

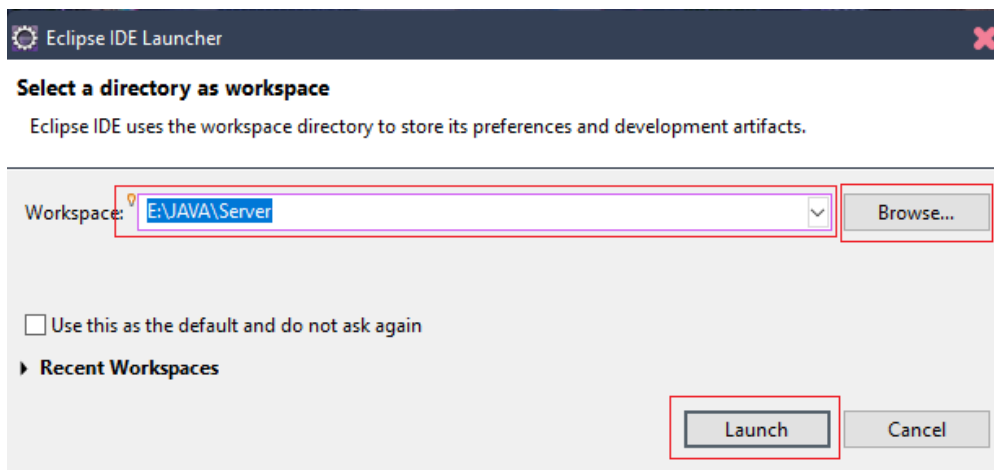
## Oracle Database Connection Pool in Tomcat: -

### Inside Eclipse: -

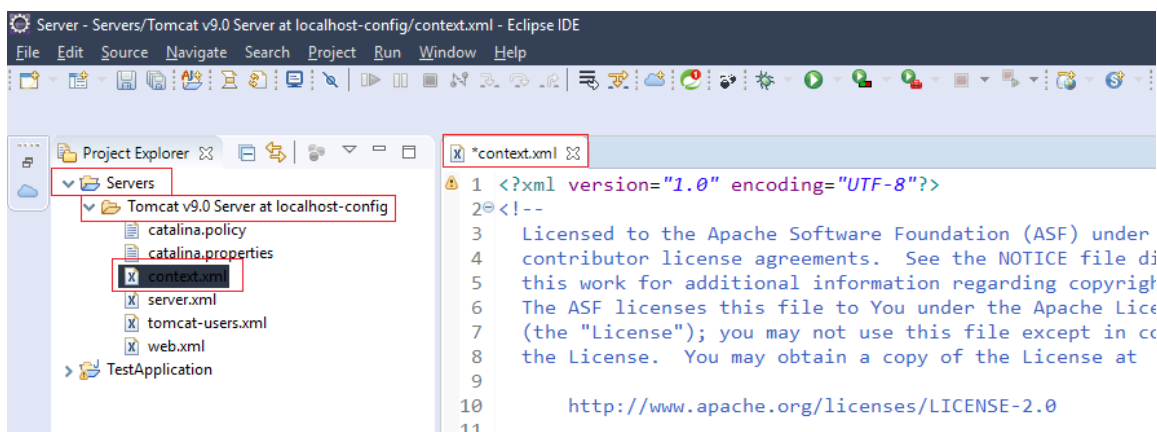
#### Step #1: Open Eclipse



Step #2: Choose/ browser location where you want to create Database Connection Pool [make sure there already Tomcat server is configured], then click on Launch



Step #3: Go to Server -> Tomcat v9.0 Server localhost-config folder, Open context.xml



Step #4: Go to browser search Tomcat 9 documentation, open JDBC Data Source from tomcat.apache.org [official website].

The screenshot shows a Google search results page for the query 'tomcat 9 documentation'. The search bar at the top contains the text 'tomcat 9 documentation'. Below the search bar, the results show 'About 29,30,000 results (0.36 seconds)'. The first result is from 'tomcat.apache.org > tomcat-9' and is titled 'Apache Tomcat 9 (9.0.36) - Documentation Index'. Below this title, there is a brief description: 'Jun 3, 2020 - Apache Tomcat version 9.0 implements the Servlet 4.0 and JavaServer Pages 2.3 specifications from the Java Community Process, and includes many additional features that make it a useful platform for developing and deploying web applications and web services.' Below the description, there are two links: 'JDBC DataSource' and 'Introduction'. The 'JDBC DataSource' link has a sub-link 'JNDI DataSource How-To. Table of Contents. Introduction ...'. The 'Introduction' link has a sub-link 'These are some of the key tomcat directories: /bin - Startup, ...'.

Step #5: Click on the Oracle option from Database Connection Pool option

The screenshot shows the 'JNDI Datasource How-To' page on the Apache Tomcat 9 website. The page has a header with the Apache Tomcat 9 logo and the version 'Version 9.0.36, Jun 3 2020'. Below the header, there is a 'Table of Contents' section with a list of links: 'Introduction', 'DriverManager, the service provider mechanism and memory leaks', 'Database Connection Pool (DBCP 2) Configurations', '1. Installation', '2. Preventing database connection pool leaks', '3. MySQL DBCP 2 Example', '4. Oracle 8i, 9i & 10g', and '5. PostgreSQL'. The 'Database Connection Pool (DBCP 2) Configurations' link is highlighted with a red box. Below the 'Table of Contents' section, there is a 'User Guide' section with a list of links: '1) Introduction', '2) Setup', '3) First webapp', '4) Deployer', '5) Manager', '6) Host Manager', and '7) Realms and AAA'.

Step #6: Copy the content of Context configuration

The screenshot shows the '1. Context configuration' section of the 'Database Connection Pool (DBCP 2) Configurations' page. The section title '1. Context configuration' is highlighted with a red box. Below the title, there is a paragraph of text: 'In a similar manner to the mysql config above, you will need to define your Datasource in your Context. Here driver to connect as user scott, password tiger to the sid called mysid. (Note: with the thin driver this sid is r default schema for the user scott.' Below the paragraph, there is a paragraph of text: 'Use of the OCI driver should simply involve a changing thin to oci in the URL string.' Below the paragraph, there is a code block containing the following XML snippet: 

```
<Resource name="jdbc/myoracle" auth="Container"
  type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
  url="jdbc:oracle:thin:@127.0.0.1:1521:mysid"
  username="scott" password="tiger" maxTotal="20" maxIdle="10"
  maxWaitMillis="-1"/>
```

Step #7. Go to context.xml and past the content in <Context> tag

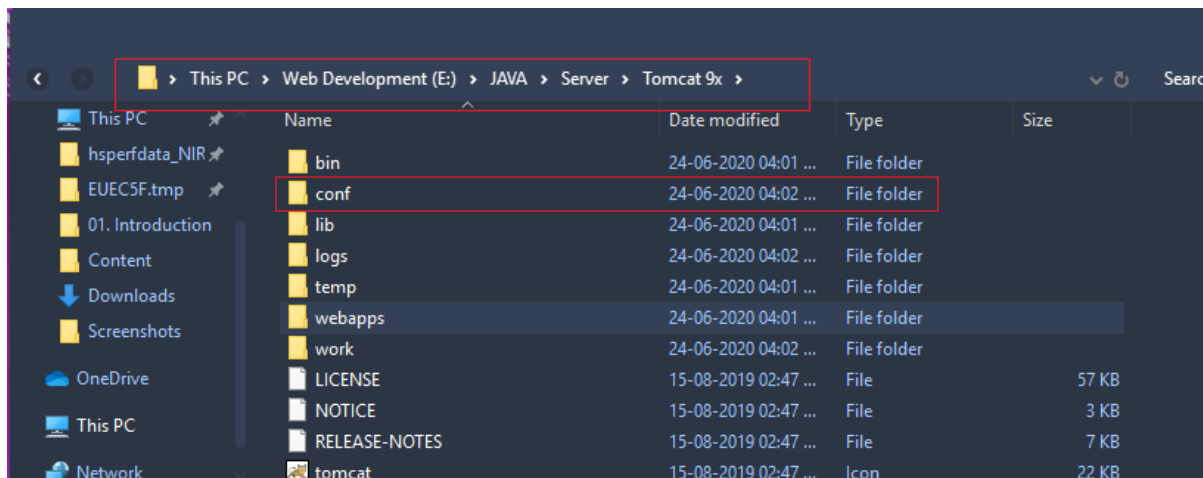
```
19 <Context>
20
21     <Resource name="jdbc/myoracle" auth="Container"
22             type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
23             url="jdbc:oracle:thin:@127.0.0.1:1521:mysid"
24             username="scott" password="tiger" maxTotal="20" maxIdle="10"
25             maxWaitMillis="-1"/>
26
27 <!-- Default set of monitored resources. If one of these changes, the -->
28 <!-- web application will be reloaded. -->
29 <WatchedResource>WEB-INF/web.xml</WatchedResource>
```

Step #8: Change the name, URL's sid, username, password as per your requirement then save it, your connection pool is done inside eclipse now you can use it in any application in that workspace.

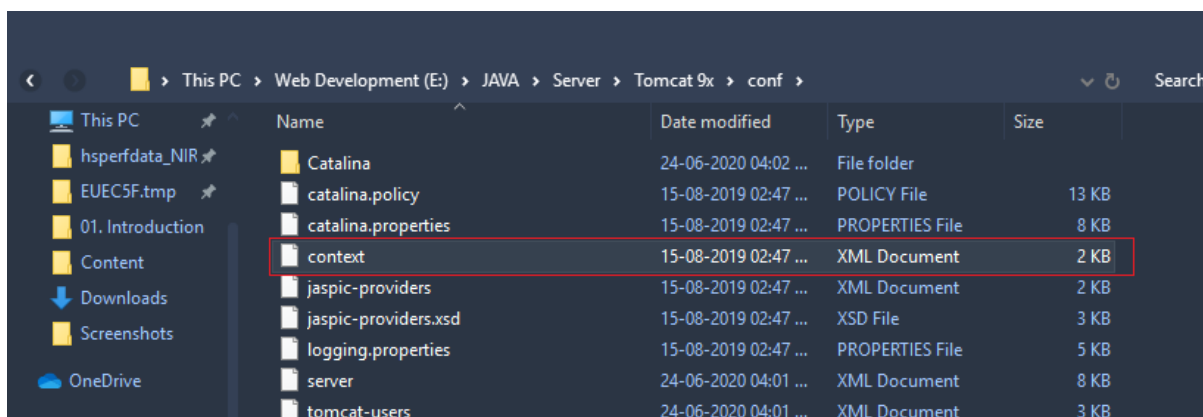
```
19 <Context>
20
21     <Resource name="DsJndi" auth="Container"
22             type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
23             url="jdbc:oracle:thin:@127.0.0.1:1521:xe"
24             username="scott" password="tiger" maxTotal="20" maxIdle="10"
25             maxWaitMillis="-1"/>
26
27 <!-- Default set of monitored resources. If one of these changes, the -->
28 <!-- web application will be reloaded. -->
```

## Outside Eclipse [Installation directory]: -

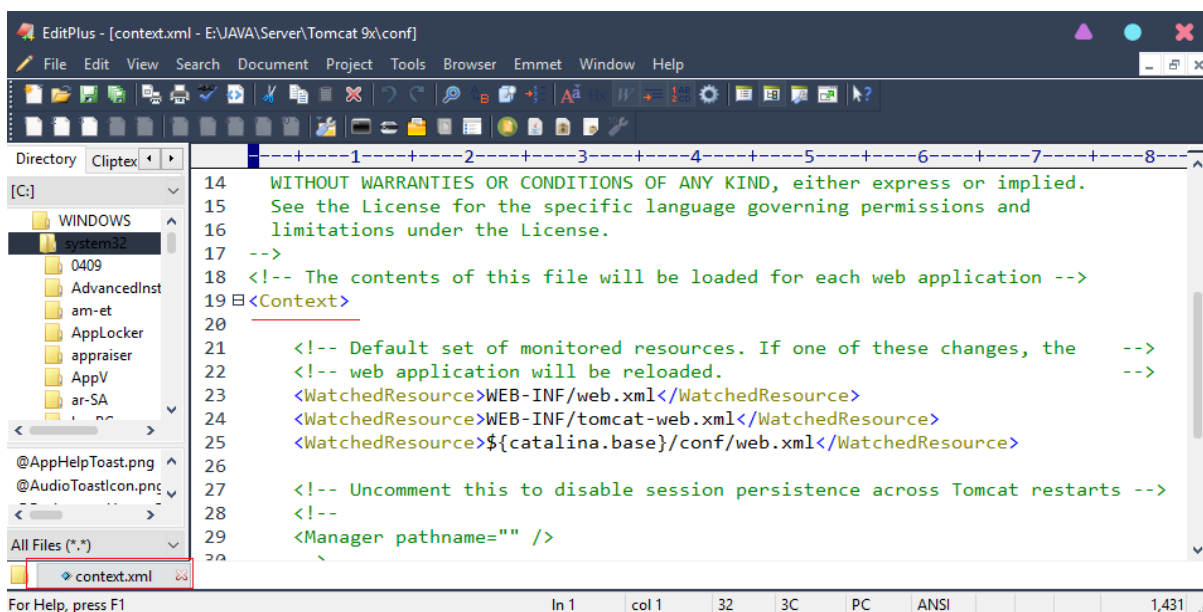
Step #1: Open the <Tomcat Home>\conf folder



Step #2: Choose context.xml



Step #3: Open Context.xml file with any editor like Edit+, Notepad++, etc.



Step #4: Go to browser search Tomcat 9 documentation, open JDBC Data Source from tomcat.apache.org [official website].

The screenshot shows a Google search results page for the query 'tomcat 9 documentation'. The search bar at the top contains the text 'tomcat 9 documentation'. Below the search bar, the results show 'About 29,30,000 results (0.36 seconds)'. The first result is from 'tomcat.apache.org > tomcat-9' and is titled 'Apache Tomcat 9 (9.0.36) - Documentation Index'. Below this title, there is a brief description: 'Jun 3, 2020 - Apache Tomcat version 9.0 implements the Servlet 4.0 and JavaServer Pages 2.3 specifications from the Java Community Process, and includes many additional features that make it a useful platform for developing and deploying web applications and web services.' Below the description, there are two links: 'JDBC DataSource' and 'Introduction'. The 'JDBC DataSource' link has a sub-link 'JNDI DataSource How-To. Table of Contents. Introduction ...'. The 'Introduction' link has a sub-link 'These are some of the key tomcat directories: /bin - Startup, ...'.

Step #5: Click on the Oracle option from Database Connection Pool option

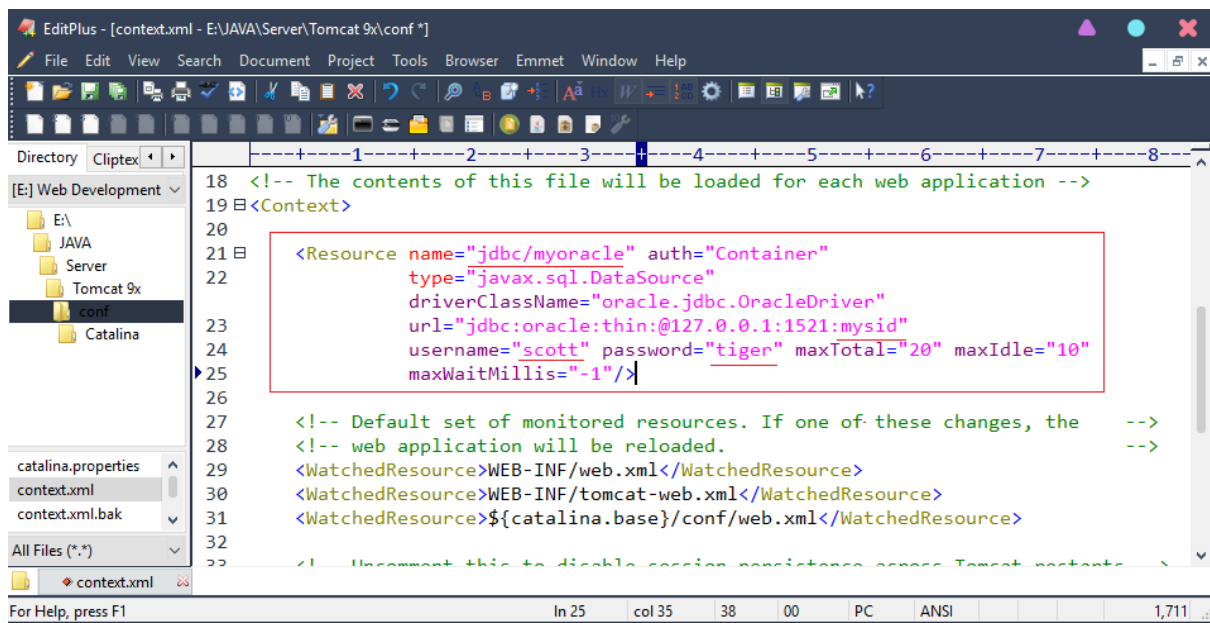
The screenshot shows the 'JNDI Datasource How-To' page on the Apache Tomcat 9 website. The page has a header with the Apache Tomcat 9 logo and the version 'Version 9.0.36, Jun 3 2020'. Below the header, there is a 'Table of Contents' section with a list of links: 'Introduction', 'DriverManager, the service provider mechanism and memory leaks', 'Database Connection Pool (DBCP 2) Configurations', '1. Installation', '2. Preventing database connection pool leaks', '3. MySQL DBCP 2 Example', '4. Oracle 8i, 9i & 10g', and '5. PostgreSQL'. The 'Database Connection Pool (DBCP 2) Configurations' link is highlighted with a red box. Below the 'Table of Contents' section, there is a 'User Guide' section with a list of links: '1) Introduction', '2) Setup', '3) First webapp', '4) Deployer', '5) Manager', '6) Host Manager', and '7) Realms and AAA'.

Step #6: Copy the content of Context configuration

The screenshot shows the '1. Context configuration' section of the 'Database Connection Pool (DBCP 2) Configurations' page. The section title is '1. Context configuration'. Below the title, there is a paragraph: 'In a similar manner to the mysql config above, you will need to define your Datasource in your Context. Here driver to connect as user scott, password tiger to the sid called mysid. (Note: with the thin driver this sid is r default schema for the user scott.' Below the paragraph, there is a paragraph: 'Use of the OCI driver should simply involve a changing thin to oci in the URL string.' Below the paragraph, there is a code block with the following XML code: 

```
<Resource name="jdbc/myoracle" auth="Container"
  type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
  url="jdbc:oracle:thin:@127.0.0.1:1521:mysid"
  username="scott" password="tiger" maxTotal="20" maxIdle="10"
  maxWaitMillis="-1"/>
```

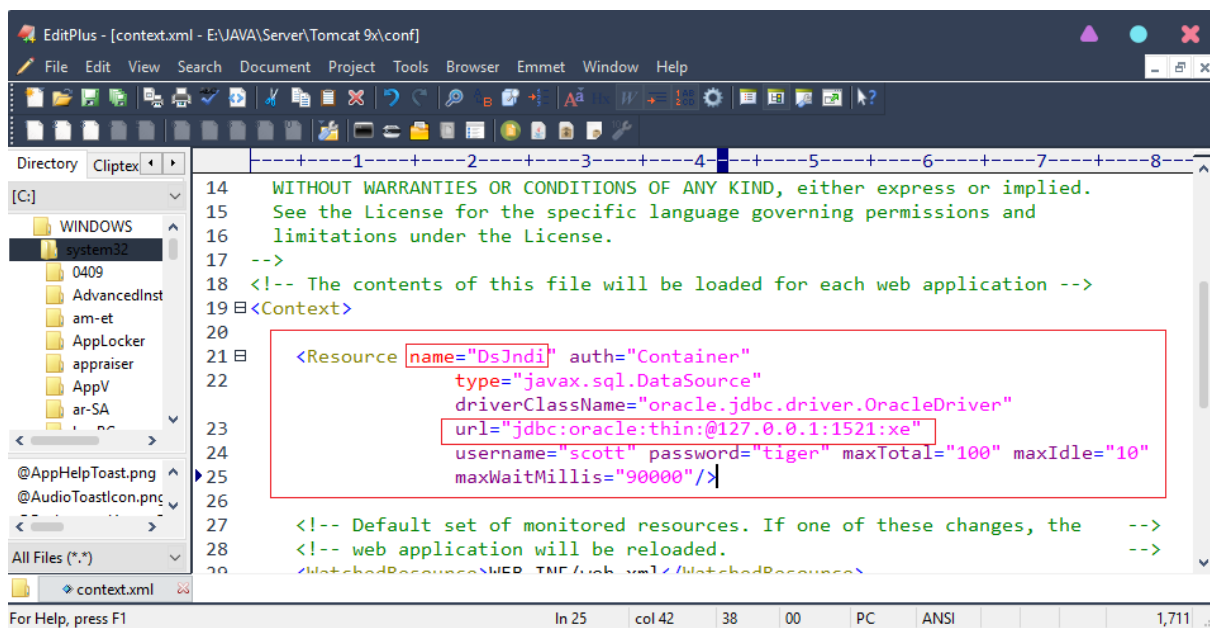
Step #7. Go to context.xml and past the content in <Context> tag



The screenshot shows the EditPlus editor with the file context.xml open. The left sidebar shows the directory structure: E:\Web Development > JAVA > Server > Tomcat 9x > conf. The main editor area shows the following XML content:

```
18 <!-- The contents of this file will be loaded for each web application -->
19 <Context>
20
21
22   <Resource name="jdbc/myoracle" auth="Container"
23     type="javax.sql.DataSource"
24     driverClassName="oracle.jdbc.OracleDriver"
25     url="jdbc:oracle:thin:@127.0.0.1:1521:mysid"
26     username="scott" password="tiger" maxTotal="20" maxIdle="10"
27     maxWaitMillis="-1"/>
28
29   <!-- Default set of monitored resources. If one of these changes, the -->
30   <!-- web application will be reloaded. -->
31   <WatchedResource>WEB-INF/web.xml</WatchedResource>
32   <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
33   <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
34
35 </Context>
```

Step #8: Change the name, URL's sid, username, password as per your requirement then save it, your connection pool is done outside of eclipse.



The screenshot shows the EditPlus editor with the file context.xml open. The left sidebar shows the directory structure: [C:] > WINDOWS > system32. The main editor area shows the following XML content:

```
14 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
15 See the License for the specific language governing permissions and
16 limitations under the License.
17 -->
18 <!-- The contents of this file will be loaded for each web application -->
19 <Context>
20
21
22   <Resource name="DsIndi" auth="Container"
23     type="javax.sql.DataSource"
24     driverClassName="oracle.jdbc.driver.OracleDriver"
25     url="jdbc:oracle:thin:@127.0.0.1:1521:xe"
26     username="scott" password="tiger" maxTotal="100" maxIdle="10"
27     maxWaitMillis="90000"/>
28
29   <!-- Default set of monitored resources. If one of these changes, the -->
30   <!-- web application will be reloaded. -->
31   <WatchedResource>WEB-INF/web.xml</WatchedResource>
32   <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
33   <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
34
35 </Context>
```