

CURSO JAVA

EJERCICIO

CREACIÓN POOL DE CONEXIONES CON GLASSFISH



Experiencia y Conocimiento para tu vida

CURSO JAVA

www.globalmentoring.com.mx

OBJETIVO DEL EJERCICIO

- El objetivo del ejercicio es crear un pool de conexiones con MySql en Glassfish. Al finalizar deberemos observar el siguiente resultado desde nuestro servidor de Glassfish.

Resources (4)

New...DeleteEnableDisable

| Select | JNDI Name | Logical JNDI Name | Enabled | Connection Pool |
|--------------------------|------------------|-----------------------------|---------|-----------------|
| <input type="checkbox"/> | jdbc/PersonaDb | | ✓ | PersonaPool |
| <input type="checkbox"/> | jdbc/__TimerPool | | ✓ | __TimerPool |
| <input type="checkbox"/> | jdbc/__default | java:comp/DefaultDataSource | ✓ | DerbyPool |
| <input type="checkbox"/> | jdbc/sample | | ✓ | SamplePool |



Experiencia y Conocimiento para tu vida

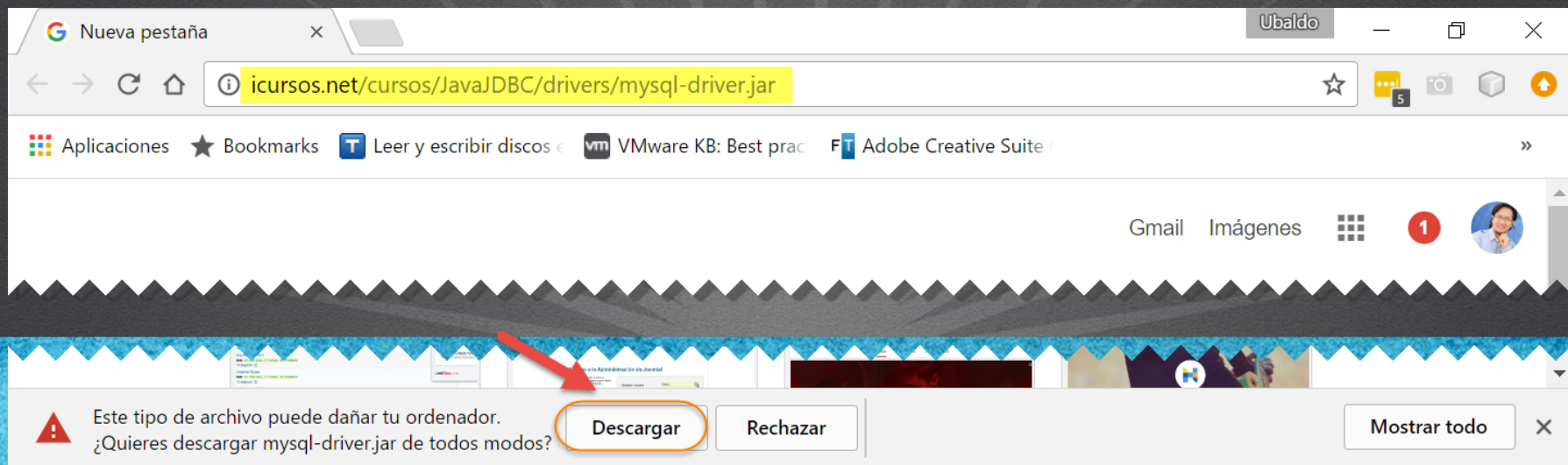
CURSO JAVA

www.globalmentoring.com.mx

PASO 1. CONFIGURACION CONEXIÓN JTA

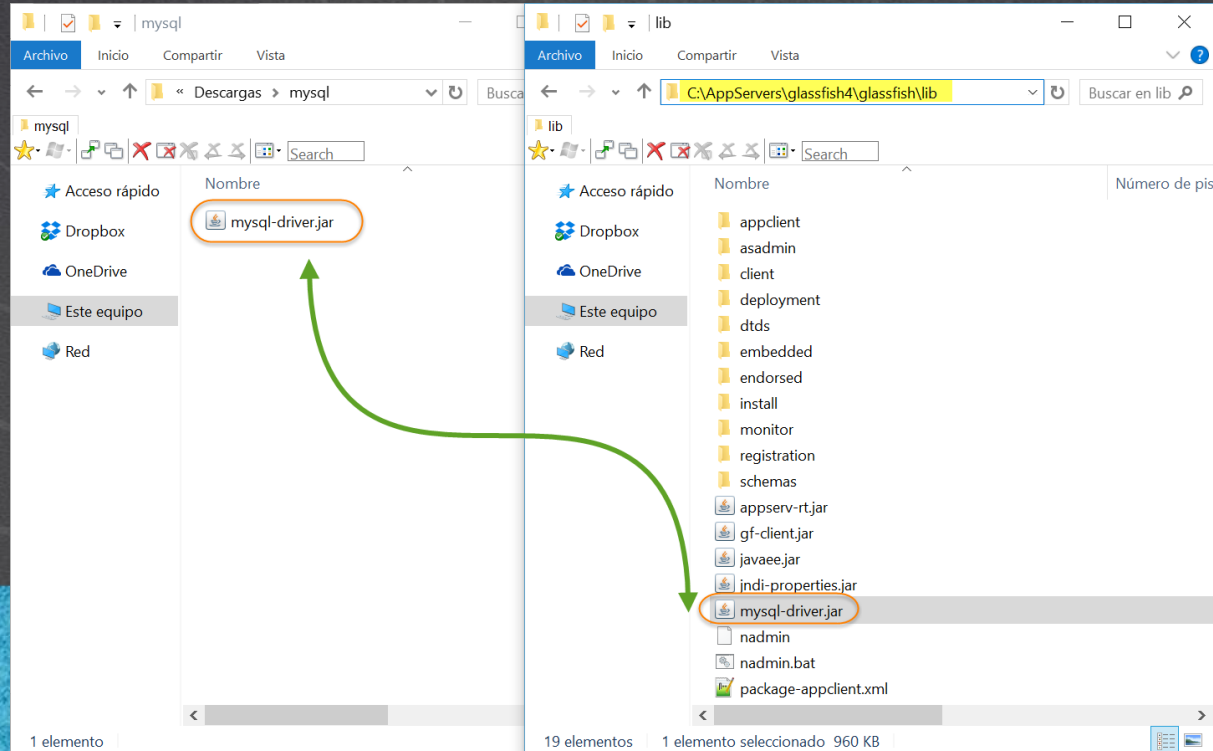
Configuramos la conexión de JTA en Glassfish. Agregamos el driver de mysql a Glassfish. Descargamos el .jar de mysql:

<http://icursos.net/cursos/JavaJDBC/drivers/mysql-driver.jar>



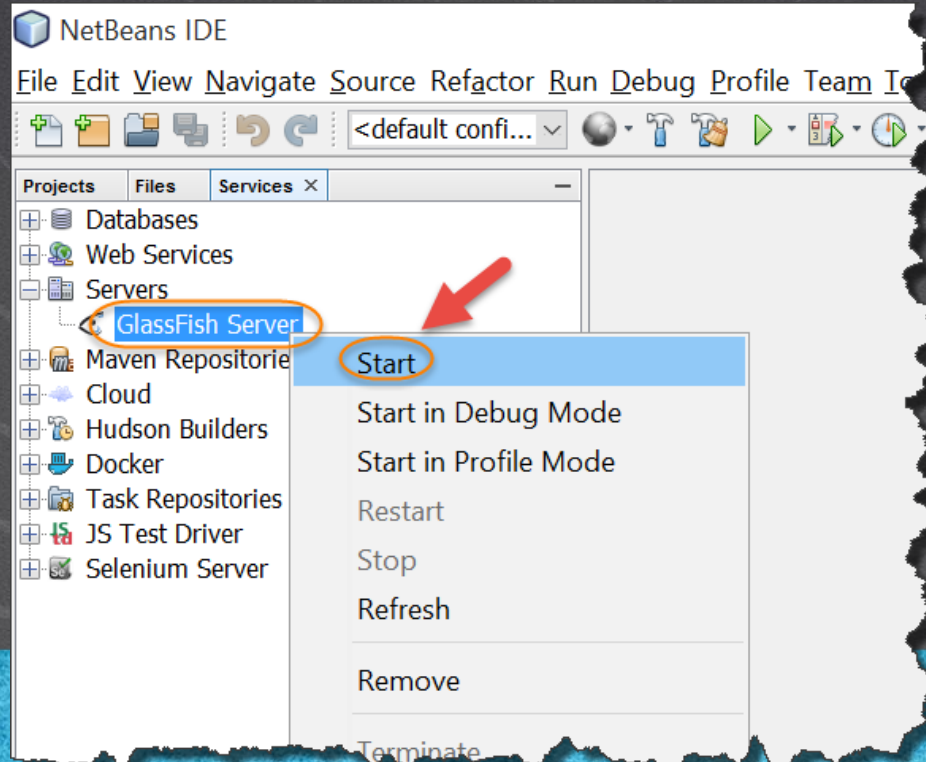
PASO 1. CONFIGURACION CONEXIÓN JTA

Copiamos el archivo recién descargado en la ruta de instalación de Glassfish. Ej. C:\AppServers\glassfish4\glassfish\lib



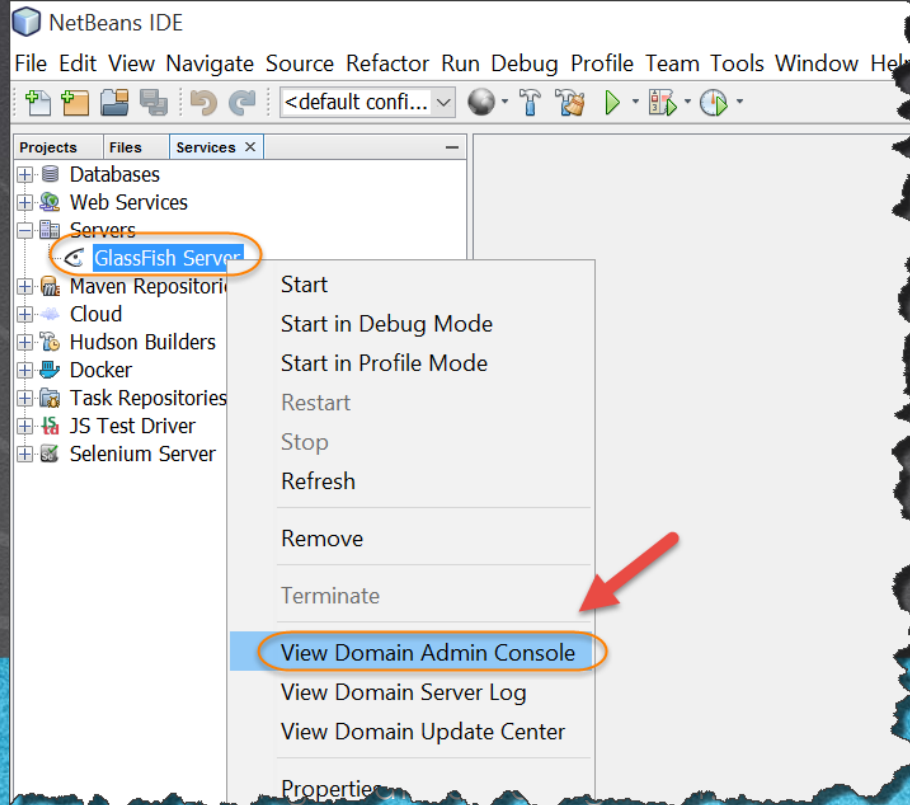
PASO 1. CONFIGURACION CONEXIÓN JTA

Abrimos Netbeas y levantamos el servidor de Glassfish:



PASO 1. CONFIGURACION CONEXIÓN JTA

Entramos a la consola de administración de Glassfish:



PASO 1. CONFIGURACION CONEXIÓN JTA

Creamos un nuevo pool de conexiones:

The screenshot shows the GlassFish Administration Console interface. The left sidebar contains a 'Tree' view with the following structure:

- Server (Admin Server)
 - Clusters
 - Standalone Instances
 - Nodes
 - Applications
 - Lifecycle Modules
 - Monitoring Data
 - Resources** (1)
 - Concurrent Resources
 - Connectors
 - JDBC** (2)
 - JDBC Resources
 - JDBC Connection Pools** (3)
 - JMS Resources
 - JNDI

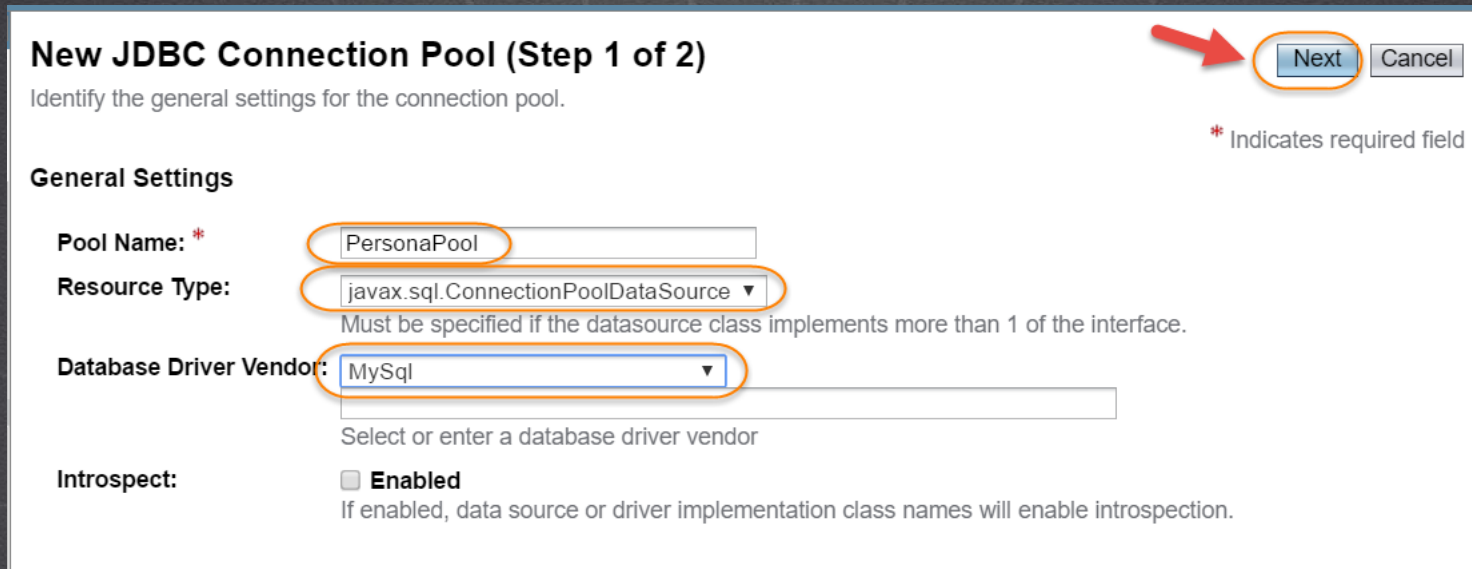
The main content area is titled 'JDBC Connection Pools'. It includes a description: 'To store, organize, and retrieve data, most applications use relational databases. Java EE applications access relational databases through the JDBC API. Before an application can access a database, it must get a connection.'

Below the description, there is a section 'Pools (3)' with a 'New...' button (4) and a 'Delete' button. A table lists the existing pools:

| Select | Pool Name | Resource Type | Classname | Description |
|--------------------------|-------------|------------------------|--|-------------|
| <input type="checkbox"/> | DerbyPool | javax.sql.DataSource | org.apache.derby.jdbc.ClientDataSource | |
| <input type="checkbox"/> | SamplePool | javax.sql.DataSource | org.apache.derby.jdbc.ClientDataSource | |
| <input type="checkbox"/> | __TimerPool | javax.sql.XADataSource | org.apache.derby.jdbc.EmbeddedXADataSource | |

PASO 1. CONFIGURACION CONEXIÓN JTA

Proporcionamos los siguientes datos para crear el pool de conexiones:



New JDBC Connection Pool (Step 1 of 2)

Identify the general settings for the connection pool.

General Settings

Pool Name: * PersonaPool

Resource Type: javax.sql.ConnectionPoolDataSource ▼

Must be specified if the datasource class implements more than 1 of the interface.

Database Driver Vendor: MySql ▼

Select or enter a database driver vendor

Introspect: ☒ **Enabled**

If enabled, data source or driver implementation class names will enable introspection.

* Indicates required field

Next **Cancel**

CURSO JAVA

www.globalmentoring.com.mx

PASO 1. CONFIGURACION CONEXIÓN JTA

Dejamos los valores por default:

New JDBC Connection Pool (Step 2 of 2)

PreviousFinishCancel

Identify the general settings for the connection pool. Datasource Classname or Driver Classname must be specified for the connection pool.

* Indicates required field

General Settings

| | |
|--------------------------------|---|
| Pool Name: | PersonaPool |
| Resource Type: | javax.sql.ConnectionPoolDataSource |
| Database Driver Vendor: | MySql |
| Datasource Classname: | <div>com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource ▼</div> <div>Select or enter vendor-specific classname that implements the DataSource and/or XADataSource APIs</div> |
| Driver Classname: | <div>▼</div> <div>Select or enter vendor-specific classname that implements the java.sql.Driver interface.</div> |
| Ping: | <input checked="" type="checkbox"/> Enabled When enabled, the pool is pinged during creation or reconfiguration to identify and warn of any erroneous values for its attributes |
| Description: | |

PASO 1. CONFIGURACION CONEXIÓN JTA

Dejamos los valores por default:

Pool Settings

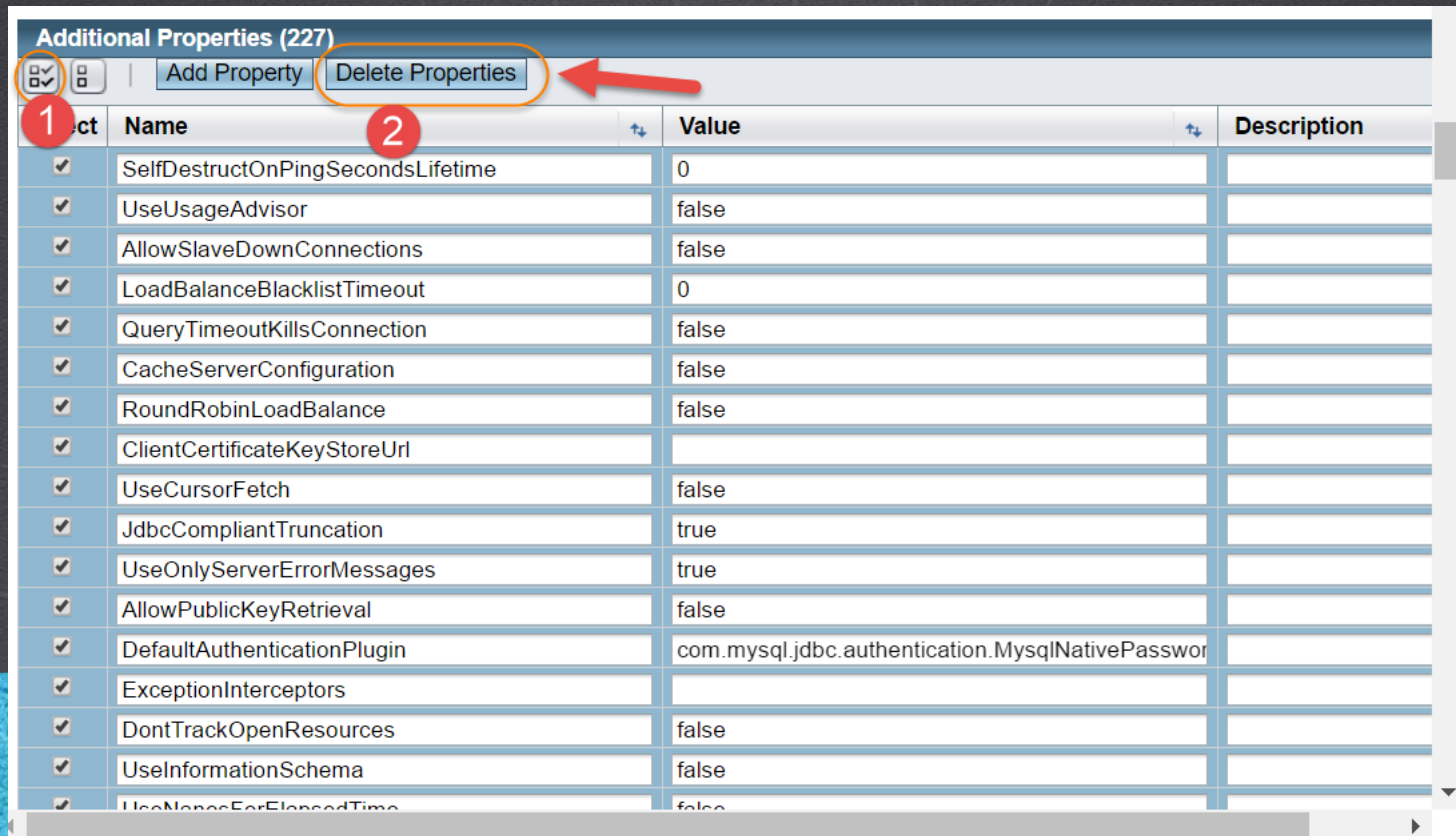
- Initial and Minimum Pool Size:** Connections
Minimum and initial number of connections maintained in the pool
- Maximum Pool Size:** Connections
Maximum number of connections that can be created to satisfy client requests
- Pool Resize Quantity:** Connections
Number of connections to be removed when pool idle timeout expires
- Idle Timeout:** Seconds
Maximum time that connection can remain idle in the pool
- Max Wait Time:** Milliseconds
Amount of time caller waits before connection timeout is sent

Transaction

- Non Transactional Connections:** ☐ **Enabled**
Returns non-transactional connections
- Transaction Isolation:**
If unspecified, use default level for JDBC Driver
- Isolation Level:** ☒ **Guaranteed**
All connections use same isolation level; requires Transaction Isolation

PASO 1. CONFIGURACION CONEXIÓN JTA

Borramos todas las propiedades:



Additional Properties (227)

1 ☒ ☐ | Add Property Delete Properties 2

| Select | Name | Value | Description |
|-------------------------------------|-----------------------------------|--|-------------|
| <input checked="" type="checkbox"/> | SelfDestructOnPingSecondsLifetime | 0 | |
| <input checked="" type="checkbox"/> | UseUsageAdvisor | false | |
| <input checked="" type="checkbox"/> | AllowSlaveDownConnections | false | |
| <input checked="" type="checkbox"/> | LoadBalanceBlacklistTimeout | 0 | |
| <input checked="" type="checkbox"/> | QueryTimeoutKillsConnection | false | |
| <input checked="" type="checkbox"/> | CacheServerConfiguration | false | |
| <input checked="" type="checkbox"/> | RoundRobinLoadBalance | false | |
| <input checked="" type="checkbox"/> | ClientCertificateKeyStoreUrl | | |
| <input checked="" type="checkbox"/> | UseCursorFetch | false | |
| <input checked="" type="checkbox"/> | JdbcCompliantTruncation | true | |
| <input checked="" type="checkbox"/> | UseOnlyServerErrorMessages | true | |
| <input checked="" type="checkbox"/> | AllowPublicKeyRetrieval | false | |
| <input checked="" type="checkbox"/> | DefaultAuthenticationPlugin | com.mysql.jdbc.authentication.MysqlNativePasswor | |
| <input checked="" type="checkbox"/> | ExceptionInterceptors | | |
| <input checked="" type="checkbox"/> | DontTrackOpenResources | false | |
| <input checked="" type="checkbox"/> | UseInformationSchema | false | |
| <input checked="" type="checkbox"/> | UseNoncForElapsedTime | false | |

PASO 1. CONFIGURACION CONEXIÓN JTA

Agregamos las siguientes propiedades:

| | |
|----------------|-----------------------|
| portNumber | 3306 |
| databaseName | recursos_humanos |
| datasourceName | com.mysql.jdbc.Driver |
| serverName | localhost |
| user | root |
| password | admin |

CURSO JAVA

www.globalmentoring.com.mx

PASO 1. CONFIGURACION CONEXIÓN JTA

Agregamos las siguientes propiedades, proporcionando los valores mostrados y después damos click en Finish. Los valores pueden variar según la base de datos y demás valores que utilicemos para conectarnos a MySQL:

Additional Properties (6)

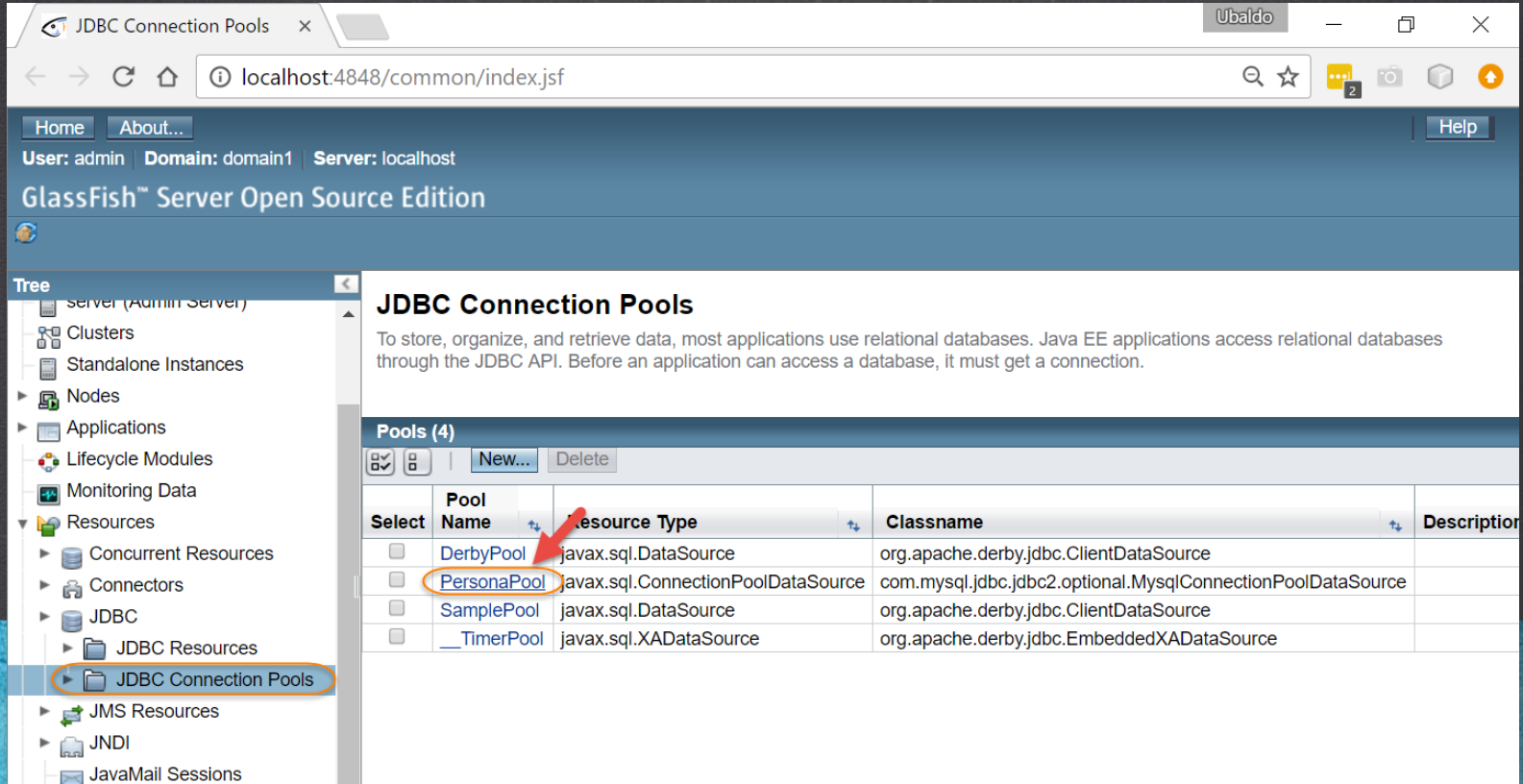
☒ ☐ | **Add Property** Delete Properties

| Select | Name | Value | Description |
|--------------------------|----------------|-----------------------|-------------|
| <input type="checkbox"/> | portNumber | 3306 | |
| <input type="checkbox"/> | databaseName | recursos_humanos | |
| <input type="checkbox"/> | datasourceName | com.mysql.jdbc.Driver | |
| <input type="checkbox"/> | serverName | localhost | |
| <input type="checkbox"/> | user | root | |
| <input type="checkbox"/> | password | admin | |

Previous **Finish** Cancel

PASO 1. CONFIGURACION CONEXIÓN JTA

Verificamos la conexión a mysql desde Glassfish:



The screenshot shows the GlassFish Administration Console interface. The browser address bar indicates the URL is `localhost:4848/common/index.jsf`. The page title is "GlassFish™ Server Open Source Edition". The left sidebar shows a tree view of the server configuration, with "JDBC Connection Pools" selected under the "Resources" section. The main content area is titled "JDBC Connection Pools" and contains a table listing the configured pools. A red arrow points to the "PersonaPool" entry, which is highlighted with a blue circle. The table columns are "Select", "Pool Name", "Resource Type", "Classname", and "Description".

| Select | Pool Name | Resource Type | Classname | Description |
|--------------------------|-------------|------------------------------------|---|-------------|
| <input type="checkbox"/> | DerbyPool | javax.sql.DataSource | org.apache.derby.jdbc.ClientDataSource | |
| <input type="checkbox"/> | PersonaPool | javax.sql.ConnectionPoolDataSource | com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource | |
| <input type="checkbox"/> | SamplePool | javax.sql.DataSource | org.apache.derby.jdbc.ClientDataSource | |
| <input type="checkbox"/> | __TimerPool | javax.sql.XADataSource | org.apache.derby.jdbc.EmbeddedXADataSource | |

PASO 1. CONFIGURACION CONEXIÓN JTA

Verificamos la conexión a mysql desde Glassfish:

The screenshot shows the GlassFish Admin Console interface. The browser address bar displays `localhost:4848/common/index.jsf`. The page title is "GlassFish™ Server Open Source Edition". The left sidebar shows a tree view of the server configuration, with "JDBC" expanded and "PersonaPool" selected. The main content area is titled "Edit JDBC Connection Pool" and has three tabs: "General", "Advanced", and "Additional Properties". The "General" tab is active. It contains a description of a JDBC connection pool and buttons for "Load Defaults", "Flush", and "Ping". A red arrow points to the "Ping" button. Below the tabs, the "General Settings" section includes fields for "Pool Name" (PersonaPool), "Resource Type" (javax.sql.ConnectionPoolDataSource), "Datasource Classname" (com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource), and "Driver Classname". The "Ping" checkbox is checked and labeled "Enabled".

Ubaldo

localhost:4848/common/index.jsf

Home About... Help

User: admin Domain: domain1 Server: localhost

GlassFish™ Server Open Source Edition

Tree

- Server (Admin Server)
 - Clusters
 - Standalone Instances
 - Nodes
 - Applications
 - Lifecycle Modules
 - Monitoring Data
 - Resources
 - Concurrent Resources
 - Connectors
 - JDBC
 - JDBC Resources
 - JDBC Connection Pools
 - DerbyPool
 - PersonaPool
 - SamplePool
 - TimerPool

General Advanced Additional Properties

Edit JDBC Connection Pool Save Cancel

Modify an existing JDBC connection pool. A JDBC connection pool is a group of reusable connections for a particular database.

Load Defaults Flush Ping

* Indicates required field

General Settings

Pool Name: PersonaPool

Resource Type: javax.sql.ConnectionPoolDataSource
Must be specified if the datasource class implements more than 1 of the interface.

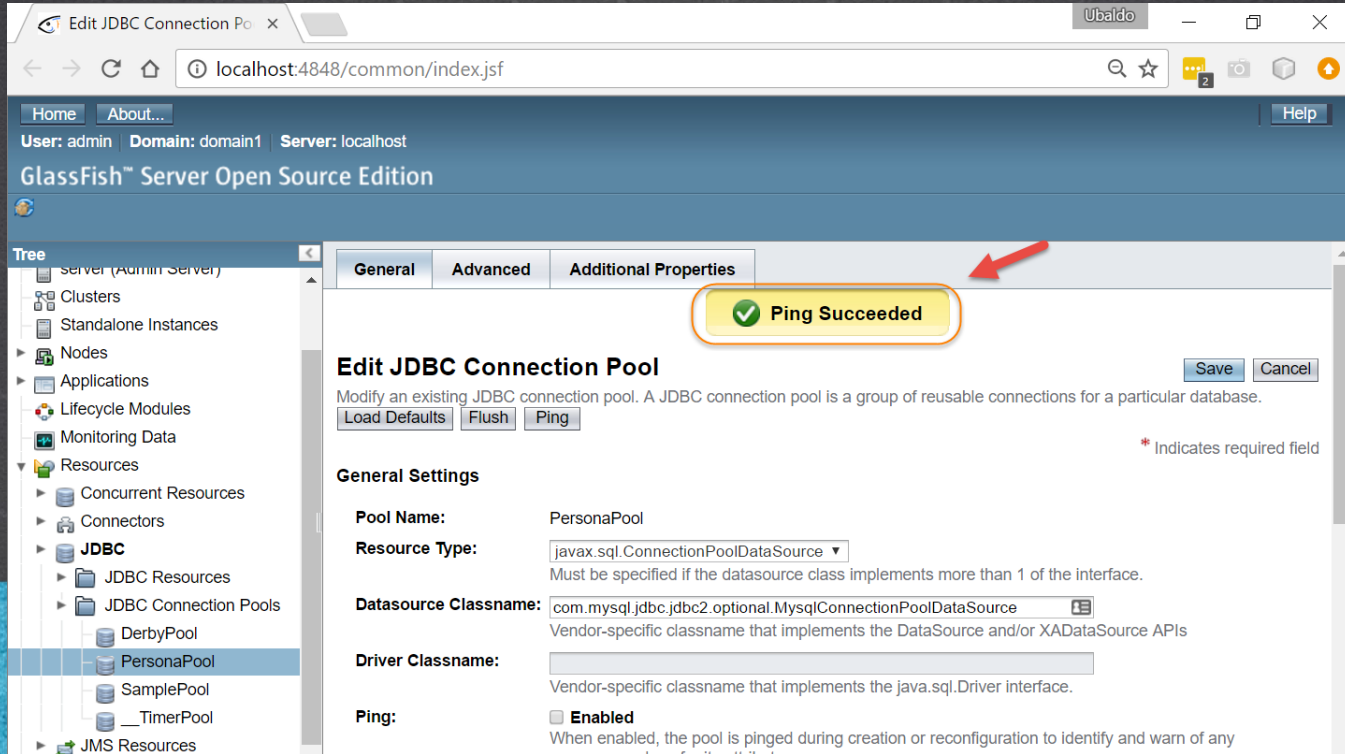
Datasource Classname: com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource
Vendor-specific classname that implements the DataSource and/or XADataSource APIs

Driver Classname:
Vendor-specific classname that implements the java.sql.Driver interface.

Ping: ☒ **Enabled**
When enabled, the pool is pinged during creation or reconfiguration to identify and warn of any erroneous values for its attributes

PASO 1. CONFIGURACION CONEXIÓN JTA

Verificamos la conexión a mysql desde Glassfish. Si hace ping la conexión ha sido exitosa:



The screenshot shows the GlassFish Admin Console interface. The left sidebar displays the navigation tree with 'Resources' expanded and 'JDBC' selected. The main content area shows the 'Edit JDBC Connection Pool' configuration for 'PersonaPool'. The 'Ping' button is highlighted with a red arrow, and a green checkmark icon with the text 'Ping Succeeded' is displayed above it. The configuration details are as follows:

| Field | Value |
|----------------------|---|
| Pool Name | PersonaPool |
| Resource Type | javax.sql.ConnectionPoolDataSource |
| Datasource Classname | com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource |
| Driver Classname | |
| Ping | <input checked="" type="checkbox"/> Enabled |

PASO 2. CONFIGURACION CONEXIÓN JTA

Creamos ahora el recurso de JDBC:

The screenshot shows the GlassFish Server Administration console. The browser address bar indicates the URL is `localhost:4848/common/index.jsf`. The page title is "JDBC Resources". The left sidebar shows the "Tree" view with "JDBC" and "JDBC Resources" highlighted. The main area shows a table of existing JDBC resources. Red annotations include a circle around "JDBC" in the tree (1), a circle around "JDBC Resources" in the tree (2), a circle around the "New..." button (3), and a red arrow pointing to the "New..." button.

JDBC Resources

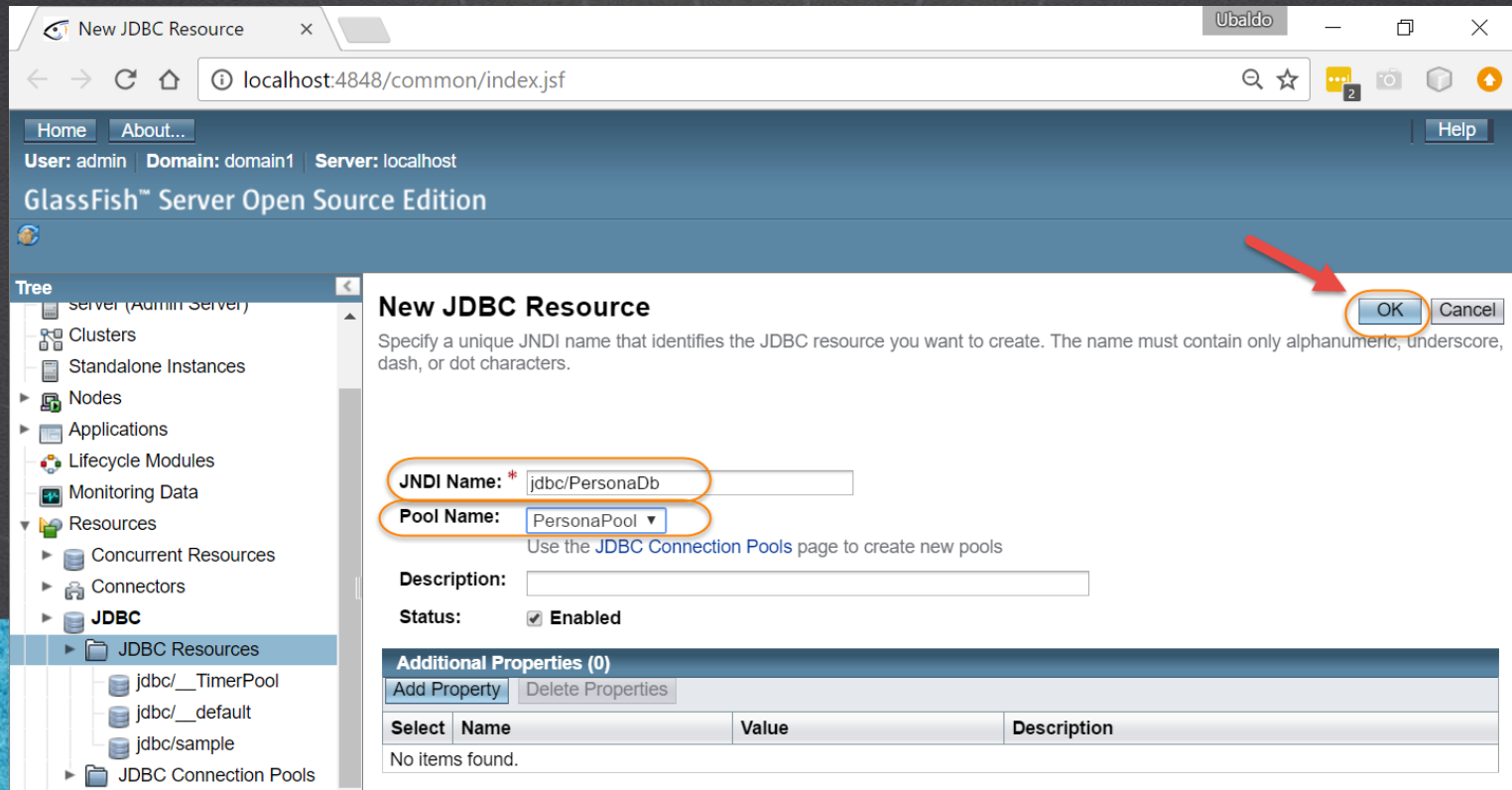
JDBC resources provide applications with a means to connect to a database.

Resources (3)

| Select | JNDI Name | Logical JNDI Name | Enabled | Connection Pool | Description |
|--------------------------|----------------------------------|---|---------|-----------------------------|-------------|
| <input type="checkbox"/> | jdbc/__TimerPool | | ✓ | __TimerPool | |
| <input type="checkbox"/> | jdbc/__default | java:comp/DefaultDataSource | ✓ | DerbyPool | |
| <input type="checkbox"/> | jdbc/sample | | ✓ | SamplePool | |

PASO 2. CONFIGURACION CONEXIÓN JTA

Creamos ahora el recurso de **jdbc/PersonaDb**. Este nombre puede ser cualquiera, sin embargo es el que utilizaremos vía JTA desde nuestras aplicaciones Java:



The screenshot shows the 'New JDBC Resource' configuration page in the GlassFish administration console. The left sidebar shows the 'Tree' view with 'JDBC Resources' selected. The main content area has the following fields:

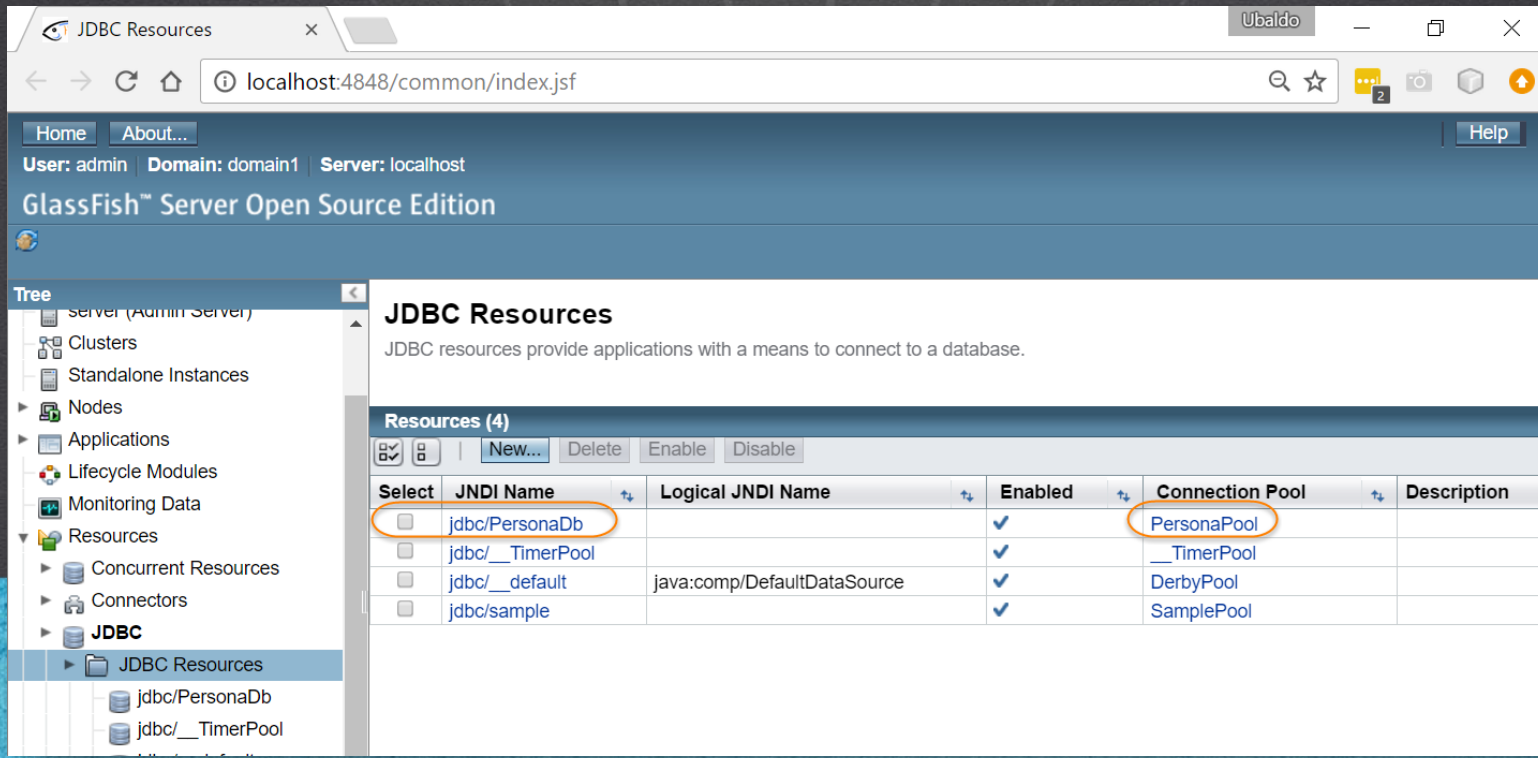
- JNDI Name:** jdbc/PersonaDb
- Pool Name:** PersonaPool
- Description:** (empty)
- Status:** ☒ Enabled

A red arrow points to the 'OK' button. Below the main fields is a section for 'Additional Properties (0)' with a table that is currently empty.

| Select | Name | Value | Description |
|-----------------|------|-------|-------------|
| No items found. | | | |

PASO 2. CONFIGURACION CONEXIÓN JTA

Con esto ya tenemos la conexión de JDBC llamada **jdbc/PersonaDb** y el pool de conexiones de MySql y podemos utilizarlo para conectarnos desde nuestra aplicación de Java vía JTA (Java Transaction API).



The screenshot shows the GlassFish Admin Console interface. The browser address bar indicates the URL is `localhost:4848/common/index.jsf`. The page title is "JDBC Resources". The left sidebar shows a tree view with "JDBC Resources" selected. The main content area displays a table of JDBC resources. The table has columns: Select, JNDI Name, Logical JNDI Name, Enabled, Connection Pool, and Description. Four resources are listed: **jdbc/PersonaDb** (highlighted with a red circle), **jdbc/__TimerPool**, **jdbc/__default**, and **jdbc/sample**. The **PersonaPool** connection pool is also highlighted with a red circle. The **PersonaDb** resource is enabled and uses the **PersonaPool** connection pool.

| Select | JNDI Name | Logical JNDI Name | Enabled | Connection Pool | Description |
|-------------------------------------|-----------------------|-----------------------------|---------|--------------------|-------------|
| <input checked="" type="checkbox"/> | jdbc/PersonaDb | | ✓ | PersonaPool | |
| <input type="checkbox"/> | jdbc/__TimerPool | | ✓ | __TimerPool | |
| <input type="checkbox"/> | jdbc/__default | java:comp/DefaultDataSource | ✓ | DerbyPool | |
| <input type="checkbox"/> | jdbc/sample | | ✓ | SamplePool | |

PASO 2. CONFIGURACION CONEXIÓN JTA

Ejemplo de Uso: Podemos observar que el mismo nombre configurado en Glassfish, es el nombre usado en el archivo persistence.xml:

| Resources (4) | | | | | |
|--|------------------|-----------------------------|---------|-----------------|--|
| <div><input checked="" type="checkbox"/> <input type="checkbox"/> <input type="button" value="New..."/> <input type="button" value="Delete"/> <input type="button" value="Enable"/> <input type="button" value="Disable"/></div> | | | | | |
| Select | JNDI Name | Logical JNDI Name | Enabled | Connection Pool | |
| <input checked="" type="checkbox"/> | jdbc/PersonaDb | | ✓ | PersonaPool | |
| <input type="checkbox"/> | jdbc/__TimerPool | | ✓ | __TimerPool | |
| <input type="checkbox"/> | jdbc/__default | java:comp/DefaultDataSource | ✓ | DerbyPool | |
| <input type="checkbox"/> | jdbc/sample | | ✓ | SamplePool | |

persistence.xml x

Source History

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <persistence version="2.1" xmlns="http://xmlns.jcp.org/xml/ns/persistence" xmlns:xsi=
3   <persistence-unit name="PersonaPU" transaction-type="JTA">
4     <jta-data-source jdbc/PersonaDb</jta-data-source>
5   </persistence-unit>
6 </persistence>
```


CONCLUSIÓN DEL EJERCICIO

- Con este ejercicio hemos creado un pool de conexiones y creamos un recurso jdbc que será el que podamos utilizar desde nuestras aplicaciones Java, por ejemplo con JPA en su archivo persistence.xml
- De esta manera podemos comunicarnos con la base de datos de MySQL, además de acceder vía JTA (Java Transaction API), el cual nos permite delegar los datos de conexión a Glassfish y así evitar configurar la conexión a base de datos desde nuestra aplicación.



Experiencia y Conocimiento para tu vida

CURSO JAVA

www.globalmentoring.com.mx

CURSO ONLINE

CURSO DE JAVA

Por: Ing. Ubaldo Acosta



Experiencia y Conocimiento para tu vida

CURSO JAVA

www.globalmentoring.com.mx