

Name	create-instance – creates a GlassFish Server instance
Synopsis	<pre>create-instance [--help] --node <i>node-name</i> [--config <i>config-name</i> --cluster <i>cluster-name</i>] [--lbenabled={true false}] [--portbase=<i>port-number</i>] [--checkports={true false}] [--systemproperties (<i>name=value</i>)[:<i>name=value</i>]*] <i>instance-name</i></pre>
Description	<p>The create-instance subcommand creates a GlassFish Server instance. This subcommand requires secure shell (SSH) to be configured on the machine where the domain administration server (DAS) is running and on the machine where the instance is to reside. You may run this command from any machine that can contact the DAS.</p> <p>A GlassFish Server instance is a single Virtual Machine for the Java platform (Java Virtual Machine or JVM machine) on a single node in which GlassFish Server is running. A node defines the machine where the GlassFish Server instance resides. The JVM machine must be compatible with the Java Platform, Enterprise Edition (Java EE).</p> <p>A GlassFish Server instance requires a reference to the following items:</p> <ul style="list-style-type: none"> ▪ The node that defines the machine where the instance resides. The node must be specified in the command to create the instance. ▪ The named configuration that defines the configuration of the instance. The configuration can be specified in the command to create the instance, but is not required. If no configuration is specified for an instance that is not joining a cluster, the subcommand creates a configuration for the instance. An instance that is joining a cluster receives its configuration from its parent cluster. <p>Each GlassFish Server instance is one of the following types of instance:</p> <p>Standalone instance</p> <p>A standalone instance does not share its configuration with any other instances or clusters. A standalone instance is created if no configuration or cluster is specified in the command to create the instance.</p> <p>When a standalone instance is created, a copy of the default-config configuration is created for the instance. The name of this configuration is <i>instance-name-config</i>, where <i>instance-name</i> represents the name of an unclustered server instance.</p> <p>Shared instance</p> <p>A shared instance shares its configuration with other instances or clusters. A shared instance is created if a named configuration is specified in the command to create the instance.</p> <p>Clustered instance</p> <p>A clustered instance inherits its configuration from the cluster to which the instance belongs and shares its configuration with other instances in the cluster. A clustered instance is created if a cluster is specified in the command to create the instance.</p>

Any instance that is not part of a cluster is considered an unclustered server instance. Therefore, standalone instances and shared instances are unclustered server instances.

By default, this subcommand attempts to resolve possible port conflicts for the instance that is being created. The subcommand also assigns ports that are currently not in use and not already assigned to other instances on the same node. The subcommand assigns these ports randomly. Use the `--systemproperties` option to resolve port conflicts for additional instances on the same node. System properties of an instance can be manipulated by using the [create-system-properties\(1\)](#) subcommand and the [delete-system-property\(1\)](#) subcommand.

This subcommand is supported in remote mode only.

Options

`--help`

`-?`

Displays the help text for the subcommand.

`--node`

The name of the node that defines the machine where the instance is to be created. The node must already exist. If the instance is to be created on the machine where the domain administration server (DAS) is running, use the predefined node `localhost`.

`--config`

Specifies the named configuration that the instance references. The configuration must exist and must not be named `default-config` or `server-config`. If the configuration name specifies a standalone configuration, an error occurs. Specifying the `--config` option creates a shared instance.

The `--config` option and the `--cluster` option are mutually exclusive. If both options are omitted, a standalone instance is created.

`--cluster`

Specifies the cluster from which the instance inherits its configuration. Specifying the `--cluster` option creates a clustered instance.

The `--config` option and the `--cluster` option are mutually exclusive. If both options are omitted, a standalone instance is created.

`--lbenabled`

Specifies whether the instance is enabled for load balancing. Possible values are as follows:

`true`

The instance is enabled for load balancing (default).

When an instance is enabled for load balancing, a load balancer sends requests to the instance.

`false`

The instance is disabled for load balancing.

When an instance is disabled for load balancing, a load balancer does not send requests to the instance.

--portbase

Determines the number with which the port assignment should start. An instance uses a certain number of ports that are statically assigned. The *portbase* value determines where the assignment should start. The values for the ports are calculated as follows:

- Administration port: *portbase* + 48
- HTTP listener port: *portbase* + 80
- IIOP listener port: *portbase* + 37
- JMX port: *portbase* + 86

When the --portbase option is specified, the output of this subcommand includes a complete list of used ports.

--checkports

Specifies whether to check for the availability of the administration, HTTP, JMS, JMX, and IIOP ports. The default value is `true`.

--systemproperties

Defines system properties for the instance. These properties override property definitions for port settings in the instance's configuration. Predefined port settings must be overridden if, for example, two clustered instances reside on the same machine. In this situation, port settings for one instance must be overridden because both instances share the same configuration.

The following properties are available:

ASADMIN_LISTENER_PORT

This property specifies the port number of the HTTP port or the HTTPS port for administration. This port is the port in the URL that you specify in your web browser to manage the instance, for example, `http://localhost:4949`. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

HTTP_LISTENER_PORT

This property specifies the port number of the port that is used to listen for HTTP requests. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

HTTP_SSL_LISTENER_PORT

This property specifies the port number of the port that is used to listen for HTTPS requests. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

IIOP_LISTENER_PORT

This property specifies the port number of the port that is used for IIOP connections. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

IIOP_SSL_LISTENER_PORT

This property specifies the port number of the port that is used for secure IIOP connections. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

IIOP_SSL_MUTUALAUTH_PORT

This property specifies the port number of the port that is used for secure IIOP connections with client authentication.

JMS_PROVIDER_PORT

This property specifies the port number for the Java Message Service provider. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

JMX_SYSTEM_CONNECTOR_PORT

This property specifies the port number on which the JMX connector listens. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

Operands *instance-name*

The name of the instance that is being created. Each instance in a domain must have a name that is unique across all nodes, GlassFish Server instances, clusters, and named configurations.

Examples **EXAMPLE 1** Creating a Standalone GlassFish Server Instance

This example creates the standalone GlassFish Server instance `pmdsainst` on the local host.

```
asadmin> create-instance --node localhost pmdsainst
Port Assignments for server instance pmdsainst:
JMX_SYSTEM_CONNECTOR_PORT=28688
JMS_PROVIDER_PORT=27678
ASADMIN_LISTENER_PORT=24850
HTTP_LISTENER_PORT=28082
IIOP_LISTENER_PORT=23702
IIOP_SSL_LISTENER_PORT=23822
HTTP_SSL_LISTENER_PORT=28183
IIOP_SSL_MUTUALAUTH_PORT=23922
```

Command `create-instance` executed successfully.

EXAMPLE 2 Creating a Standalone GlassFish Server Instance With Custom Port Assignments

This example creates the standalone GlassFish Server instance `pmdcpinst` on the local host. Custom port numbers are assigned to the following ports:

- HTTP listener port
- HTTPS listener port
- IIOP connections port
- Secure IIOP connections port

EXAMPLE 2 Creating a Standalone GlassFish Server Instance With Custom Port Assignments
(Continued)

- Secure IIOP connections port with mutual authentication
- JMX connector port

```
asadmin> create-instance --node localhost
--systemproperties HTTP_LISTENER_PORT=58294:
HTTP_SSL_LISTENER_PORT=58297:
IIOP_LISTENER_PORT=58300:
IIOP_SSL_LISTENER_PORT=58303:
IIOP_SSL_MUTUALAUTH_PORT=58306:
JMX_SYSTEM_CONNECTOR_PORT=58309 pmdcpinst
Port Assignments for server instance pmdcpinst:
JMS_PROVIDER_PORT=27679
ASADMIN_LISTENER_PORT=24851
```

Command `create-instance` executed successfully.

EXAMPLE 3 Creating a Shared GlassFish Server Instance

This example creates the shared GlassFish Server instance `pmdsharedinst1` on the local host. The shared configuration of this instance is `pmdsharedconfig`.

```
asadmin create-instance --node localhost --config pmdsharedconfig
pmdsharedinst1
Port Assignments for server instance pmdsharedinst1:
JMX_SYSTEM_CONNECTOR_PORT=28687
JMS_PROVIDER_PORT=27677
ASADMIN_LISTENER_PORT=24849
HTTP_LISTENER_PORT=28081
IIOP_LISTENER_PORT=23701
IIOP_SSL_LISTENER_PORT=23821
HTTP_SSL_LISTENER_PORT=28182
IIOP_SSL_MUTUALAUTH_PORT=23921
```

Command `create-instance` executed successfully.

EXAMPLE 4 Creating a Clustered GlassFish Server Instance

This example creates the clustered GlassFish Server instance `pmdinst1` on the local host. The instance is a member of the cluster `pmdclust1`.

```
asadmin> create-instance --node localhost --cluster pmdclust pmdinst1
Port Assignments for server instance pmdinst1:
JMX_SYSTEM_CONNECTOR_PORT=28686
JMS_PROVIDER_PORT=27676
HTTP_LISTENER_PORT=28080
ASADMIN_LISTENER_PORT=24848
IIOP_SSL_LISTENER_PORT=23820
```

EXAMPLE 4 Creating a Clustered GlassFish Server Instance *(Continued)*

```
IIOP_LISTENER_PORT=23700
HTTP_SSL_LISTENER_PORT=28181
IIOP_SSL_MUTUALAUTH_PORT=23920
```

Command `create-instance` executed successfully.

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

See Also [create-local-instance\(1\)](#), [create-node-config\(1\)](#), [create-node-ssh\(1\)](#),
[create-system-properties\(1\)](#), [delete-instance\(1\)](#), [delete-system-property\(1\)](#),
[list-instances\(1\)](#), [setup-ssh\(1\)](#), [start-instance\(1\)](#), [stop-instance\(1\)](#)

[asadmin\(1M\)](#)

