diagram shows a high-level modular architecture of the application.



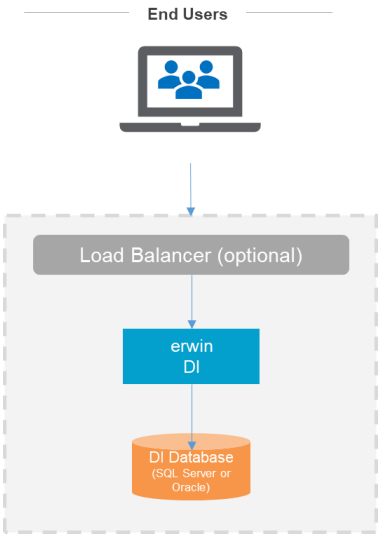
Web Application Architecture



Tiers

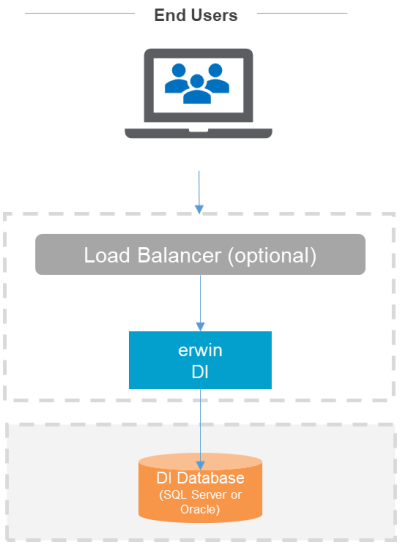
The Quest Data Intelligence application supports both single server (application and database on the same server) and distributed (application and database on the different servers and/or multiple app instances under a load balanced set up) architectures.

Single Server Setup



- Only Recommended for POC or Demo environments

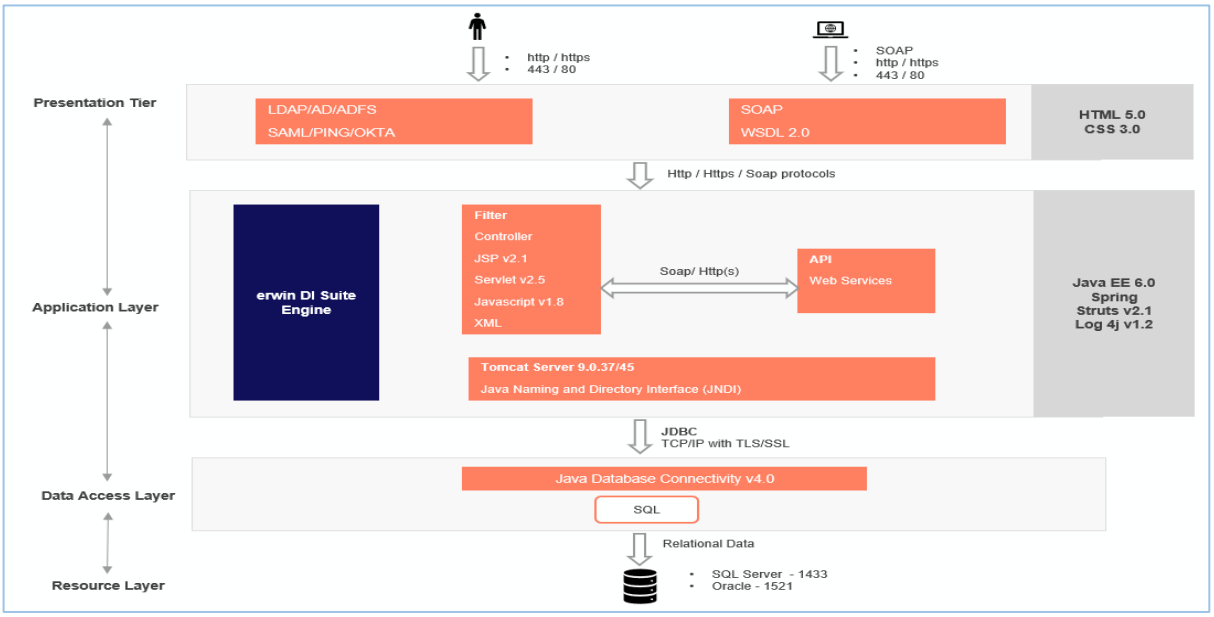
Distributed Server Setup



- Highly Recommended for Production Deployments

Technology Stack and Components

The Quest DI application follows multi-tier architecture consisting of Presentation, Application, Data Access, and Resource layers. The following is a high-level diagram depicting these layers.



System Specifications and Software Requirements

Important Note: The following specifications are for the Quest Data Intelligence application only and do not include the Quest Data Quality module (DQLabs). We recommend the erwin Data Quality (DQLabs) be installed on a separate server.

For Production Deployments

Application Tier – Minimum Compute & Software Requirements	
Node Options	Single / Multi
Operating System	Windows Server / Linux Server
Processor	64 Bit
CPU Cores / vCPUs / RAM [1]	4 Cores / 8 vCPUs / 64 GiB RAM recommended (32 GiB RAM Minimum)
Local Storage	100 -200 GB
Java JDK	Eclipse Temurin Adoptium JDK version 17.0.x
Java Servlet Container / Web Server	Apache Tomcat version 10.1.x
Web Browsers	MS Edge (v86.0+), Google Chrome (v86.0+), Firefox (v82.0+)

- [1] RAM GiB required is based on the number of concurrent users that will use the application.
  For optimal performance, we recommend about a minimum of 0.5 GB space per login user on the application server.
- If you have 30 users logging in concurrently, the application will need to have a minimum of 15 GB (30*0.5=15) free RAM space allocated to it. This is not the RAM of the server machine. It is the physical RAM allocated to the application server (tomcat JVM) itself.

Database Tier – Minimum Compute & Software Requirements	
Database Server	MS SQL Server: 2016, 2017, 2019, 2022 Oracle Database: 18c, 19c
Processor	64 Bit
CPU Cores / vCPUs / RAM	4 Cores / 8 vCPUs / 64 GiB RAM recommended (32 GiB RAM Minimum)
Database Storage	200 GB storage recommended as minimum starting size. Oracle Table Space 100 GB recommended as minimum starting size.

- Quest DI requires a dedicated database/schema, NOT a dedicated server instance.
- The storage/tablespace allocated initially will need to increase over time based on product usage and data growth.
- Azure Cloud databases supported: Azure SQL Managed Instance, Azure SQL Database (PaaS) or SQL Server in a VM.
  AWS Cloud databases supported: AWS RDS SQL or AWS RDS Oracle.

Operating Systems Supported	
Microsoft Windows	Windows Server 2016 and above
Linux Distributions	Linux Versions (Linux Kernel version 4.18 and above)
Amazon Linux	2023
Red Hat Enterprise Linux	v8, v9
SUSE Enterprise / openSUSE	15 sp4 / Leap 15.4
Ubuntu Server	20.04 LTS, 22.04 LTS

- Server class operating system is recommended for production deployments.
- Choice of operating system should be based on customer's skill set and ability to support, manage, maintain the server.

Suggested Cloud Instance Sizing			
Azure VM Series		Amazon EC2 Instance Types	
Intel	(8vCPU/64 GiB) Standard_E8s_v5, Standard_E8ds_v5	Intel	(8vCPU/64 GiB) r6i.2xlarge, r6id.2xlarge r5.2xlarge, r5d.2xlarge
	(8vCPU/32 GiB) Standard_D8s_v5, Standard_D8ds_v5		(8vCPU/32 GiB) m6i.2xlarge, m6id.2xlarge, m5.2xlarge, m5d.2xlarge
AMD	(8vCPU/64 GiB) Standard_E8as_v5, Standard_E8ads_v5	AMD	(8vCPU/64 GiB) r5a.2xlarge, r5ad.2xlarge, r6a.2xlarge
	(8vCPU/32 GiB) Standard_D8as_v5, Standard_D8ads_v5		(8vCPU/32 GiB) m6a.2xlarge, m5a.2xlarge, m5ad.2xlarge
Azure E-series memory optimized VM types recommended Azure Application Gateway or third-party Layer 7 load balancer required for multi-node deployments. Suggested sizes are a starting point only, you may need to upsize instances based on concurrent usage and performance needs		AWS r-family memory optimized instance types recommended. Application Load Balancer or third-party Layer 7 load balancer required for multi-node deployments.	

Note: We highly recommend that you stay compliant with the above-mentioned system requirements for the best experience. In case you need to use a software (database version, browser etc.) that is not listed in the above system requirements, we recommend that you reach out to your erwin support or professional services contact so we can provide a recommendation on the compatibility.

For Proof of Concept/Development Server

Application Tier – Minimum Compute & Software Requirements	
Node Options	Single / Multi
Operating System	Windows Server / Linux Server
Processor	64 Bit
CPU Cores / vCPUs / RAM [1]	2 Cores / 4 vCPUs / 32 GiB RAM recommended (16 GiB RAM Minimum)
Local Storage	100 -200 GB
Java JDK	Eclipse Temurin Adoptium JDK version 17.0.x
Java Servlet Container / Web Server	Apache Tomcat version 10.1.x
Web Browsers	MS Edge (v86.0+), Google Chrome (v86.0+), Firefox (v82.0+)

- [1] RAM GiB required is based on the number of concurrent users that will use the application.
  For optimal performance, we recommend about a minimum of 0.5 GB space per login user on the application server.
- If you have 30 users logging in concurrently, the application will need to have a minimum of 15 GB (30*0.5=15) free RAM space allocated to it. This is not the RAM of the server machine. It is the physical RAM allocated to the application server (tomcat JVM) itself.

Database Tier – Minimum Compute & Software Requirements	
Database Server	MS SQL Server: 2016, 2017, 2019, 2022 Oracle Database: 18c, 19c
Processor	64 Bit
CPU Cores / vCPUs / RAM	2 Cores / 4 vCPUs / 32 GiB RAM recommended (16 GiB RAM Minimum)
Database Storage	100 GB storage is recommended as minimum starting size. Oracle Table Space 75 GB recommended as minimum starting size.

- Quest DI requires a dedicated database/schema, NOT a dedicated server instance.
- The storage/tablespace allocated initially will need to increase over time based on product usage and data growth.
- Azure Cloud databases supported: Azure SQL Managed Instance, Azure SQL Database (PaaS) or SQL Server in a VM.
  AWS Cloud databases supported: AWS RDS SQL or AWS RDS Oracle.

Operating Systems Supported	
Microsoft Windows	Windows Server 2016 and above
Linux Distributions	Linux Versions (Linux Kernel version 4.18 and above)
Amazon Linux	2023
Red Hat Enterprise Linux	v8, v9
SUSE Enterprise / openSUSE	15 sp4 / Leap 15.4
Ubuntu Server	20.04 LTS, 22.04 LTS

- Server class operating system is recommended for production deployments.
- Choice of operating system should be based on customer's skill set and ability to support, manage, maintain the server.

Suggested Cloud Instance Sizing			
Azure VM Series		Amazon EC2 Instance Types	
Intel	(4vCPU/32 GiB) Standard_E8s_v5, Standard_E8ds_v5	Intel	(4vCPU/32 GiB) r6i.2xlarge, r6id.2xlarge r5.2xlarge, r5d.2xlarge
	(4vCPU/16 GiB) Standard_D8s_v5, Standard_D8ds_v5		(4vCPU/16 GiB) m6i.2xlarge, m6id.2xlarge, m5.2xlarge, m5d.2xlarge
AMD	(4vCPU/32 GiB) Standard_E8as_v5, Standard_E8ads_v5	AMD	(4vCPU/32 GiB) r5a.2xlarge, r5ad.2xlarge, r6a.2xlarge
	(4vCPU/16 GiB) Standard_D8as_v5, Standard_D8ads_v5		(4vCPU/16 GiB) m6a.2xlarge, m5a.2xlarge, m5ad.2xlarge
Azure E-series memory optimized VM types recommended. Azure Application Gateway or third-party Layer 7 load balancer required for multi-node deployments.		AWS r-family memory optimized instance types recommended. Application Load Balancer or third-party Layer 7 load balancer required for multi-node deployments.	
Suggested cloud instance sizes are a starting point only. Upsizing may be required based on concurrent usage and performance needs.			

Pre-requisites to install Quest DI

Eclipse Temurin Adoptium Java JRE and Tomcat webserver are standard prerequisites to install and deploy the Quest Data Intelligence application.

The Quest Data Intelligence 16.0 software is certified to run on the following versions of Tomcat and Java.

Tomcat Webserver	Tomcat 10.1.x
Java	Eclipse Temurin Adoptium 17.0.x

* Important Note: The Quest Data Intelligence v16.0 has been officially certified on Tomcat 10.1.x and Java 17.0.x. We recommend that you install tomcat 10.1.x versions to avoid any compatibility issues. If you are on the older Tomcat 8x or 9x version, it is *mandatory* that you upgrade to Tomcat 10.1.x before installing DI 16.0.

Additional Note

- We recommend that you use the certified versions of Tomcat and Java for the best experience. In case you need to use a point version that is above or below the certified versions, the product might still work as expected on the non-conformant point versions, but we recommend that you reach out to your erwin support or professional services contact so we can provide a recommendation on the compatibility.
- From a *best practice perspective*, although not mandatory, we recommend that you install the Java and Tomcat software versions and the Quest DI application on the *D drive* (versus C drive as the C drive is typically reserved for server maintenance and monitoring tools). This also avoids filling up the C: drive and preventing the physical server from starting.

Memory Allocation to Web Server

Allocate memory as high as possible to the tomcat web server based on the RAM size of the server.

E.g. If the server has a 32 GB RAM, the web server needs to be allocated a minimum of 50% of the RAM to begin with i.e. 16 GB minimum. The higher the memory allocation, the better for the functioning of the application.

An example of the recommended Memory allocation to Tomcat would look as follows:

Physical RAM on Server	Allocation to Tomcat
16 GB	12 GB
32 GB	16 – 28 GB
64 GB	48 – 54 GB

Note: 32-64 GB is recommended for Production installs, while 16 GB is recommended for Proof of Concepts (POCs)/Dev.

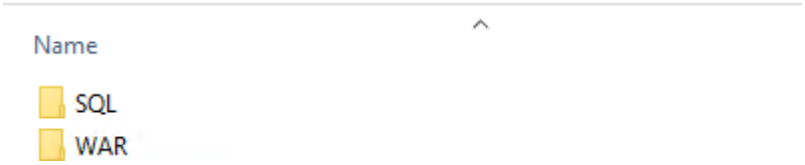
End-user Machine/Laptop Specifications

End User Machine/Laptop Configuration	
Processor	i3 and above
Minimum RAM	8 GB
Minimum Free Space available	1 – 2 GB

- The CPU should have minimum 1 – 2 GB RAM free space while accessing the Quest Data Intelligence application via a web browser.
- e.g., If you have a 4GB laptop and any application is occupying 100%CPU space, then the Quest Data Intelligence web pages will not load until some physical memory is freed up.

Getting the Quest DI 16.0 software ready

Download the Quest DI 16.0 software zip file and store it in an accessible Linux directory (or another equivalent drive) that can be easily accessed during the installation process.



Installing the Quest Data Intelligence software

Installing Quest Data Intelligence on Linux OS is easy and straight forward by following these 7 high level steps:

Step 1: Install latest Linux OS patches and security updates.

Step 2: Create Linux user 'QuestDI' for tomcat service account.

Step 3: Install certified version Adoptium OpenJDK from Eclipse Foundation.

Step 4: Download and Install certified version of Apache Tomcat v10.1.x.

Step 5: Download and Deploy Quest Data Intelligence v16.0.

Step 6: Create and Configure the database for QuestDI v16.0.

Step 7: Access the Quest DI Login screen.

Detailed Walkthroughs for Installation of Quest Data Intelligence on Linux OS

Detailed walkthroughs of the 7 high-level steps and their related sub-tasks are organized into separate chapters for each supported Linux distribution.

Pause for an installation readiness check!
Before proceeding with installation…

Verify all system requirements are met.
Review the minimum system requirements for: CPU, RAM, Storage, and supported Linux OS version before proceeding.

Verify Internet Access is available.
The Linux server will require Internet access to download Linux OS updates, and the various packages and components required for completing the installation.

Linux 'sudo' access rights required.
Before beginning, verify your Linux user account can elevate its privileges using 'sudo' to allow commands to execute as the root user.

Additional Administrator rights.
During installation you will need to create a database schema and execute DDL scripts to configure the schema for the application. You will require DBA admin rights to complete this step.

You may also require an administrator to assist with updating firewall rules, create a DNS hostname, create TLS/SSL certificates, or perform other tasks requiring Administrator rights.

Check if selinux is enabled.
If selinux is enabled on your Linux server (or an equivalent tool that manages the capability to run executables), please make sure that it can be disabled for the install or the executable for Tomcat is added to the permissive list (this will need to be done ongoing, but for the initial install it can be disabled to allow the process to complete).

Are you ready?
Skip to the chapter page for your chosen Linux distribution to continue the installation. Let's go!

1 - Red Hat Enterprise Linux 8 & 9

1.1 - Update Linux OS repos, install patches, security updates and other packages.

```
sudo dnf upgrade -y
```

1.2 - Create Linux user 'QuestDI' for tomcat service account.

As a security best practice, Apache Tomcat should never be run under "root" user account.
We recommend creating a Linux system account user that is restricted from shell login.

With security best practices in mind, using the commands below we will create a system account user named: '**QuestDI**' with home folder path **`/opt/QuestDI`** and the user is restricted with no shell access.

```
sudo useradd -r -c "QuestDI service" -m -d /opt/QuestDI -s /sbin/nologin -U QuestDI
```

1.2.1 - Set up additional folders in the QuestDI home folder.

The following will create some folders in `/opt/QuestDI` used by Quest Data Intelligence. The folders will be owned by the user and group 'QuestDI' which is the service account used by Tomcat.

Note: the '\' indicates a multiline shell command. Be sure to copy/paste the entire text block to your shell.

```
sudo -u QuestDI mkdir -p \
/opt/QuestDI/Downloads \
/opt/QuestDI/DISTemp \
/opt/QuestDI/discover_assets \
/opt/QuestDI/iccdocuments
```

1.3 - Install certified Adoptium OpenJDK from Eclipse Foundation

Quest\erwin recommends installing the certified version of JDK 17 using the package manager for your chosen Linux distribution. The instructions below provide the steps to install Adoptium JDK using Linux (RPM/DEB) installer packages.

The Adoptium JDK 17 installer updates the Linux 'alternatives' system to set the default java. We do not need to set the JAVA_HOME environment variable as we will configure that in a later step for the tomcat systemd unit file.

For additional information about installing Adoptium OpenJDK please refer to the official documentation from the Adoptium site at the following URL:
https://adoptium.net/installation/linux/

1.3.1 - Add the Adoptium repository to your Linux distribution.

Note: Be sure to copy/paste the entire text block to your shell.

```
sudo tee -a /etc/yum.repos.d/adoptium.repo >/dev/null << EOF
[Adoptium]
name=Adoptium
baseurl=https://packages.adoptium.net/artifactory/rpm/rhel/\$releasever/\$basearch
enabled=1
gpgcheck=1
gpgkey=https://packages.adoptium.net/artifactory/api/gpg/key/public
EOF
```

1.3.2 - Update the repository cache.

```
sudo dnf upgrade --repo Adoptium
```

1.3.3 - Install the specific version of Adoptium JDK certified for Quest Data Intelligence

```
sudo dnf install -y temurin-17-jdk-17.0.12.0.0+7
```

1.3.4 - Disable the Adoptium repository.

To prevent unintended future JDK 17 version updates, Quest\erwin recommends disabling the Adoptium repository to prevent unintended installation of future JDK 17 version updates.

This step is optional but recommended.
Installing uncertified versions of JDK 17 may cause unexpected results.
Disabling the repository will avoid potential problems.

```
sudo dnf config-manager --disable Adoptium
```

1.3.4.1 - Verify the Adoptium repository is disabled.

```
dnf repolist Adoptium
```

The output should show the repository status is disabled. Preventing unintended updates.

```
[erwin@rhel8 ~]$ dnf repolist Adoptium
Not root, Subscription Management repositories not updated
repo id                                 repo name                                 status
Adoptium                                Adoptium                                  disabled
[erwin@rhel8 ~]$
```

1.3.4.1.1 - Enable the repository again if needed for update.

If you require installing a newer JDK from the Adoptium repository, you can enable the repo again with the following command:

```
sudo dnf config-manager --enable Adoptium
```

1.4 - Download and Install Apache Tomcat v10.1.x

Quest\erwin QA has certified specifically Apache Tomcat version 10.1.x for use with the Quest Data Intelligence v16.0 release. It is strongly recommended to install only a certified version of Apache Tomcat.

1.4.1 - Download the Apache Tomcat tar to `/opt/QuestDI/Downloads` directory.

Note: the '\' indicates a multiline shell command. Be sure to copy/paste the entire text block to your shell.

```
sudo -u QuestDI wget -P /opt/QuestDI/Downloads \
https://archive.apache.org/dist/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
```

1.4.1.1 - Unpack the tar to QuestDI home directory.

```
sudo -u QuestDI tar -xvf /opt/QuestDI/Downloads/apache-tomcat-10.1.28.tar.gz \
-C /opt/QuestDI
```

1.4.1.2 - Create a symbolic link for the apache-tomcat-10.1.x path.

Creating a symbolic link will make it easier to reference the tomcat installation path, and to manage future upgraded versions.

```
sudo -u QuestDI ln -s /opt/QuestDI/apache-tomcat-10.1.28 /opt/QuestDI/tomcat
```

1.4.1.3 - Verify the contents of the QuestDI home directory to see the results.

```
sudo ls -l /opt/QuestDI/
```

Your output should now match the following example (Tomcat version may vary):

```
[erwin@rhel8 ~]$ sudo ls -l /opt/erwindis/
total 4
drwxr-xr-x. 9 erwindis erwindis 4096 Jun 13 17:32  apache-tomcat-10.1.17
drwxr-xr-x. 2 erwindis erwindis    6 Jun 13 16:58 discover_assets
drwxr-xr-x. 2 erwindis erwindis    6 Jun 13 16:58 DISTemp
drwxr-xr-x. 2 erwindis erwindis    6 Jun 13 17:32 Downloads
drwxr-xr-x. 2 erwindis erwindis    6 Jun 13 16:58 iccdocuments
lrwxrwxrwx. 1 erwindis erwindis   34 Jun 13 17:34 tomcat -> /opt/erwindis/ apache-tomcat-10.1.17
[erwin@rhel8 ~]$
```

1.4.2 - Configure a systemd service unit file to manage starting/stopping tomcat as a service.

We recommend managing the start/stop/restart of apache tomcat service with the built in Linux SYSTEMD system. We use the 'tee' command below to create a new **systemd** service unit file named **QuestDI.service** located in `/etc/systemd/system`.

Caution!

The example below assumes Tomcat version 10.1.28, sets the JAVA_HOME path, and assumes the tomcat install path to be `/opt/QuestDI/tomcat` and will allocate **24 GiB RAM** for the Java Virtual Machine (JVM).

Review the paths and the settings below, and modify if needed, to set correct values for your environment. It is recommended to set the -Xms and -Xmx settings to 75% of the server's memory.

-Xms and -Xmx should be set to equal RAM values or -Xms is 50% of -Xmx for efficient java garbage collection.

1.4.2.1 - Create the /etc/systemd/system/QuestDI.service file.

Copy/Paste the entire text block to your shell to create the service unit file using the 'tee' command (spans two pages)

```
sudo tee -a /etc/systemd/system/QuestDI.service >/dev/null << EOF
[Unit]
Description=Tomcat 10.1.28 servlet container for QuestDI v14
Documentation=https://support.quest.com/erwin-data-intelligence-suite/
Wants=network.target
After=syslog.target network.target

[Service]
Type=forking
Restart=always
RestartSec=5
TimeoutStartSec=300

# Tomcat should never be running as 'root'
# Here we specify a user account created as a service account for tomcat.
# Be sure the user and group with correct permissions has been created.
User=QuestDI
Group=QuestDI

# Port 443 is a privileged port on Linux OS
# To allow tomcat to bind to ports below 1024 without running it as root
# Uncomment the next line to allow 443 instead of 8443 for ssl connector
#AmbientCapabilities=CAP_NET_BIND_SERVICE

# Set JAVA_HOME path for Adoptium temurin-17-jdk
Environment="JAVA_HOME=/usr/lib/jvm/temurin-17-jdk/"

# Setting the java.security.egd system property to use/dev/urandom configures
# the Tomcat JVM to use /dev/urandom as the source of entropy for the SecureRandom
# class to avoid potential delays during startup.
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom"

# Set CATALINA environment variables
Environment="CATALINA_BASE=/opt/QuestDI/tomcat"
Environment="CATALINA_HOME=/opt/QuestDI/tomcat"
Environment="CATALINA_PID=/opt/QuestDI/tomcat/temp/tomcat.pid"
# Set the JVM memory allocation.
# As a best practice for efficient garbage collection it is recommended to
# set -Xms value equal to -Xmx, or set -Xms at 50% of -Xmx
# Quest Data Intelligence requires 16G as a minimum.
# Each user session requires 0.5 GB of RAM, so if 48 concurrent users are
# expected to login to the application, the recommended JVM memory allocation
# would be: -Xms=24G -Xmx=24G or -Xms-12G -Xmx=24G.
Environment="CATALINA_OPTS=-Xms24G -Xmx24G -server -XX:+UseParallelGC"

# Setting the java.security.egd system property to use/dev/urandom configures
# the Tomcat JVM to use /dev/urandom as the source of entropy for the SecureRandom
# class to avoid potential delays during startup.
# QuestDI v13.x and 14.x require additional Java parameter options to be set
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom --add-opens=java.base/java.lang=ALL-
UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --
add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-opens=java.base/java.net=ALL-UNNAMED --add-
opens=java.management/sun.management=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-
opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --add-
opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.util.regex=ALL-UNNAMED --
add-opens=java.base/java.net=ALL-UNNAMED --add-exports java.base/sun.nio.ch=ALL-UNNAMED  --add-
opens=java.base/java.util.concurrent=ALL-UNNAMED"

# Paths to Start/Stop scripts
ExecStart=/opt/QuestDI/tomcat/bin/startup.sh
ExecStop=/opt/QuestDI/tomcat/bin/shutdown.sh

# The SuccessExitStatus=143 directive tells systemd to consider the exit code 143 (which is what
# Java returns when it is terminated by a SIGTERM signal) as a successful termination.
# This can help ensure that systemd allows Quest DIS Lineage Graph Export to complete before
```

```
# the java process is terminated
SuccessExitStatus=143
TimeoutStopSec=60

[Install]
WantedBy=multi-user.target
EOF
```

Reload systemd daemons.

```
sudo systemctl daemon-reload
```

1.4.2.2 - Enable the QuestDI.service file to start tomcat during server reboots.

```
sudo systemctl enable QuestDI.service
```

1.4.2.3 - Verify the unit file is enabled.

```
sudo systemctl status QuestDI.service
```

1.4.2.4 - Start the tomcat server using the QuestDI.service unit file with the following command.

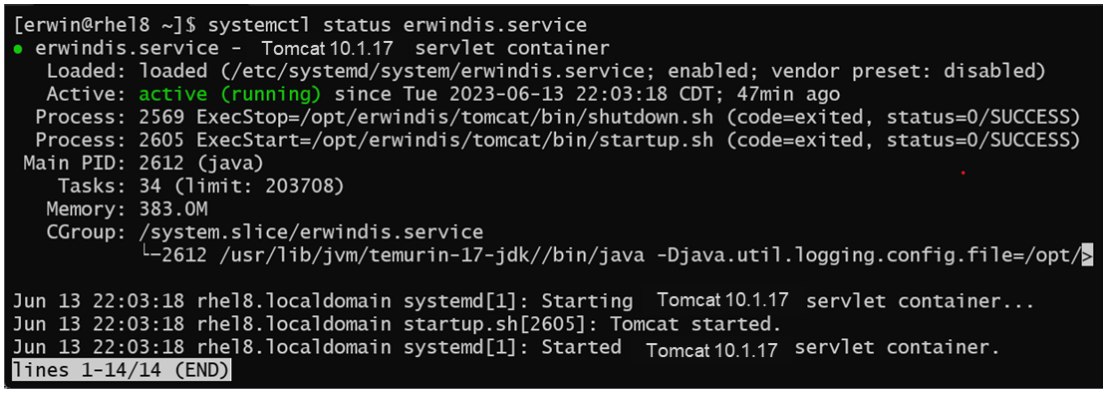
```
sudo systemctl start QuestDI.service
```

1.4.2.5 - Check status to view the result.

Verify tomcat is running using the QuestDI.service unit file.

```
sudo systemctl status QuestDI.service
```

Your result should be like:



If you see Active: active (running) we are good to go. Leave tomcat running as we will soon need it when deploying the Quest Data Intelligence webapp WAR file.

1.4.3 - Configure the RHEL 8 firewall to allow inbound port tcp/8080.

This section covers opening tcp ports on the default firewall included with RHEL 8, firewalld. If you are employing a different type of firewall, please refer to your firewall's instructions to create the allow rule required to open access to the tomcat connector on port 8080/tcp.

1.4.3.1 - Open port 8080/tcp and make the rule persistent across reboots.

```
sudo firewall-cmd --permanent --zone=public --add-port=8080/tcp
```

1.4.3.2 - Reload the firewall configuration.

```
sudo firewall-cmd --reload
```

1.5 - Download and Deploy Quest Data Intelligence v16.0

Customers should download the latest v16.0 GA release zip file from http://support.quest.com to ensure they are obtaining the latest GA release.

1.5.1 - Download the erwin_Data_Intelligence_v16.0 zip file.

The following command block downloads the zip file from the https://support.quest.com site. Then unzips the file and copies the QuestDI.war file to the `/opt/QuestDI/tomcat/webapps/` directory.

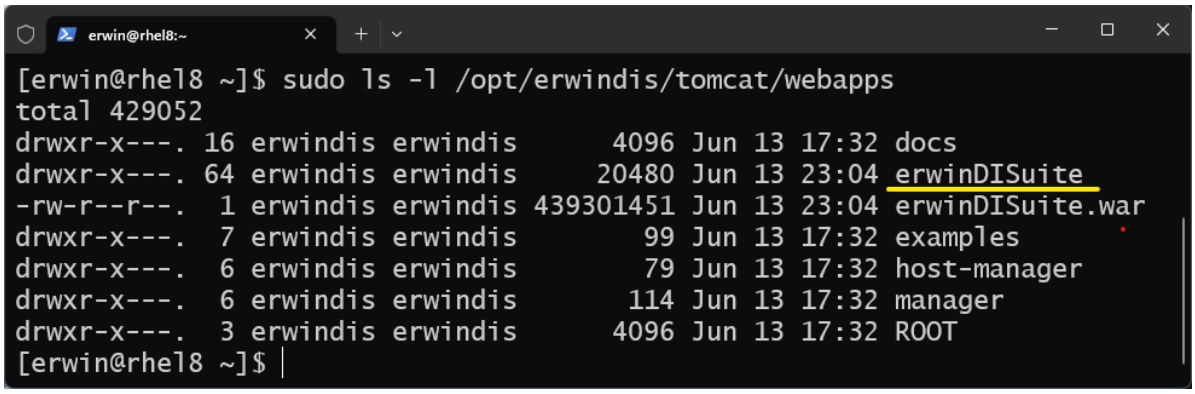
Note: The "\" indicates a multiline shell command. Be sure to copy/paste the entire text block to your shell.

```
sudo -u QuestDI wget -P /opt/QuestDI/Downloads \
https://erwin-us.s3.amazonaws.com/Support/ADS/v16.0/erwinDI_v16.0.zip \
&& sudo -u QuestDI unzip /opt/QuestDI/Downloads/erwinDI_v16.0.zip \
-d /opt/QuestDI/Downloads/erwinDI_v16.0 \
&& sudo -u QuestDI cp /opt/QuestDI/Downloads/erwinDI_v16.0/WAR/QuestDI.war \
/opt/QuestDI/tomcat/webapps/QuestDI.war
```

1.5.1.1 - Verify the QuestDI.war has expanded to create its project folder.

```
sudo ls -l /opt/QuestDI/tomcat/webapps
```

Your result should be like this example directory listing showing the QuestDI project folder has expanded from QuestDI.war after being deployed by Tomcat.



1.5.2 - Stop the tomcat service: QuestDI.

We can shut down Tomcat for now. The next few steps will walk you through creating and configuring the application database, and edits to the Quest Data Intelligence application properties files.

```
sudo systemctl stop QuestDI
```

1.6 - Create and configure the database for Quest Data Intelligence (SQL Server or Oracle).

Create the Quest DI Database in SQL SERVER

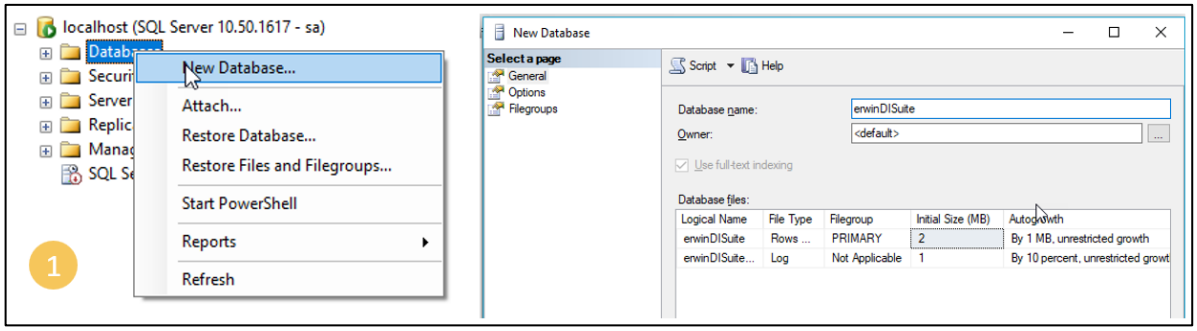
The following steps are for a **SQL SERVER** database.

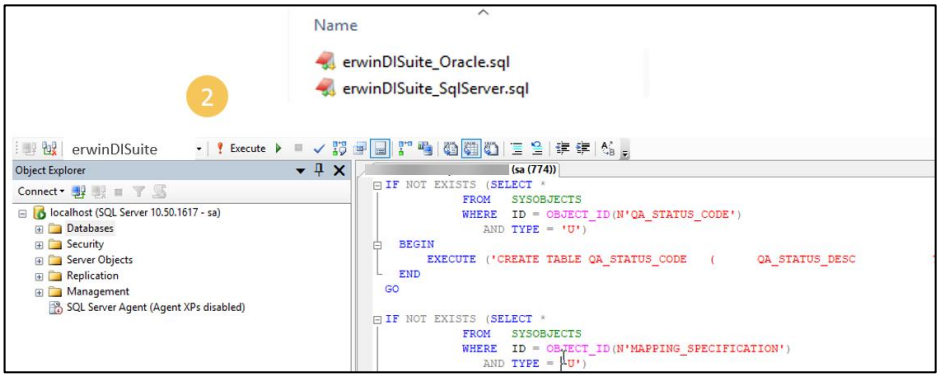
1. Create a new Database/Schema name for Quest DI e.g. "**QuestDI**".
2. From the SQL folder of the installation software, run the "**QuestDataIntelligence_SqlServer.sql**" file against the newly created SQL Server Database.
3. In the SQL folder, you will also see a file **QuestDataIntelligence_SqlServer_Prerequisite.sql**. Please note that this file is not required for a new install and is only required in the case you are upgrading from an older version to the 11.1 version. In the case of upgrade, please refer to the upgrade guide.
4. The required database tables for the software are created in the SQL Server database.

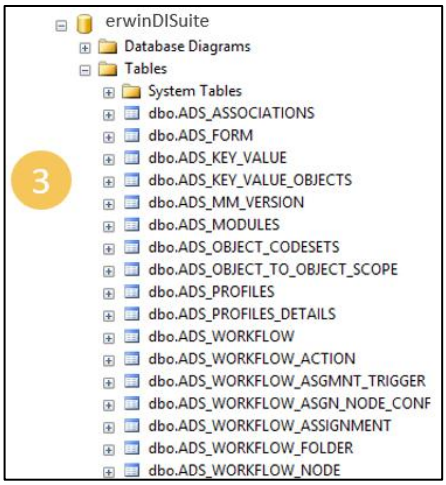
****IMPORTANT NOTE****:

A **dedicated database** needs to be created in SQL Server for the software and the DDL needs to be executed against this dedicated database.

The DDL should not be executed against the MASTER schema.



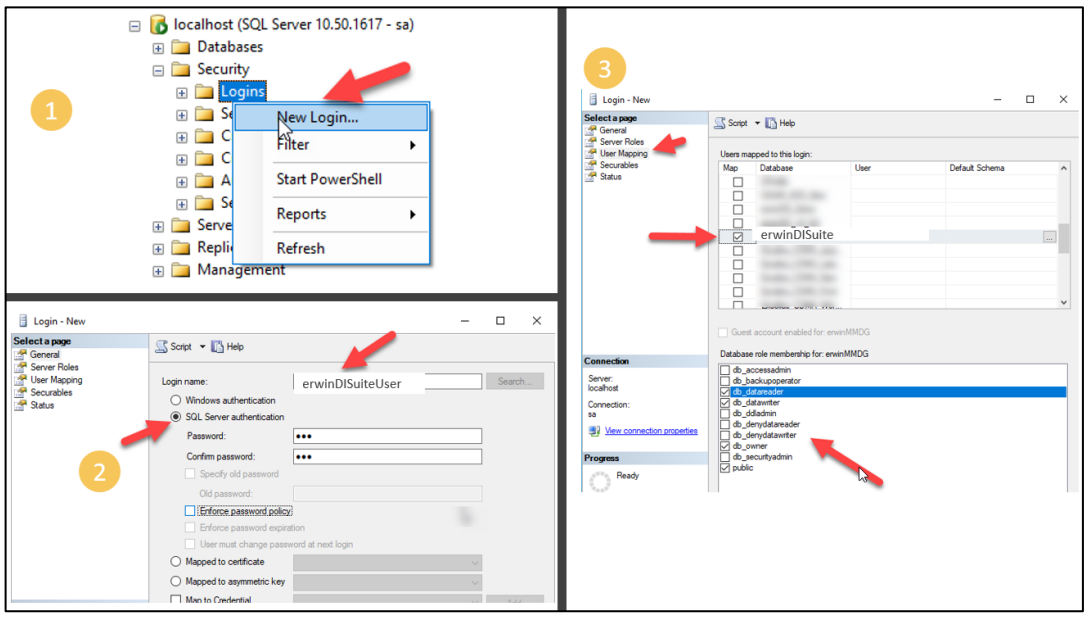




Create a dedicated DB User Account for the Quest DI database

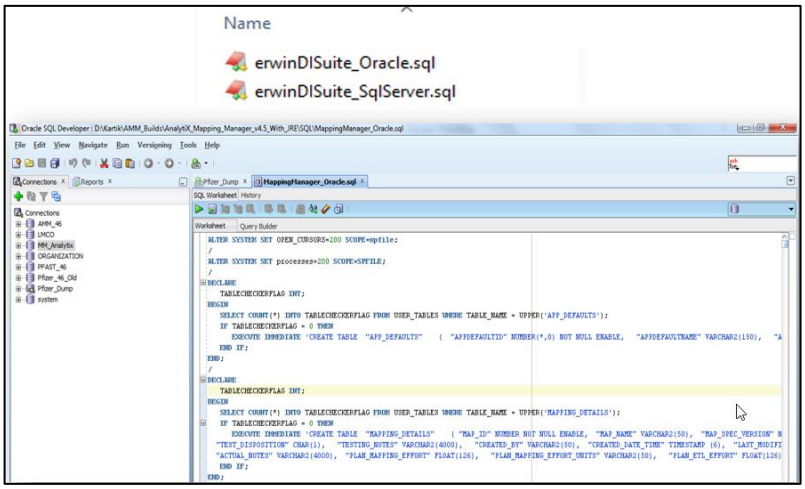
1. Create a new Database login role for the QuestDI Database (e.g. create a new DB role as "**QuestDIUser**" for the previously created database "QuestDI".
2. Both Windows Authentication and SQL Server Authentication modes are supported for a SQL Server database.
    a. If you select the "**SQL Server Authentication**" mode for the new login role, remember to uncheck the enforce password policy option and change the password on first login. Configure the relevant parameters in the *database.properties* file.
    b. For Windows Authentication mode support, the database configuration file *database.properties* has a section dedicated to entering the connectivity parameters that support windows authentication.
3. Grant the new login the following roles.

    - Public, db_owner, data_reader, data_writer



Create a Schema in Oracle Database

1. Create a new Database/Schema name for the DI application in the Oracle database e.g. "**QuestDI**"
2. Provide the following privileges to the "**QuestDI**" user/schema
    - Resource
    - Connect
    - Create a View privileges
    - GRANT UNLIMITED TABLESPACE
3. From the SQL folder of the installation software, run the "**QuestDataIntelligence_Oracle.sql**" file against the newly created Oracle Schema
4. The required database tables for the software are created in the Oracle schema.



IMPORTANT NOTE:

A dedicated schema name needs to be created in Oracle for the DI Suite and the DDL needs to be executed against this dedicated schema.

The DDL should not be executed against SYS or SYSTEM schemas.

1.7 - Configuring the 'database.properties' file

1.7.1 - Configuring the 'database.properties' file for MS SQL Server database

Return to your RHEL server and open the database.properties file with the editor of your choice.
In this example we will edit the file with the nano editor using elevated privileges via sudo.

```
sudo nano /opt/QuestDI/tomcat/webapps/QuestDI/WEB-INF/database/database.properties
```

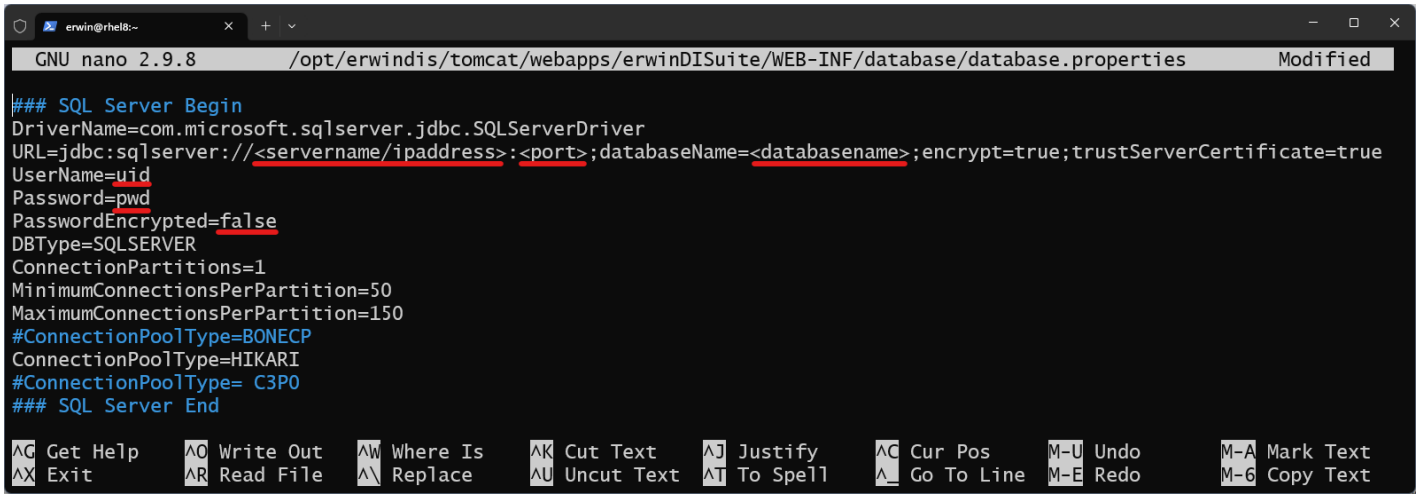
1.7.1.1 - SQL Authentication Mode

Uncomment the SQL Server section by removing the # at the beginning of each line (between SQL SERVER BEGIN and SQL SERVER END section)

Enter the following parameters

- SERVER NAME
- PORT# (default 1433)
- Database Name
- User Name
- Password
- PasswordEncrypted = false

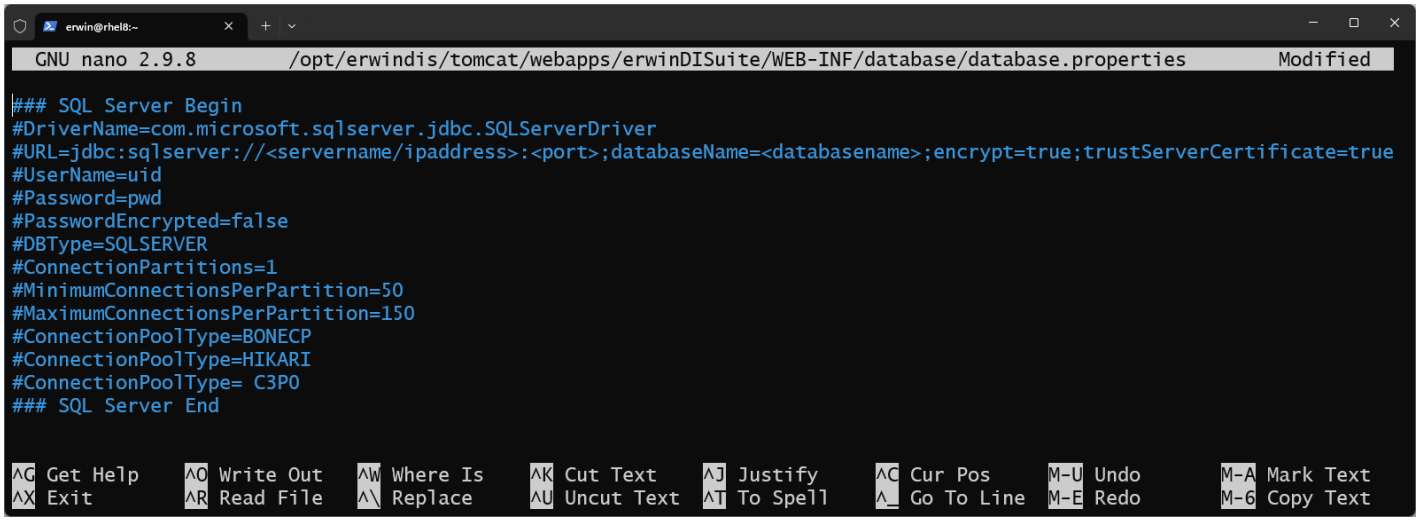
The parameters you will need to edit are indicated below in red.



Once your edits are complete. Save the file and skip to the step: **Configuring the application documents repository path.**

1.7.1.2 - Windows Authentication Mode

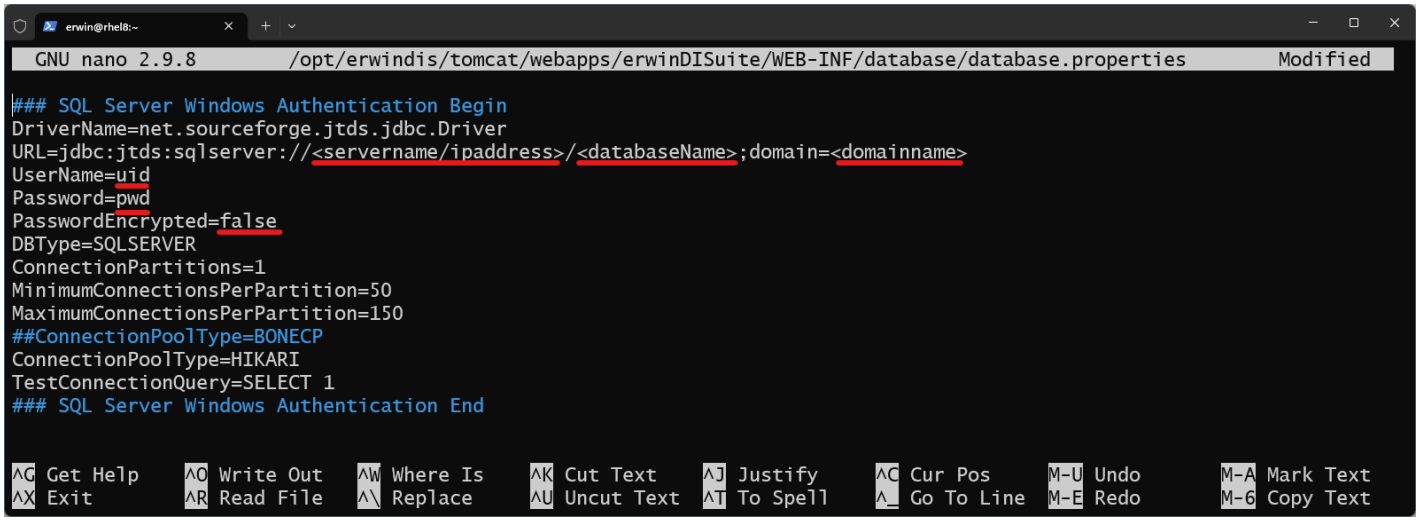
If you require using Windows Authentication mode, you will first need to comment out all rows between **SQL Server Begin** and **SQL Server End** section as shown in the example screen below.



Then scroll down to the **SQL Server Windows Authentication** section and uncomment the section by removing the # at the beginning of each line between **SQL SERVER Windows Authentication BEGIN** and **SQL SERVER Windows Authentication END** section as shown in the example below. Then enter the following parameters for your SQL Server environment.

- SERVER NAME or IP Address
- Database Name
- Domain
- UserName (this is still required for windows authentication)
- Password (this is still required for windows authentication)
- PasswordEncrypted = false

The parameters you will need to edit are indicated below in red.



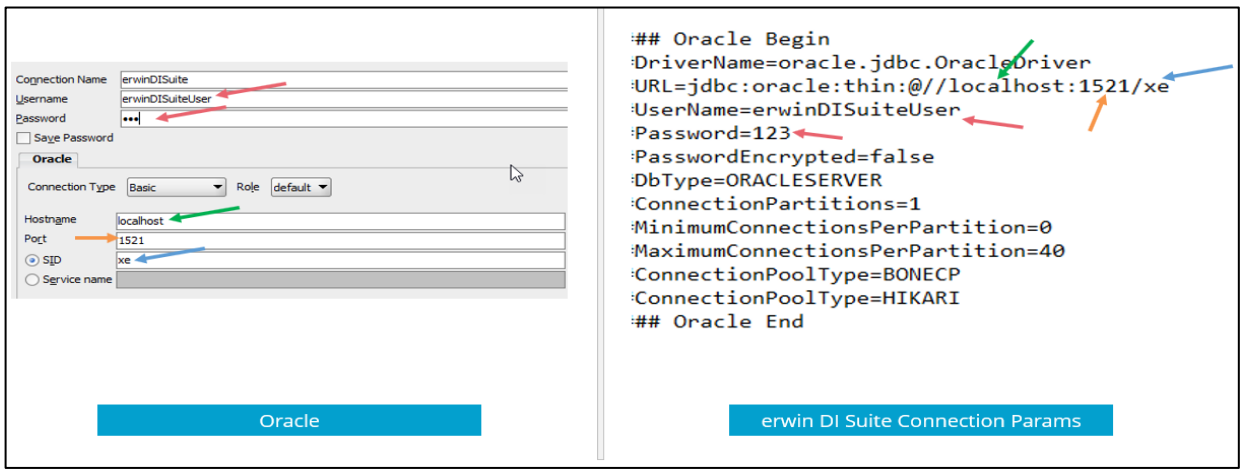
Potential additional parameters to the URL, ;useNTLMv2=true and/or ;ssl=required, if unable to connect.

1.7.2 - Configuring the “database.properties” file for Oracle Database

Uncomment the ORACLE section by removing the # at the beginning of each line (between ORACLE BEGIN and ORACLE END section)

Enter the following parameters

- SERVER NAME
- PORT# (default 1521)
- Database Name
- User Name
- Password
- PasswordEncrypted = false
- Sample screenshot of Parameters is shown below.



1.8 - Configuring the application documents repository path

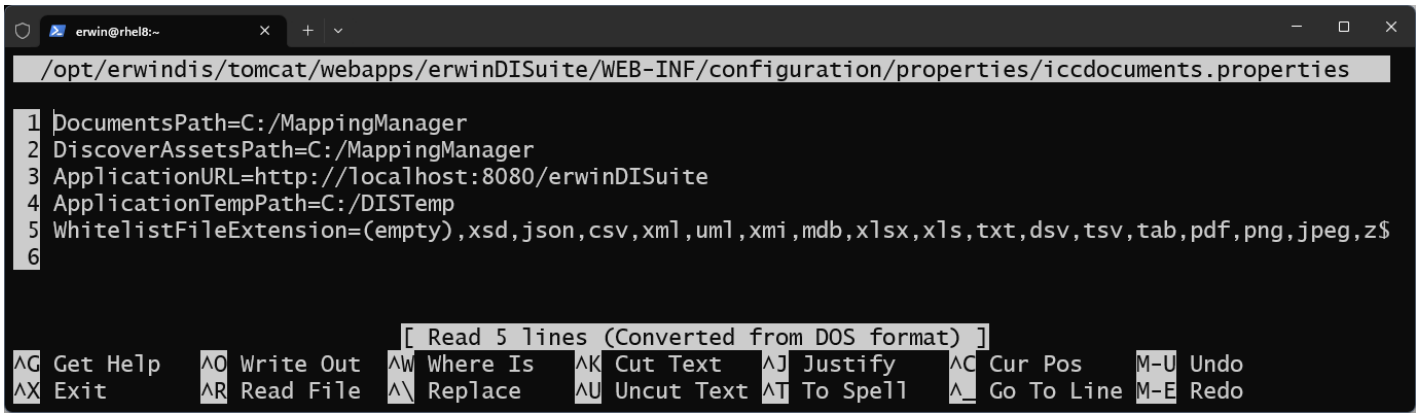
1.8.1 - Update the 'iccdocuments.properties' file.

Return to your RHEL server and open the 'iccdocuments.properties file with the editor of your choice.
In this example we will edit the file with the nano editor using elevated privileges via sudo.

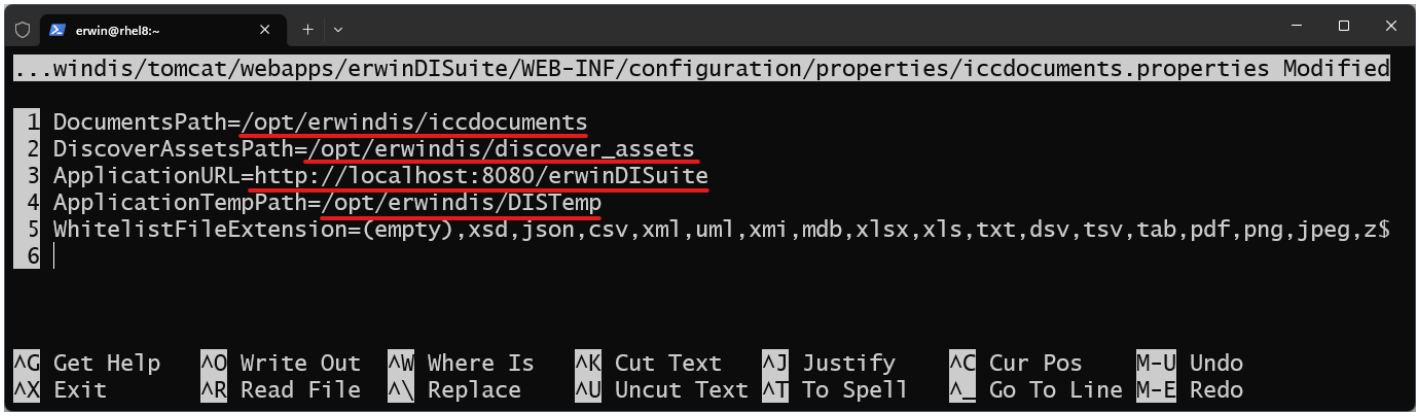
```
sudo nano -l /opt/QuestDI/tomcat/webapps/QuestDI/WEB-INF/configuration/properties/iccdocuments.properties
```

Note: The -l switch will cause the nano editor to display line numbers.

Once the file is open, notice that lines 1, 2, and 4 contain Windows filesystem paths.
We will need to edit these lines to point to the correct Linux filesystem paths.



Edit the lines to match the correct paths for your Linux environment. In the example below, we are using the paths previously created when we created additional folders in the QuestDI home `/opt/QuestDI` folder.



1.8.1.1 - Explanation of Paths

- **DocumentsPath**=Client_Folder_Path **E.g.** DocumentsPath=/opt/QuestDI/iccdocuments
  - This directory stores all the documents that are uploaded via the product as part of the rich media library and other attachment functionalities and this folder is used for backup and restoration purposes in case a shift to another server is required in the future.
- **ApplicationURL**= Client_DI_App_URL **E.g.** ApplicationURL= http:/Questserver:8080/QuestDI
  - This is the base Quest DI application URL that is used as part of the capabilities that generate Unique URLs to assets.
- **ApplicationTempPath**=Provide_New_Path_For_Temp_Files **E.g.** ApplicationTempPath= /opt/QuestDI/DISTemp
  - This path is used to store all the temp files uploaded into the application (typically outside the tomcat directory) and can be periodically deleted by an admin.
- **DiscoverAssetsPath**=Provide_Folder_Path **E.g.** DocumentsPath= /opt/QuestDI/discover_assets
  - This is typically the same path as the *ApplicationTempPath* variable but can be changed if needed. This path is used to sync the assets with the Discover Assets module and will be used going forward to support multiple instances.

Important: Restart the Tomcat server once the Application and Database Properties file have been updated for these changes to come into effect.

Start the tomcat server using the QuestDI.service unit file.

```
sudo systemctl start QuestDI.service
```

1.8.2 - Check status to view the result.

Verify tomcat is running using the QuestDI.service unit file.

```
sudo systemctl status QuestDI.service
```

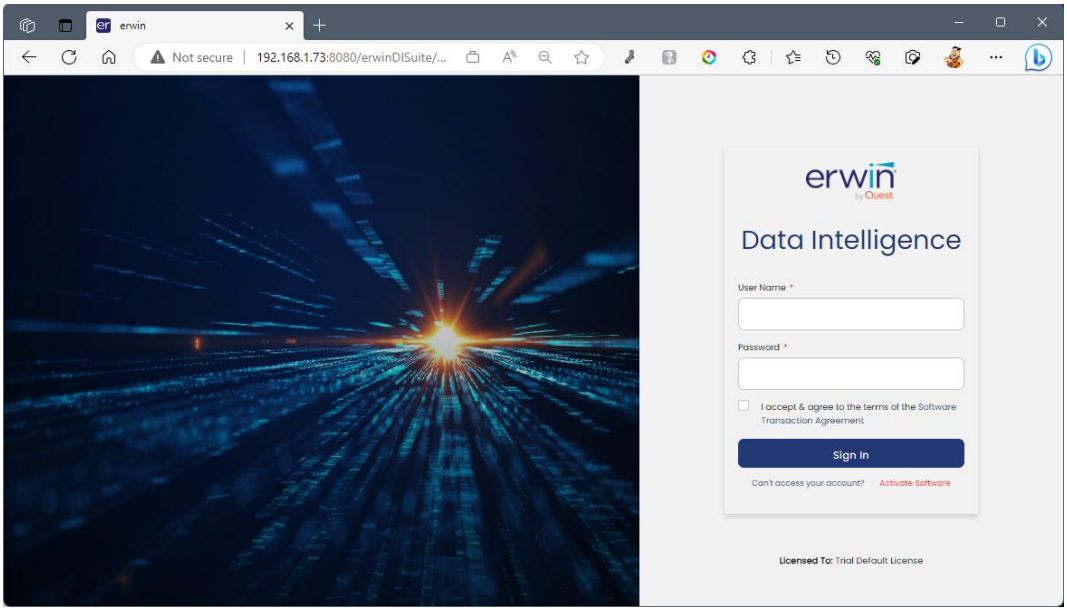
1.9 - Access the Quest Data Intelligence Login Screen

Now it is time to test your work. If you have faithfully completed all the steps outlined in this chapter, you should now be able to access the Quest Data Intelligence Login screen from your local browser.

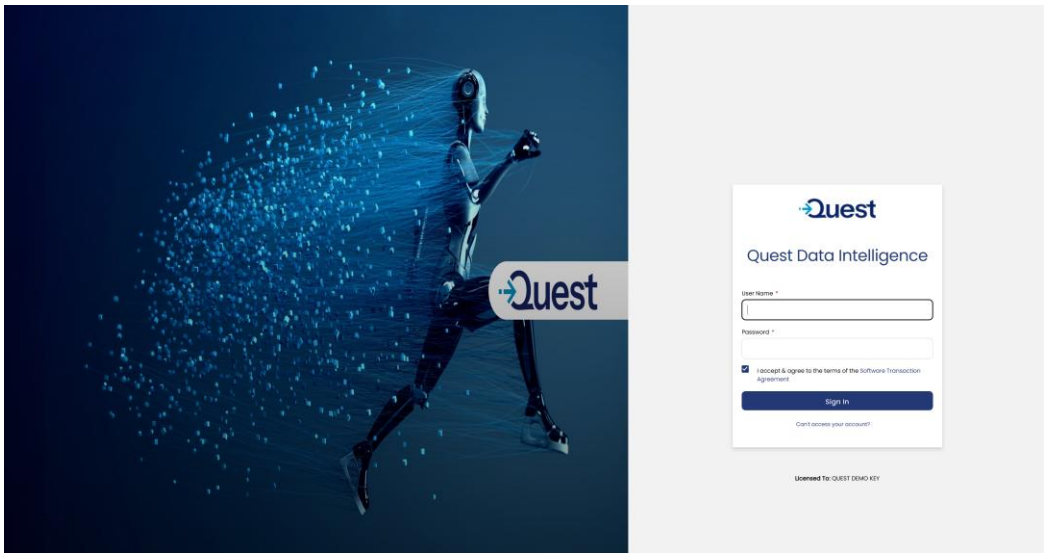
1.9.1 - Open a web browser and type the URL for your server.

The URL format is **http://<ip-address/hostname>:8080/QuestDI**

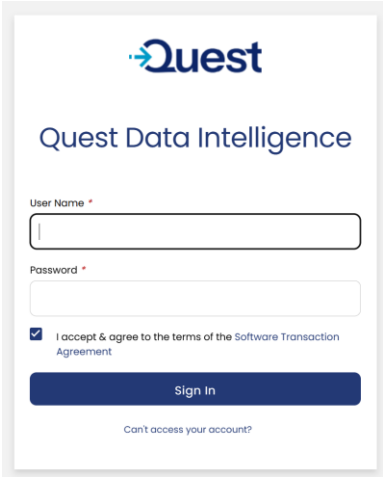
- Replace ***<ip-address/hostname>*** with your correct IP address or DNS hostname.
- Port '***8080***' is the default port used earlier when configuring Tomcat. If you changed the port used during Tomcat configuration, remember to use the correct port for your environment.
- URI path '***/QuestDI***' is the default application name and URI path.
  If you deployed the war file with a different name, use the correct /path for your environment and replace the ***QuestDI*** variable accordingly e.g. ***erwinDataIntelligence, erwinDI***
- If all steps were completed accurately, and the correct URL is used your browser should display the Quest Data Intelligence login screen along with an option to apply your license key.



- Use the license key that has been provided to you to Activate the software (*How to activate the software* section below)
- You should now see the login screen with the license key successfully applied.



- Use your Administrator *User ID* and *Password* credentials to login to the application for the first time and configure all other aspects.



- We recommend that you change the default Administrator password upon login for security purposes.

1.9.2 - Activate the software.

A license key is required to activate the software. Login to the application will not work until the license key has been applied. If you do not have a license key, please reach out to your Quest\erwin sales contact to obtain your license key.

Once the license key has been obtained, return to the login screen, and click the 'Activate Software' link under the Sign in button.

2 - Ubuntu Server, and Debian

2.1 - Update Linux OS repos, patches, security updates and other packages.

```
sudo apt update ; sudo apt upgrade -y ; sudo apt install -y apt-transport-https zip
```

2.2 - Create Linux user 'QuestDI' for tomcat service account.

As a security best practice, Apache Tomcat should never be run under "root" user account.
We recommend creating a Linux system account user that is restricted from shell login.

With security best practices in mind, using the commands below we will create a system account user named: '**QuestDI**' with home folder path **`/opt/QuestDI`** and the user is restricted with no shell access.

```
sudo useradd -r -c "QuestDI service" -m -d /opt/QuestDI -s /usr/sbin/nologin -U QuestDI
```

2.2.1 - Set up additional folders in the QuestDI home folder.

The following will create some folders in `/opt/QuestDI` used by Quest Data Intelligence. The folders will be owned by the user and group 'QuestDI' which is the service account used by Tomcat.

Note: the '\' indicates a multiline shell command. Be sure to copy/paste the entire text block to your shell.

```
sudo -u QuestDI mkdir -p \
/opt/QuestDI/Downloads \
/opt/QuestDI/DISTemp \
/opt/QuestDI/discover_assets \
/opt/QuestDI/iccdocuments
```

2.3 - Install certified Adoptium OpenJDK from Eclipse Foundation

Quest\erwin recommends installing the certified version of JDK 17 using the package manager for your chosen Linux distribution. The instructions below provide the steps to install Adoptium JDK using Linux (RPM/DEB) installer packages.

The Adoptium JDK 17 installer updates the Linux 'alternatives' system to set the default java. We do not need to set the JAVA_HOME environment variable as we will configure that in a later step for the tomcat systemd unit file.

For additional information about installing Adoptium OpenJDK please refer to the official documentation from the Adoptium site at the following URL:
https://adoptium.net/installation/linux/

2.3.1 - Download the Eclipse Adoptium GPG Key

Note: Be sure to copy/paste the entire text block to your shell.

```
sudo mkdir -p /etc/apt/keyrings
sudo wget -O - https://packages.adoptium.net/artifactory/api/gpg/key/public | sudo tee \
/etc/apt/keyrings/adoptium.asc
```

2.3.2 - Add the Adoptium repository to your Linux distribution.

Note: Be sure to copy/paste the entire text block to your shell.

```
echo "deb [signed-by=/etc/apt/keyrings/adoptium.asc] \
https://packages.adoptium.net/artifactory/deb $(awk -F= '/^VERSION_CODENAME/{print$2}' \
/etc/os-release) main" | sudo tee /etc/apt/sources.list.d/adoptium.list
```

2.3.3 - Update the repository cache.

```
sudo apt update
```

2.3.4 - Install the specific version of Adoptium JDK certified for Quest Data Intelligence

```
sudo apt install -y temurin-17-jdk=17.0.12.0.0+7 -V
```

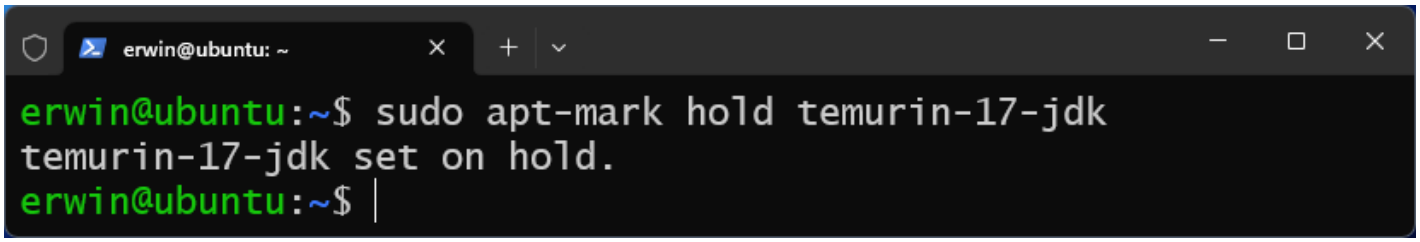
2.3.5 - Disable the Adoptium repository.

To prevent unintended future JDK 17 version updates, Quest\erwin recommends disabling the Adoptium repository to prevent unintended installation of future JDK 17 version updates.

This step is optional but recommended.
Installing uncertified versions of JDK 17 may cause unexpected results.
Disabling the repository will avoid potential problems.

```
sudo apt-mark hold temurin-17-jdk
```

The output should show 'temurin-17-jdk set on hold' indicating the repository is on hold. Preventing unintended updates.



2.3.5.1 - Enable the repository again if needed for update.

If you require installing a newer JDK from the Adoptium repository, you can enable the repo again with the following command:

```
sudo apt-mark unhold temurin-17-jdk
```

2.4 - Download and Install Apache Tomcat v10.1.x

Quest\erwin QA has certified specifically Apache Tomcat versions 10.1.x for use with the
Quest Data Intelligence v16.0 release. It is strongly recommended to install only a certified version of Apache Tomcat.

2.4.1 - Download the Apache Tomcat tar to `/opt/QuestDI/Downloads` directory.

Note: the '\' indicates a multiline shell command. Be sure to copy/paste the entire text block to your shell.

```
sudo -u QuestDI wget -P /opt/QuestDI/Downloads \
https://archive.apache.org/dist/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
```

2.4.1.1 - Unpack the tar to QuestDI home directory.

```
sudo -u QuestDI tar -xvf /opt/QuestDI/Downloads/apache-tomcat-10.1.28.tar.gz \
-C /opt/QuestDI && sudo rm /opt/QuestDI/Downloads/apache-tomcat-10.1.28.tar.gz
```

2.4.1.2 - Create a symbolic link for the apache-tomcat-10.1.x path.

Creating a symbolic link will make it easier to reference the tomcat installation path, and to manage future upgraded versions.

```
sudo -u QuestDI ln -s /opt/QuestDI/apache-tomcat-10.1.28 /opt/QuestDI/tomcat
```

2.4.1.3 - Verify the contents of the QuestDI home directory to see the results.

```
sudo ls -l /opt/QuestDI/
```

Your output should now match the following example (Tomcat version may vary):

```
erwin@ubuntu:~$ sudo ls -l /opt/erwindis/
total 20
drwxrwxr-x 9 erwindis erwindis 4096 Jun 17 20:51 apache-tomcat-10.1.17
drwxrwxr-x 2 erwindis erwindis 4096 Jun 17 20:16 discover_assets
drwxrwxr-x 2 erwindis erwindis 4096 Jun 17 20:16 DISTemp
drwxrwxr-x 2 erwindis erwindis 4096 Jun 17 20:51 Downloads
drwxrwxr-x 2 erwindis erwindis 4096 Jun 17 20:16 iccdocuments
lrwxrwxrwx 1 erwindis erwindis   34 Jun 17 20:52 tomcat -> /opt/erwindis/ apache-tomcat-10.1.17
erwin@ubuntu:~$
```

2.4.2 - Configure a systemd service unit file to manage starting/stopping tomcat as a service.

We recommend managing the start/stop/restart of apache tomcat service with the built in Linux SYSTEMD system. We use the 'tee' command below to create a new **systemd** service unit file named **QuestDI.service** located in `/etc/systemd/system`.

Caution!

The example below assumes Tomcat version 10.1.28, sets the JAVA_HOME path, and assumes the tomcat install path to be `/opt/QuestDI/tomcat` and will allocate **24 GiB RAM** for the Java Virtual Machine (JVM).

Review the paths and the settings below, and modify if needed, to set correct values for your environment. It is recommended to set the -Xms and -Xmx settings to 75% of the server's memory.

-Xms and -Xmx should be set to equal RAM values or -Xms is 50% of -Xmx for efficient java garbage collection.

2.4.2.1 - Create the /etc/systemd/system/QuestDI.service file.

Copy/Paste the entire text block to your shell to create the service unit file using the 'tee' command.

Note: Be sure to copy/paste the entire text block to your shell.

```
sudo tee -a /etc/systemd/system/QuestDI.service >/dev/null << EOF
[Unit]
Description=Tomcat 10.1.28 servlet container
Documentation=https://support.quest.com/erwin-data-intelligence-suite/
Wants=network.target
After=syslog.target network.target

[Service]
Type=forking
Restart=always
RestartSec=5
TimeoutStartSec=300

# Tomcat should never be running as 'root'
# Here we specify a user account created as a service account for tomcat.
# Be sure the user and group with correct permissions has been created.
User=QuestDI
Group=QuestDI

# Port 443 is a privileged port on Linux OS
# To allow tomcat to bind to ports below 1024 without running it as root
# Uncomment the next line to allow 443 instead of 8443 for ssl connector
#AmbientCapabilities=CAP_NET_BIND_SERVICE

# Set JAVA_HOME path for Adoptium temurin-17-jdk
Environment="JAVA_HOME=/usr/lib/jvm/temurin-17-jdk-amd64/"

# Setting the java.security.egd system property to use/dev/urandom configures
# the Tomcat JVM to use /dev/urandom as the source of entropy for the SecureRandom
# class to avoid potential delays during startup.
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom"

# Set CATALINA environment variables
Environment="CATALINA_BASE=/opt/QuestDI/tomcat"
Environment="CATALINA_HOME=/opt/QuestDI/tomcat"
Environment="CATALINA_PID=/opt/QuestDI/tomcat/temp/tomcat.pid"
# Set the JVM memory allocation.
# As a best practice for efficient garbage collection it is recommended to
# set -Xms value equal to -Xmx, or set -Xms at 50% of -Xmx
# Quest Data Intelligence requires 16G as a minimum.
# Each user session requires 0.5 GB of RAM, so if 48 concurrent users are
# expected to login to the application, the recommended JVM memory allocation
# would be: -Xms=24G -Xmx=24G or -Xms-12G -Xmx=24G.
Environment="CATALINA_OPTS=-Xms24G -Xmx24G -server -XX:+UseParallelGC"

# Setting the java.security.egd system property to use/dev/urandom configures
# the Tomcat JVM to use /dev/urandom as the source of entropy for the SecureRandom
# class to avoid potential delays during startup.
# QuestDI v13.x and 14.x require additional Java parameter options to be set
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom --add-opens=java.base/java.lang=ALL-
UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --
add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-opens=java.base/java.net=ALL-UNNAMED --add-
opens=java.management/sun.management=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-
opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --add-
opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.util.regex=ALL-UNNAMED --
add-opens=java.base/java.net=ALL-UNNAMED --add-exports java.base/sun.nio.ch=ALL-UNNAMED  --add-
opens=java.base/java.util.concurrent=ALL-UNNAMED”

# Paths to Start/Stop scripts
ExecStart=/opt/QuestDI/tomcat/bin/startup.sh
ExecStop=/opt/QuestDI/tomcat/bin/shutdown.sh

# The SuccessExitStatus=143 directive tells systemd to consider the exit code 143 (which is what
```

```
# Java returns when it is terminated by a SIGTERM signal) as a successful termination.
# This can help ensure that systemd allows Quest DIS Lineage Graph Export to complete before
# the java process is terminated
SuccessExitStatus=143
TimeoutStopSec=60

[Install]
WantedBy=multi-user.target
EOF
```

Reload systemd daemons.

```
sudo systemctl daemon-reload
```

2.4.2.2 - Enable the QuestDI.service file to start tomcat during server reboots.

```
sudo systemctl enable QuestDI.service
```

2.4.2.3 - Verify the unit file is enabled.

```
sudo systemctl status QuestDI.service
```

2.4.2.4 - Start the tomcat server using the QuestDI.service unit file with the following command.

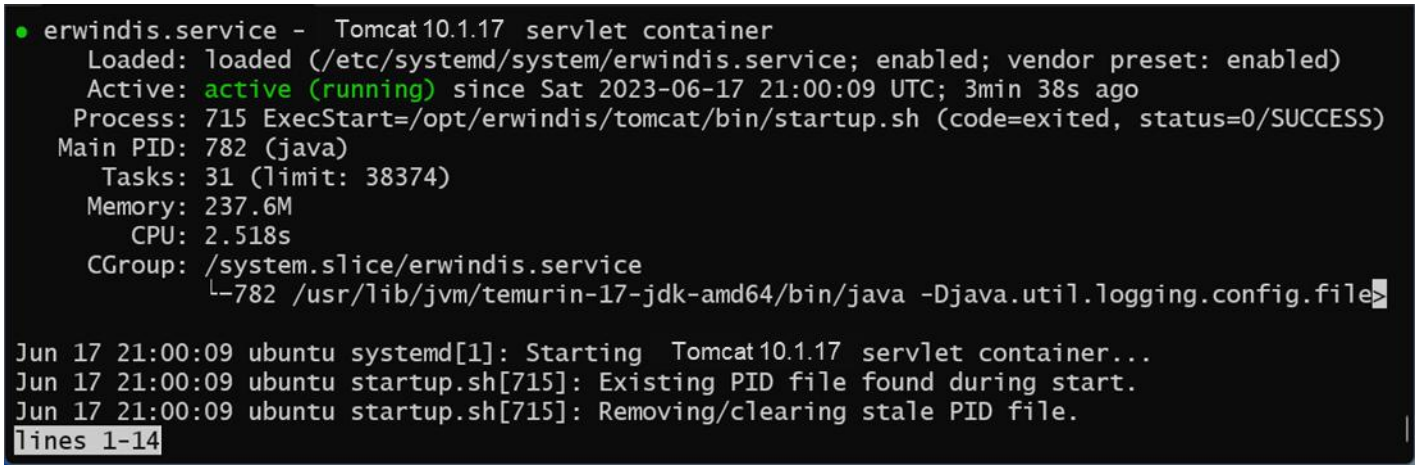
```
sudo systemctl start QuestDI.service
```

2.4.2.5 - Check status to view the result.

Verify tomcat is running using the QuestDI.service unit file.

```
sudo systemctl status QuestDI.service
```

Your result should be like:



If you see Active: active (running) we are good to go. Leave tomcat running as we will soon need it when deploying the Quest Data Intelligence webapp WAR file.

2.4.3 - Optional: configure the Ubuntu firewall

This section covers opening tcp ports on the default 'ufw' firewall included with Ubuntu. 'ufw' is disabled by default. If you wish to enable ufw, the instructions below open port 8080/tcp for Tomcat, and port 22/tpc for OpenSSH. If you are employing a different type of firewall, please refer to your firewall's instructions.

2.4.3.1 - Open Tomcat port 8080/tcp, and ssh port 22/tcp

Note: A rule allowing OpenSSH (port 22/tpc) is added to avoid interrupting existing ssh connections and losing access to the server.
Verify the ports you need to allow before proceeding. Verify rules with your security or firewall administrator to ensure rules created are in compliance with your organization's security policies.

```
sudo ufw allow OpenSSH; sudo ufw allow 8080/tcp; sudo ufw enable; sudo ufw reload
```

2.5 - Download and Deploy Quest Data Intelligence v16.0

Customers should download the latest v16.0 release zip file from http://support.quest.com to ensure they are obtaining the latest GA release.

2.5.1 - Download the erwin_Data_Intelligence_v16.0 zip file.

The following command block downloads the zip file from the https://support.quest.com site. Then unzips the file and copies the QuestDI.war file to the `/opt/QuestDI/tomcat/webapps/` directory.

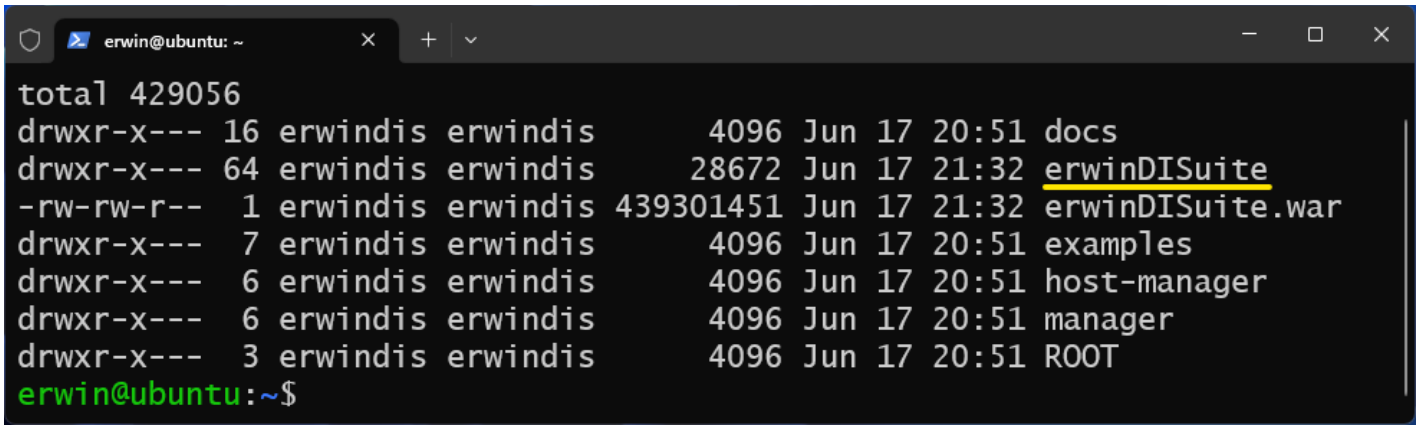
Note: The "\" indicates a multiline shell command. Be sure to copy/paste the entire text block to your shell.

```
sudo -u QuestDI wget -P /opt/QuestDI/Downloads \
https://erwin-us.s3.amazonaws.com/Support/ADS/v16.0/erwinDI_v16.0.zip \
&& sudo -u QuestDI unzip /opt/QuestDI/Downloads/erwinDI_v16.0.zip \
-d /opt/QuestDI/Downloads/QuestDI_v16.0 \
&& sudo -u QuestDI cp /opt/QuestDI/Downloads/erwinDI_v16.0/WAR/QuestDI.war \
/opt/QuestDI/tomcat/webapps/QuestDI.war
```

2.5.1.1 - Verify the QuestDI.war has expanded to create its project folder.

```
sudo ls -l /opt/QuestDI/tomcat/webapps
```

Your result should be like this example directory listing showing the QuestDI project folder has expanded from QuestDI.war after being deployed by Tomcat.



2.5.2 - Stop the tomcat service: QuestDI.

We can shut down Tomcat for now. The next few steps will walk you through creating and configuring the application database, and edits to the Quest Data Intelligence application properties files.

```
sudo systemctl stop QuestDI
```

2.6 - Create and configure the database for Quest Data Intelligence (SQL Server or Oracle).

Create the Quest DI Database in SQL SERVER

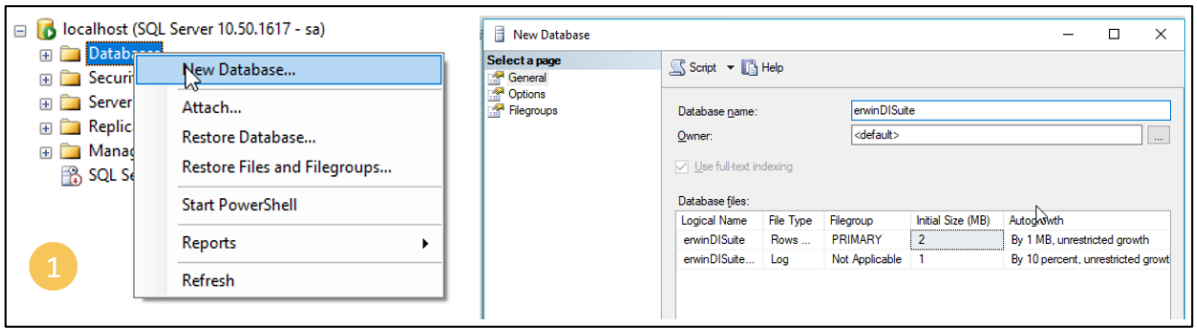
The following steps are for a **SQL SERVER** database.

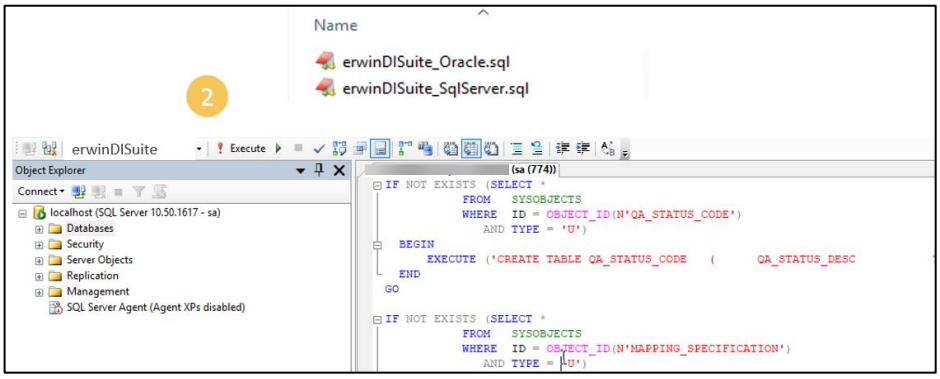
1. Create a new Database/Schema name for Quest DI e.g. "**QuestDI**".
2. From the SQL folder of the installation software, run the "**QuestDI_SqlServer.sql**" file against the newly created SQL Server Database.
3. In the SQL folder, you will also see a file **QuestDI_SqlServer_Prerequisite.sql**. Please note that this file is not required for a new install and is only required in the case you are upgrading from an older version to the 11.1 version. In the case of upgrade, please refer to the upgrade guide.
4. The required database tables for the software are created in the SQL Server database.

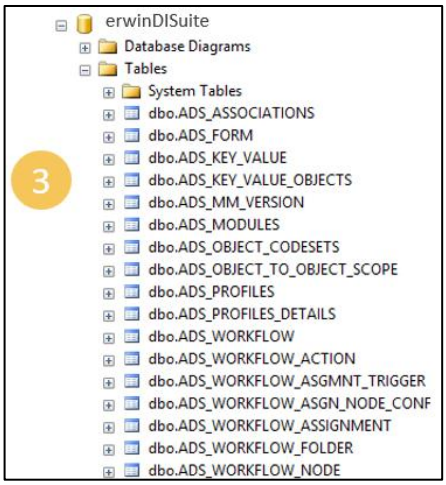
****IMPORTANT NOTE****:

A **dedicated database** needs to be created in SQL Server for the software and the DDL needs to be executed against this dedicated database.

The DDL should not be executed against the MASTER schema.



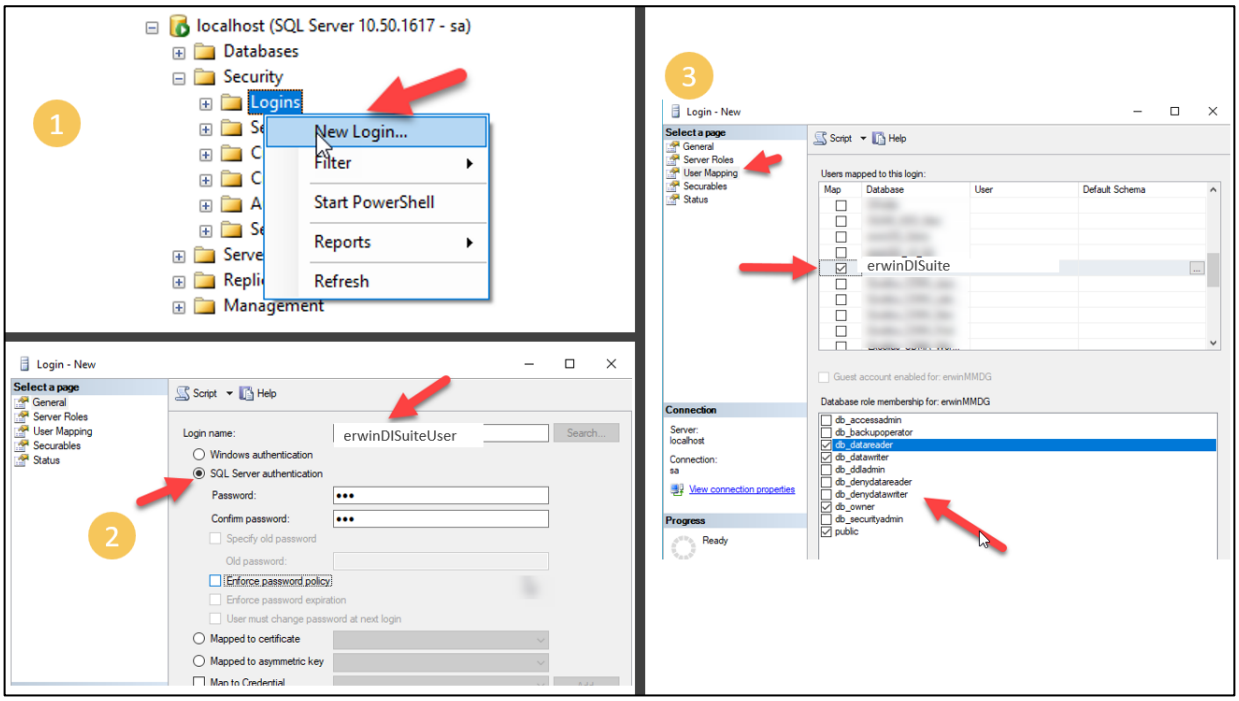




Create a dedicated DB User Account for the Quest DI database

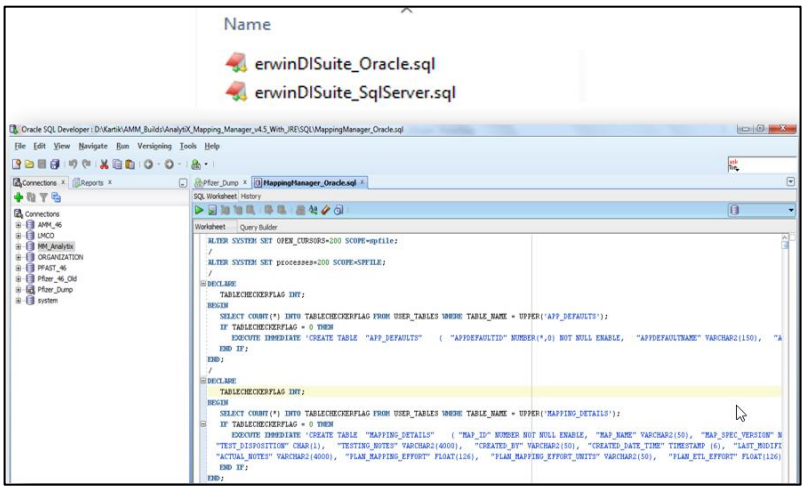
1. Create a new Database login role for the QuestDI Database (e.g. create a new DB role as "**QuestDIUser**" for the previously created database "QuestDI".
2. Both Windows Authentication and SQL Server Authentication modes are supported for a SQL Server database.
    a. If you select the "**SQL Server Authentication**" mode for the new login role, remember to uncheck the enforce password policy option and change the password on first login. Configure the relevant parameters in the *database.properties* file.
    b. For Windows Authentication mode support, the database configuration file *database.properties* has a section dedicated to entering the connectivity parameters that support windows authentication.
3. Grant the new login the following roles.

    - Public, db_owner, data_reader, data_writer



Create a Schema in Oracle Database

1. Create a new Database/Schema name for the DI application in the Oracle database e.g. "**QuestDI**"
2. Provide the following privileges to the "**QuestDI**" user/schema
    - Resource
    - Connect
    - Create a View privileges
    - GRANT UNLIMITED TABLESPACE
3. From the SQL folder of the installation software, run the "**QuestDI_Oracle.sql**" file against the newly created Oracle Schema
4. The required database tables for the software are created in the Oracle schema.



IMPORTANT NOTE:

A dedicated schema name needs to be created in Oracle for the DI Suite and the DDL needs to be executed against this dedicated schema.

The DDL should not be executed against SYS or SYSTEM schemas.

2.7 - Configuring the 'database.properties' file

2.7.1 - Configuring the 'database.properties' file for MS SQL Server database

Return to your UBUNTU server and open the database.properties file with the editor of your choice.
In this example we will edit the file with the nano editor using elevated privileges via sudo.

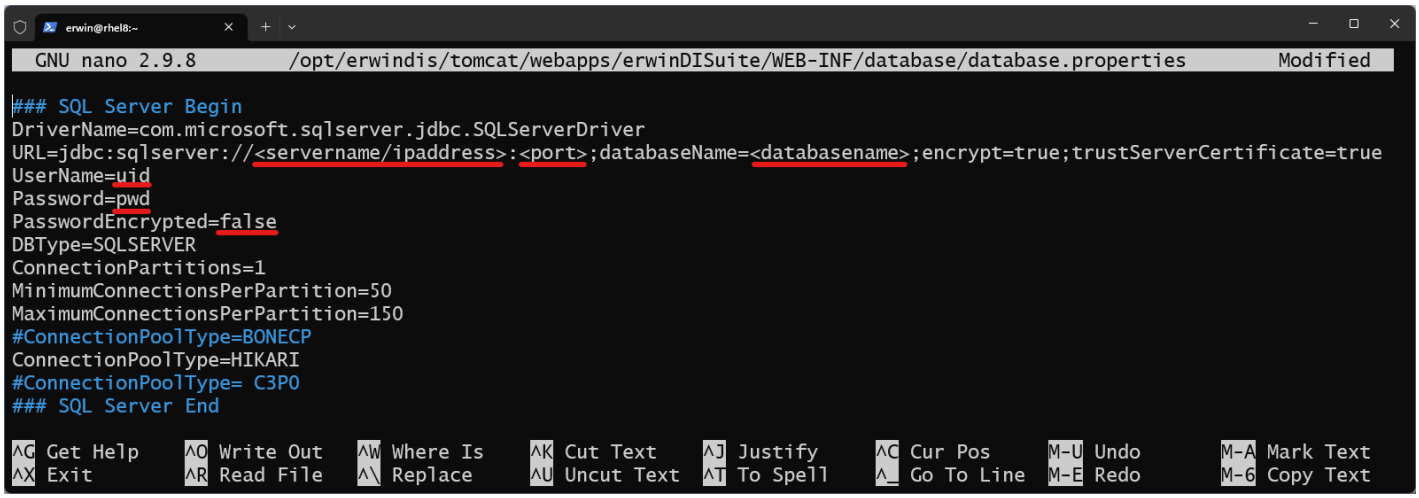
```
sudo nano /opt/QuestDI/tomcat/webapps/QuestDI/WEB-INF/database/database.properties
```

2.7.1.1 - SQL Authentication Mode

If necessary, uncomment the SQL Server section by removing the # at the beginning of each line (between SQL SERVER BEGIN and SQL SERVER END section) and enter the following parameters for your SQL Server environment.

- Server Name
- Port # (default 1433)
- Database Name
- UserName
- Password
- PasswordEncrypted = false

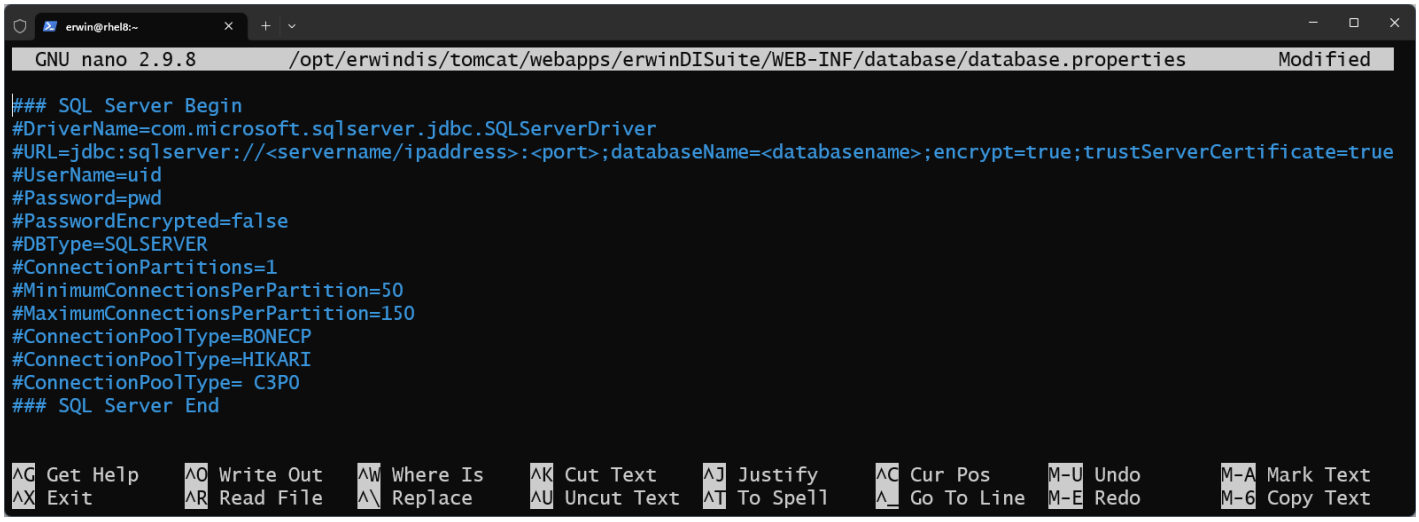
The parameters you will need to edit are indicated below in red.



Once your edits are complete. Save the file and skip to the step: **Configuring the application documents repository path.**

2.7.1.2 - Windows Authentication Mode

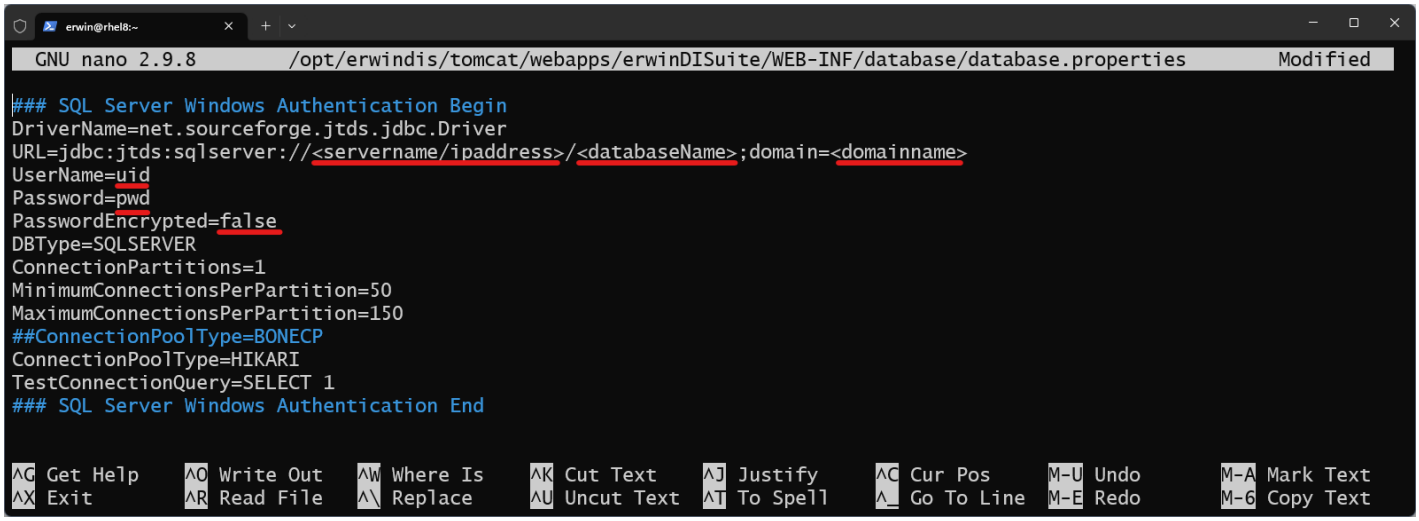
If you require using Windows Authentication mode, you will first need to comment out all rows between **SQL Server Begin** and **SQL Server End** section as shown in the example screen below.



Then scroll down to the **SQL Server Windows Authentication** section and uncomment the section by removing the # at the beginning of each line between **SQL SERVER Windows Authentication BEGIN** and **SQL SERVER Windows Authentication END** section as shown in the example below. Then enter the following parameters for your SQL Server environment.

- SERVER NAME or IP Address
- Database Name
- Domain
- UserName (this is still required for windows authentication)
- Password (this is still required for windows authentication)
- PasswordEncrypted = false

The parameters you will need to edit are indicated below in red.



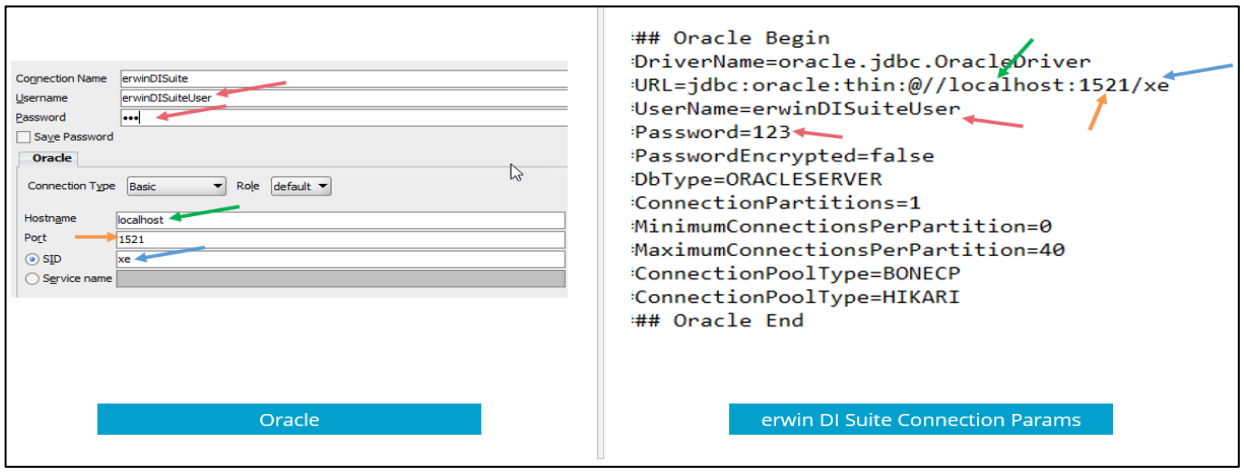
Potential additional parameters to the URL, ;useNTLMv2=true and/or ;ssl=required, if unable to connect.

2.7.2 - Configuring the “database.properties” file for Oracle Database

Uncomment the ORACLE section by removing the # at the beginning of each line (between ORACLE BEGIN and ORACLE END section)

Enter the following parameters

- SERVER NAME
- PORT# (default 1521)
- Database Name
- User Name
- Password
- PasswordEncrypted = false
- Sample screenshot of Parameters is shown below.



2.8 - Configuring the application documents repository path

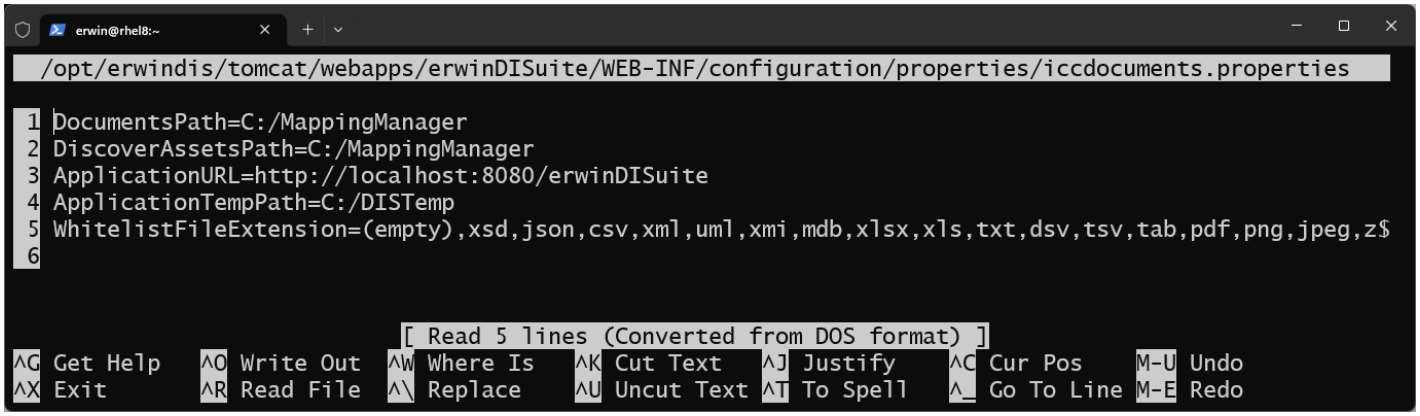
2.8.1 - Update the 'iccdocuments.properties' file.

Return to your UBUNTU server and open the 'iccdocuments.properties file with the editor of your choice.
In this example we will edit the file with the nano editor using elevated privileges via sudo.

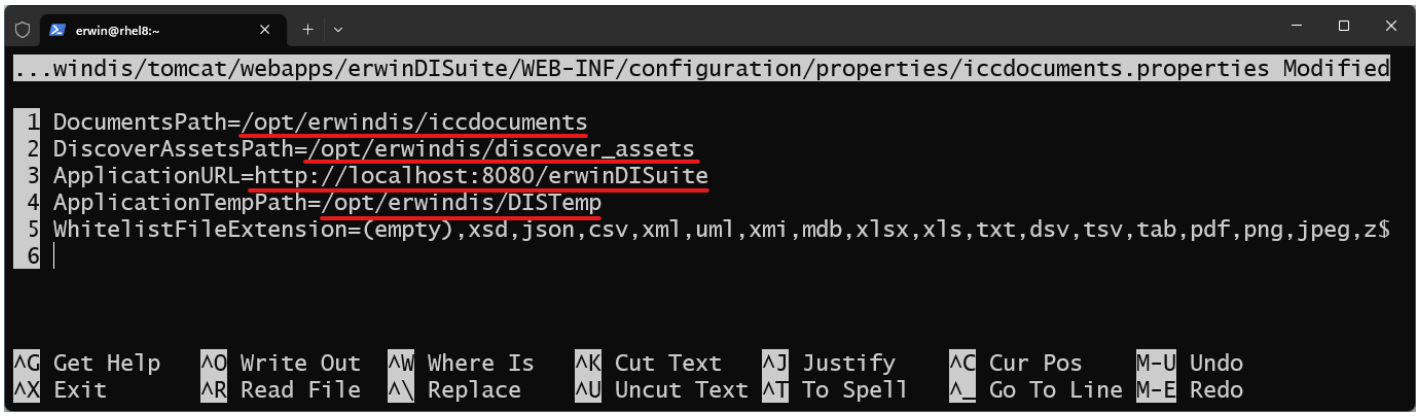
```
sudo nano -l /opt/QuestDI/tomcat/webapps/QuestDI/WEB-INF/configuration/properties/iccdocuments.properties
```

Note: The -l switch will cause the nano editor to display line numbers.

Once the file is open, notice that lines 1, 2, and 4 contain Windows filesystem paths.
We will need to edit these lines to point to the correct Linux filesystem paths.



Edit the lines to match the correct paths for your Linux environment. In the example below, we are using the paths previously created when we created additional folders in the QuestDI home `/opt/QuestDI` folder.



2.8.1.1 - Explanation of Paths

- **DocumentsPath**=Client_Folder_Path **E.g.** DocumentsPath=/opt/QuestDI/iccdocuments
  - This directory stores all the documents that are uploaded via the product as part of the rich media library and other attachment functionalities and this folder is used for backup and restoration purposes in case a shift to another server is required in the future.
- **ApplicationURL**= Client_DI_App_URL **E.g.** ApplicationURL= http://erwinserver:8080/QuestDI
  - This is the base Quest DI application URL that is used as part of the capabilities that generate Unique URLs to assets.
- **ApplicationTempPath**=Provide_New_Path_For_Temp_Files **E.g.** ApplicationTempPath= /opt/QuestDI/DISTemp
  - This path is used to store all the temp files uploaded into the application (typically outside the tomcat directory) and can be periodically deleted by an admin.
- **DiscoverAssetsPath**=Provide_Folder_Path **E.g.** DocumentsPath=/opt/QuestDI/discover_assets
  - This is typically the same path as the *ApplicationTempPath* variable but can be changed if needed. This path is used to sync the assets with the Discover Assets module and will be used going forward to support multiple instances.
- **Important:** Restart the Tomcat server once the Application and Database Properties file have been updated for these changes to come into effect.

2.8.2 - Start the tomcat server using the QuestDI.service unit file with the following command.

```
sudo systemctl start QuestDI.service
```

2.8.3 - Check status to view the result.

Verify tomcat is running using the QuestDI.service unit file.

```
sudo systemctl status QuestDI.service
```

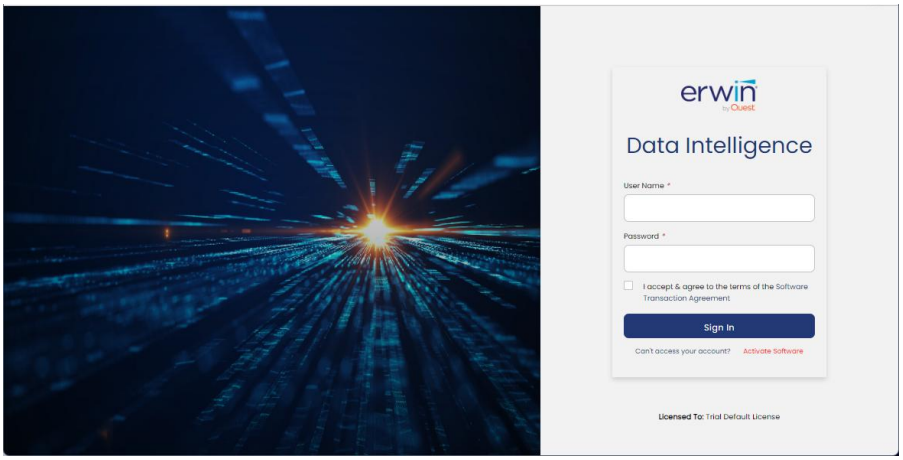
2.9 - Access the Quest Data Intelligence Login Screen

Now it is time to test your work. If you have faithfully completed all the steps outlined in this chapter, you should now be able to access the Quest Data Intelligence Login screen from your local browser.

2.9.1 - Open a web browser and type the URL for your server.

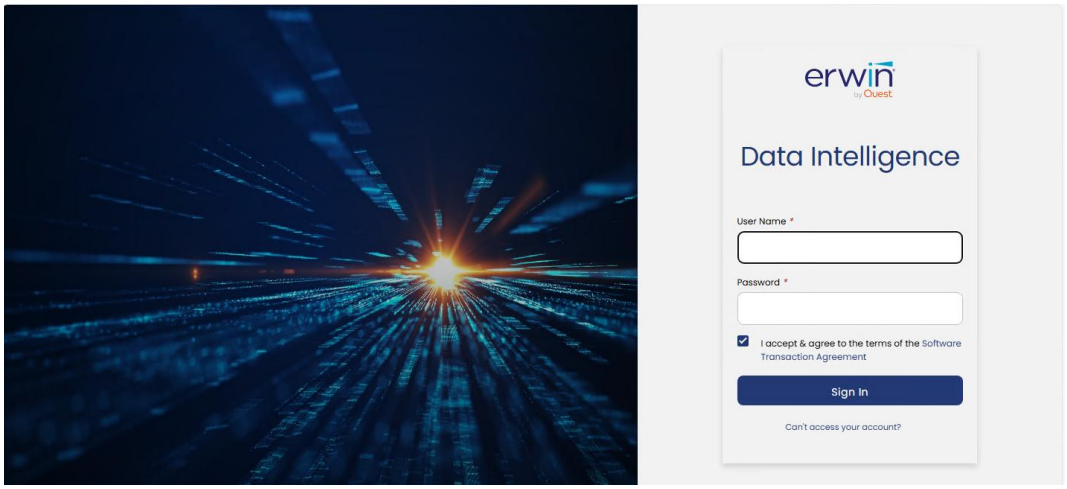
The URL format is **http://<ip-address/hostname>:8080/QuestDI**

- Replace ***<ip-address/hostname>*** with your correct IP address or DNS hostname.
- Port '***8080***' is the default port used earlier when configuring Tomcat. If you changed the port used during Tomcat configuration, remember to use the correct port for your environment.
- URI path '***/QuestDI***' is the default application name and URI path.
  If you deployed the war file with a different name, use the correct /path for your environment and replace the ***QuestDI*** variable accordingly e.g. ***erwinDataIntelligence***
- If all steps were completed accurately, and the correct URL is used your browser should display the Quest Data Intelligence login screen along with an option to apply your license key.

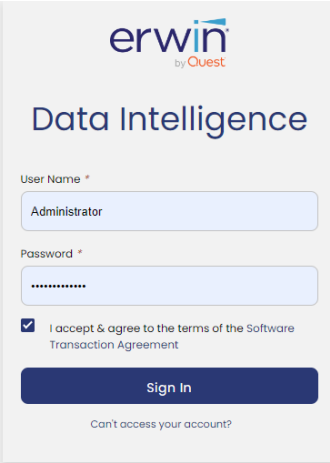


- Use the license key that has been provided to you to Activate the software (*How to activate the software* section below)

- You should now see the login screen with the license key successfully applied.



- Use your Administrator *User ID* and *Password* credentials to login to the application for the first time and configure all other aspects.



- We recommend that you change the default Administrator password upon login for security purposes.

2.9.2 - Activate the software.

A license key is required to activate the software. Login to the application will not work until the license key has been applied. If you do not have a license key, please reach out to your Quest\erwin sales contact to obtain your license key.

Once the license key has been obtained, return to the login screen, and click the 'Activate Software' link under the Sign in button.