

Name add-resources– creates the resources specified in an XML file

Synopsis add-resources [--help] [--target *target*] *xml_file_path*

Description The add-resources subcommand creates the resources named in the specified XML file. The resources that can be created with this subcommand are listed in See Also in this help page.

The *xml_file_path* is the path to the XML file that contains the resources to be created. The DOCTYPE must be specified as http://www.sun.com/software/dtd/appserver/sun-resources_1_3.dtd in the resources.xml file.

This subcommand is supported in remote mode only.

Options --help
Displays the help text for the subcommand.

--target
Do not specify this option. This option is retained for compatibility with other releases. If you specify this option, a syntax error does not occur. Instead, the subcommand runs successfully and the option is silently ignored.

Operands *xml_file_path*
The path to the XML file that contains the resource(s) to be created. The XML file must reside in the *install-dir*/domains/domain1/config directory. If you specify a relative path or simply provide the name of the XML file, this subcommand will prepend *install-dir*/domains/domain1/config to this operand.

An example XML file follows.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE resources PUBLIC
    "-//Sun Microsystems Inc./DTD Application Server 9.1 Domain//EN"
    "*http://www.sun.com/software/appserver/dtds/sun-resources_1_2.dtd*">
<resources>
  <jdbc-connection-pool name="SPECjPool" steady-pool-size="100"
    max-pool-size="150" max-wait-time-in-millis="60000"
    pool-resize-quantity="2" idle-timeout-in-seconds="300"
    is-isolation-level-guaranteed="true"
    is-connection-validation-required="false"
    connection-validation-method="auto-commit"
    fail-all-connections="false"
    datasource-classname="oracle.jdbc.pool.OracleDataSource">
    <property name="URL"
      value="jdbc:oracle:thin:@iasperfsol12:1521:specdb"/>
    <property name="User" value="spec"/>
    <property name="Password" value="spec"/>
    <property name="MaxStatements" value="200"/>
    <property name="ImplicitCachingEnabled" value="true"/>
  </jdbc-connection-pool>
  <jdbc-resource enabled="true" pool-name="SPECjPool">
```

```
jndi-name="jdbc/SPECjDB"/>
</resources>
```

Examples **EXAMPLE 1** Adding Resources

This example creates resources using the contents of the XML file `resource.xml`:

```
asadmin> add-resources resource.xml
Command : Connector connection pool jms/testQFactoryPool created.
Command : Administered object jms/testQ created.
Command : Connector resource jms/testQFactory created.
Command : Resource adapter config myResAdapterConfig created successfully
Command : JDBC connection pool DerbyPoolA created successfully.
Command : JDBC resource jdbc/___defaultA created successfully.
Command add-resources executed successfully.
```

Exit Status	0	subcommand executed successfully
	1	error in executing the subcommand

See Also `create-jdbc-connection-pool(1)`, `create-jdbc-resource(1)`, `create-jms-resource(1)`, `create-jndi-resource(1)`, `create-javamail-resource(1)`, `create-custom-resource(1)`, `create-connector-resource(1)`, `create-connector-work-security-map(1)`, `create-admin-object(1)`, `create-resource-adapter-config(1)`

`asadmin(1M)`

