| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/POAOperations.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/omg/PortableServer/POAManagerOperations.html)   [**NEXT CLASS**](http://docs.google.com/org/omg/PortableServer/REQUEST_PROCESSING_POLICY_ID.html) | [**FRAMES**](http://docs.google.com/index.html?org/omg/PortableServer/POAOperations.html)    [**NO FRAMES**](http://docs.google.com/POAOperations.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **org.omg.PortableServer**

Interface POAOperations

**All Known Subinterfaces:** [POA](http://docs.google.com/org/omg/PortableServer/POA.html)

public interface **POAOperations**

A POA object manages the implementation of a collection of objects. The POA supports a name space for the objects, which are identified by Object Ids. A POA also provides a name space for POAs. A POA is created as a child of an existing POA, which forms a hierarchy starting with the root POA. A POA object must not be exported to other processes, or externalized with ORB::object\_to\_string.

| **Method Summary** | |
| --- | --- |
| void | [**activate\_object\_with\_id**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#activate_object_with_id(byte%5B%5D,%20org.omg.PortableServer.Servant))(byte[] id, [Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)            This operation enters an association between the specified Object Id and the specified servant in the Active Object Map. |
| byte[] | [**activate\_object**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#activate_object(org.omg.PortableServer.Servant))([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)            This operation generates an Object Id and enters the Object Id and the specified servant in the Active Object Map. |
| [IdAssignmentPolicy](http://docs.google.com/org/omg/PortableServer/IdAssignmentPolicy.html) | [**create\_id\_assignment\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_id_assignment_policy(org.omg.PortableServer.IdAssignmentPolicyValue))([IdAssignmentPolicyValue](http://docs.google.com/org/omg/PortableServer/IdAssignmentPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| [IdUniquenessPolicy](http://docs.google.com/org/omg/PortableServer/IdUniquenessPolicy.html) | [**create\_id\_uniqueness\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_id_uniqueness_policy(org.omg.PortableServer.IdUniquenessPolicyValue))([IdUniquenessPolicyValue](http://docs.google.com/org/omg/PortableServer/IdUniquenessPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| [ImplicitActivationPolicy](http://docs.google.com/org/omg/PortableServer/ImplicitActivationPolicy.html) | [**create\_implicit\_activation\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_implicit_activation_policy(org.omg.PortableServer.ImplicitActivationPolicyValue))([ImplicitActivationPolicyValue](http://docs.google.com/org/omg/PortableServer/ImplicitActivationPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| [LifespanPolicy](http://docs.google.com/org/omg/PortableServer/LifespanPolicy.html) | [**create\_lifespan\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_lifespan_policy(org.omg.PortableServer.LifespanPolicyValue))([LifespanPolicyValue](http://docs.google.com/org/omg/PortableServer/LifespanPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| [POA](http://docs.google.com/org/omg/PortableServer/POA.html) | [**create\_POA**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_POA(java.lang.String,%20org.omg.PortableServer.POAManager,%20org.omg.CORBA.Policy%5B%5D))([String](http://docs.google.com/java/lang/String.html) adapter\_name, [POAManager](http://docs.google.com/org/omg/PortableServer/POAManager.html) a\_POAManager, [Policy](http://docs.google.com/org/omg/CORBA/Policy.html)[] policies)            This operation creates a new POA as a child of the target POA. |
| [Object](http://docs.google.com/org/omg/CORBA/Object.html) | [**create\_reference\_with\_id**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_reference_with_id(byte%5B%5D,%20java.lang.String))(byte[] oid, [String](http://docs.google.com/java/lang/String.html) intf)            This operation creates an object reference that encapsulates the specified Object Id and interface repository Id values. |
| [Object](http://docs.google.com/org/omg/CORBA/Object.html) | [**create\_reference**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_reference(java.lang.String))([String](http://docs.google.com/java/lang/String.html) intf)            This operation creates an object reference that encapsulates a POA-generated Object Id value and the specified interface repository id. |
| [RequestProcessingPolicy](http://docs.google.com/org/omg/PortableServer/RequestProcessingPolicy.html) | [**create\_request\_processing\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_request_processing_policy(org.omg.PortableServer.RequestProcessingPolicyValue))([RequestProcessingPolicyValue](http://docs.google.com/org/omg/PortableServer/RequestProcessingPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| [ServantRetentionPolicy](http://docs.google.com/org/omg/PortableServer/ServantRetentionPolicy.html) | [**create\_servant\_retention\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_servant_retention_policy(org.omg.PortableServer.ServantRetentionPolicyValue))([ServantRetentionPolicyValue](http://docs.google.com/org/omg/PortableServer/ServantRetentionPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| [ThreadPolicy](http://docs.google.com/org/omg/PortableServer/ThreadPolicy.html) | [**create\_thread\_policy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#create_thread_policy(org.omg.PortableServer.ThreadPolicyValue))([ThreadPolicyValue](http://docs.google.com/org/omg/PortableServer/ThreadPolicyValue.html) value)            These operations each return a reference to a policy object with the specified value. |
| void | [**deactivate\_object**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#deactivate_object(byte%5B%5D))(byte[] oid)            This operation causes the ObjectId specified in the oid parameter to be deactivated. |
| void | [**destroy**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#destroy(boolean,%20boolean))(boolean etherealize\_objects, boolean wait\_for\_completion)            This operation destroys the POA and all descendant POAs. |
| [POA](http://docs.google.com/org/omg/PortableServer/POA.html) | [**find\_POA**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#find_POA(java.lang.String,%20boolean))([String](http://docs.google.com/java/lang/String.html) adapter\_name, boolean activate\_it)            If the target POA is the parent of a child POA with the specified name (relative to the target POA), that child POA is returned. |
| [ServantManager](http://docs.google.com/org/omg/PortableServer/ServantManager.html) | [**get\_servant\_manager**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#get_servant_manager())()            If the ServantRetentionPolicy of the POA is RETAIN, then the ServantManager argument (imgr) shall support the ServantActivator interface. |
| [Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) | [**get\_servant**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#get_servant())()            This operation returns the default servant associated with the POA. |
| [Object](http://docs.google.com/org/omg/CORBA/Object.html) | [**id\_to\_reference**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#id_to_reference(byte%5B%5D))(byte[] oid)            If an object with the specified Object Id value is currently active, a reference encapsulating the information used to activate the object is returned. |
| [Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) | [**id\_to\_servant**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#id_to_servant(byte%5B%5D))(byte[] oid)            If the POA has the RETAIN policy and the specified ObjectId is in the Active Object Map, this operation returns the servant associated with that object in the Active Object Map. |
| byte[] | [**id**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#id())()            This returns the unique id of the POA in the process in which it is created. |
| byte[] | [**reference\_to\_id**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#reference_to_id(org.omg.CORBA.Object))([Object](http://docs.google.com/org/omg/CORBA/Object.html) reference)            This operation returns the Object Id value encapsulated by the specified reference. |
| [Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) | [**reference\_to\_servant**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#reference_to_servant(org.omg.CORBA.Object))([Object](http://docs.google.com/org/omg/CORBA/Object.html) reference)            If the POA has the RETAIN policy and the specified object is present in the Active Object Map, this operation returns the servant associated with that object in the Active Object Map. |
| byte[] | [**servant\_to\_id**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#servant_to_id(org.omg.PortableServer.Servant))([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)            This operation has four possible behaviors. |
| [Object](http://docs.google.com/org/omg/CORBA/Object.html) | [**servant\_to\_reference**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#servant_to_reference(org.omg.PortableServer.Servant))([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)            This operation requires the RETAIN policy and either the UNIQUE\_ID or IMPLICIT\_ACTIVATION policies if invoked outside the context of an operation dispatched by this POA. |
| void | [**set\_servant\_manager**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#set_servant_manager(org.omg.PortableServer.ServantManager))([ServantManager](http://docs.google.com/org/omg/PortableServer/ServantManager.html) imgr)            This operation sets the default servant manager associated with the POA. |
| void | [**set\_servant**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#set_servant(org.omg.PortableServer.Servant))([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)            This operation registers the specified servant with the POA as the default servant. |
| [AdapterActivator](http://docs.google.com/org/omg/PortableServer/AdapterActivator.html) | [**the\_activator**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#the_activator())()            This attribute identifies the adapter activator associated with the POA. |
| void | [**the\_activator**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#the_activator(org.omg.PortableServer.AdapterActivator))([AdapterActivator](http://docs.google.com/org/omg/PortableServer/AdapterActivator.html) newThe\_activator)            This attribute identifies the adapter activator associated with the POA. |
| [POA](http://docs.google.com/org/omg/PortableServer/POA.html)[] | [**the\_children**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#the_children())()            This attribute identifies the current set of all child POAs of the POA. |
| [String](http://docs.google.com/java/lang/String.html) | [**the\_name**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#the_name())()            This attribute identifies the POA relative to its parent. |
| [POA](http://docs.google.com/org/omg/PortableServer/POA.html) | [**the\_parent**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#the_parent())()            This attribute identifies the parent of the POA. |
| [POAManager](http://docs.google.com/org/omg/PortableServer/POAManager.html) | [**the\_POAManager**](http://docs.google.com/org/omg/PortableServer/POAOperations.html#the_POAManager())()            This attribute identifies the POA manager associated with the POA. |

| **Method Detail** |
| --- |

### create\_POA

[POA](http://docs.google.com/org/omg/PortableServer/POA.html) **create\_POA**([String](http://docs.google.com/java/lang/String.html) adapter\_name,  
 [POAManager](http://docs.google.com/org/omg/PortableServer/POAManager.html) a\_POAManager,  
 [Policy](http://docs.google.com/org/omg/CORBA/Policy.html)[] policies)  
 throws [AdapterAlreadyExists](http://docs.google.com/org/omg/PortableServer/POAPackage/AdapterAlreadyExists.html),  
 [InvalidPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/InvalidPolicy.html)

This operation creates a new POA as a child of the target POA.

**Parameters:**adapter\_name - identifies the new POA with respect to other POAs with the same parent POA.a\_POAManager - specifies the POA Manager to be associated with the new POA.policies - specifies policy objects to be associated with the POA to control its behavior. **Throws:** AdapterAlreadyExists - specifies that the target POA already has a child POA with the specified name. InvalidPolicy - is raised if any of the policy objects are not valid for the ORB, or are in conflict, or require an administrative action that has not been performed.

### find\_POA

[POA](http://docs.google.com/org/omg/PortableServer/POA.html) **find\_POA**([String](http://docs.google.com/java/lang/String.html) adapter\_name,  
 boolean activate\_it)  
 throws [AdapterNonExistent](http://docs.google.com/org/omg/PortableServer/POAPackage/AdapterNonExistent.html)

If the target POA is the parent of a child POA with the specified name (relative to the target POA), that child POA is returned.

**Parameters:**adapter\_name - POA name to be found.activate\_it - if a POA with the specified name does not exist and the value of the activate\_it parameter is TRUE, the target POA's AdapterActivator, if one exists, is invoked. **Returns:**POA if one exists or is activated by the AdapterActivator. **Throws:** [AdapterNonExistent](http://docs.google.com/org/omg/PortableServer/POAPackage/AdapterNonExistent.html)

### destroy

void **destroy**(boolean etherealize\_objects,  
 boolean wait\_for\_completion)

This operation destroys the POA and all descendant POAs. All descendant POAs are destroyed (recursively) before the destruction of the containing POA. The POA so destroyed (that is, the POA with its name) may be re-created later in the same process.

**Parameters:**etherealize\_objects - flag to indicate whether etherealize operation on servant manager needs to be called.wait\_for\_completion - flag to indicate whether POA and its children need to wait for active requests and the etherealization to complete.

### create\_thread\_policy

[ThreadPolicy](http://docs.google.com/org/omg/PortableServer/ThreadPolicy.html) **create\_thread\_policy**([ThreadPolicyValue](http://docs.google.com/org/omg/PortableServer/ThreadPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**ThreadPolcy Object

### create\_lifespan\_policy

[LifespanPolicy](http://docs.google.com/org/omg/PortableServer/LifespanPolicy.html) **create\_lifespan\_policy**([LifespanPolicyValue](http://docs.google.com/org/omg/PortableServer/LifespanPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**LifespanPolicy Object.

### create\_id\_uniqueness\_policy

[IdUniquenessPolicy](http://docs.google.com/org/omg/PortableServer/IdUniquenessPolicy.html) **create\_id\_uniqueness\_policy**([IdUniquenessPolicyValue](http://docs.google.com/org/omg/PortableServer/IdUniquenessPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**IdUniquenessPolicy Object.

### create\_id\_assignment\_policy

[IdAssignmentPolicy](http://docs.google.com/org/omg/PortableServer/IdAssignmentPolicy.html) **create\_id\_assignment\_policy**([IdAssignmentPolicyValue](http://docs.google.com/org/omg/PortableServer/IdAssignmentPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**IdAssignmentPolicy Object.

### create\_implicit\_activation\_policy

[ImplicitActivationPolicy](http://docs.google.com/org/omg/PortableServer/ImplicitActivationPolicy.html) **create\_implicit\_activation\_policy**([ImplicitActivationPolicyValue](http://docs.google.com/org/omg/PortableServer/ImplicitActivationPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**ImplicitActivationPolicy Object.

### create\_servant\_retention\_policy

[ServantRetentionPolicy](http://docs.google.com/org/omg/PortableServer/ServantRetentionPolicy.html) **create\_servant\_retention\_policy**([ServantRetentionPolicyValue](http://docs.google.com/org/omg/PortableServer/ServantRetentionPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**ServantRetentionPolicy Object.

### create\_request\_processing\_policy

[RequestProcessingPolicy](http://docs.google.com/org/omg/PortableServer/RequestProcessingPolicy.html) **create\_request\_processing\_policy**([RequestProcessingPolicyValue](http://docs.google.com/org/omg/PortableServer/RequestProcessingPolicyValue.html) value)

These operations each return a reference to a policy object with the specified value.

**Parameters:**value - policy type **Returns:**RequestProcessingPolicy Object.

### the\_name

[String](http://docs.google.com/java/lang/String.html) **the\_name**()

This attribute identifies the POA relative to its parent. This name is assigned when the POA is created.

### the\_parent

[POA](http://docs.google.com/org/omg/PortableServer/POA.html) **the\_parent**()

This attribute identifies the parent of the POA. The parent of the root POA is null.

### the\_children

[POA](http://docs.google.com/org/omg/PortableServer/POA.html)[] **the\_children**()

This attribute identifies the current set of all child POAs of the POA. The set of child POAs includes only the POA's immediate children, and not their descendants.

### the\_POAManager

[POAManager](http://docs.google.com/org/omg/PortableServer/POAManager.html) **the\_POAManager**()

This attribute identifies the POA manager associated with the POA.

### the\_activator

[AdapterActivator](http://docs.google.com/org/omg/PortableServer/AdapterActivator.html) **the\_activator**()

This attribute identifies the adapter activator associated with the POA.

### the\_activator

void **the\_activator**([AdapterActivator](http://docs.google.com/org/omg/PortableServer/AdapterActivator.html) newThe\_activator)

This attribute identifies the adapter activator associated with the POA.

### get\_servant\_manager

[ServantManager](http://docs.google.com/org/omg/PortableServer/ServantManager.html) **get\_servant\_manager**()  
 throws [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

If the ServantRetentionPolicy of the POA is RETAIN, then the ServantManager argument (imgr) shall support the ServantActivator interface. For a NON\_RETAIN policy, the ServantManager shall support the ServantLocator interface. If the argument is nil, or does not support the required interface, then the OBJ\_ADAPTER exception is raised.

**Returns:**ServantManager associated with a POA or null if none exists. **Throws:** WrongPolicy - raised if the USE\_SERVANT\_MANAGER policy is not specified.

### set\_servant\_manager

void **set\_servant\_manager**([ServantManager](http://docs.google.com/org/omg/PortableServer/ServantManager.html) imgr)  
 throws [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation sets the default servant manager associated with the POA. This operation may only be invoked once after a POA has been created. Attempting to set the servant manager after one has already been set will result in the BAD\_INV\_ORDER exception being raised.

**Parameters:**imgr - servant manager to be used as a default. **Throws:** WrongPolicy - raised if the USE\_SERVANT\_MANAGER policy is not specified.

### get\_servant

[Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) **get\_servant**()  
 throws [NoServant](http://docs.google.com/org/omg/PortableServer/POAPackage/NoServant.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation returns the default servant associated with the POA.

**Returns:**p\_servant default servant associated with a POA. **Throws:** NoServant - raised if no default servant is associated with the POA. WrongPolicy - raised if the USE\_DEFAULT\_SERVANT policy is not specified.

### set\_servant

void **set\_servant**([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)  
 throws [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation registers the specified servant with the POA as the default servant. This servant will be used for all requests for which no servant is found in the Active Object Map.

**Parameters:**p\_servant - servant to be used as a default. **Throws:** WrongPolicy - raised if the USE\_DEFAULT\_SERVANT policy is not specified.

### activate\_object

byte[] **activate\_object**([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)  
 throws [ServantAlreadyActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ServantAlreadyActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation generates an Object Id and enters the Object Id and the specified servant in the Active Object Map.

**Parameters:**p\_servant - servant to be associated with an object to be activated. **Returns:**POA generated object id. **Throws:** ServantAlreadyActive - is raised if the POA has UNIQUE\_ID policy and servant is is already in the Active Object Map. WrongPolicy - raised if the SYSTEM\_ID and RETAIN policies are not specified.

### activate\_object\_with\_id

void **activate\_object\_with\_id**(byte[] id,  
 [Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)  
 throws [ServantAlreadyActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ServantAlreadyActive.html),  
 [ObjectAlreadyActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ObjectAlreadyActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation enters an association between the specified Object Id and the specified servant in the Active Object Map.

**Parameters:**id - object id for the object to be activated.p\_servant - servant to be associated with the object. **Throws:** ServantAlreadyActive - raised if the POA has the UNIQUE\_ID policy and the servant is already in the Active Object Map. ObjectAlreadyActive - raised if the object is already active in the POA. WrongPolicy - raised if the RETAIN policy is is not specified.

### deactivate\_object

void **deactivate\_object**(byte[] oid)  
 throws [ObjectNotActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ObjectNotActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation causes the ObjectId specified in the oid parameter to be deactivated. An ObjectId which has been deactivated continues to process requests until there are no active requests for that ObjectId. A deactivated ObjectId is removed from the Active Object Map when all requests executing for that ObjectId have completed.

**Parameters:**oid - Object Id for the object to be deactivated. **Throws:** ObjectNotActive - if the object with the specified oid is not in the Active Object Map. WrongPolicy - raised if the RETAIN policy is is not specified.

### create\_reference

[Object](http://docs.google.com/org/omg/CORBA/Object.html) **create\_reference**([String](http://docs.google.com/java/lang/String.html) intf)  
 throws [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation creates an object reference that encapsulates a POA-generated Object Id value and the specified interface repository id.

**Parameters:**intf - rep id for creating an object reference. **Returns:**object reference created using intf. **Throws:** WrongPolicy - if SYSTEM\_ID policy is not specified.

### create\_reference\_with\_id

[Object](http://docs.google.com/org/omg/CORBA/Object.html) **create\_reference\_with\_id**(byte[] oid,  
 [String](http://docs.google.com/java/lang/String.html) intf)

This operation creates an object reference that encapsulates the specified Object Id and interface repository Id values. It does not cause an activation to take place. The resulting reference may be passed to clients, so that subsequent requests on those references will cause the object to be activated if necessary, or the default servant used, depending on the applicable policies.

**Parameters:**oid - object id for creating an objrefintf - rep id for creating an objref **Returns:**object reference created using oid and intf **Throws:** BAD\_PARAM - is raised if the POA has the SYSTEM\_ID policy and it detects that the Object Id value was not generated by the system or for this POA.

### servant\_to\_id

byte[] **servant\_to\_id**([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)  
 throws [ServantNotActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ServantNotActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation has four possible behaviors. 1. If the POA has the UNIQUE\_ID policy and the specified servant is active, the Object Id associated with that servant is returned. 2. If the POA has the IMPLICIT\_ACTIVATION policy and either the POA has the MULTIPLE\_ID policy or the specified servant is not active, the servant is activated using a POA-generated Object Id and the Interface Id associated with the servant, and that Object Id is returned. 3. If the POA has the USE\_DEFAULT\_SERVANT policy, the servant specified is the default servant, and the operation is being invoked in the context of executing a request on the default servant, then the ObjectId associated with the current invocation is returned. 4. Otherwise, the ServantNotActive exception is raised.

**Parameters:**p\_servant - servant for which the object disi returned. **Returns:**object id associated with the servant. **Throws:** ServantNotActive - if the above rules and policy combination is not met. WrongPolicy - if the USE\_DEFAULT\_SERVANT policy or a combination of the RETAIN policy and either the UNIQUE\_ID or IMPLICIT\_ACTIVATION policies are not present.

### servant\_to\_reference

[Object](http://docs.google.com/org/omg/CORBA/Object.html) **servant\_to\_reference**([Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) p\_servant)  
 throws [ServantNotActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ServantNotActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation requires the RETAIN policy and either the UNIQUE\_ID or IMPLICIT\_ACTIVATION policies if invoked outside the context of an operation dispatched by this POA. It has four possible behaviors. 1. If the POA has both the RETAIN and the UNIQUE\_ID policy and the specified servant is active, an object reference encapsulating the information used to activate the servant is returned. 2. If the POA has both the RETAIN and the IMPLICIT\_ACTIVATION policy and either the POA has the MULTIPLE\_ID policy or the specified servant is not active, the servant is activated using a POA-generated Object Id and the Interface Id associated with the servant, and a corresponding object reference is returned. 3. If the operation was invoked in the context of executing a request on the specified servant, the reference associated with the current invocation is returned. 4. Otherwise, the ServantNotActive exception is raised.

**Parameters:**p\_servant - servant for which the object reference needs to be obtained. **Returns:**object reference associated with the servant. **Throws:** WrongPolicy - if the operation is not invoked in the context of executing a request on the specified servant and the required policies are not present. ServantNotActive - if the above specified policies and rules are not met.

### reference\_to\_servant

[Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) **reference\_to\_servant**([Object](http://docs.google.com/org/omg/CORBA/Object.html) reference)  
 throws [ObjectNotActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ObjectNotActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html),  
 [WrongAdapter](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongAdapter.html)

If the POA has the RETAIN policy and the specified object is present in the Active Object Map, this operation returns the servant associated with that object in the Active Object Map. Otherwise, if the POA has the USE\_DEFAULT\_SERVANT policy and a default servant has been registered with the POA, this operation returns the default servant. If the object reference was not created by this POA, the WrongAdapter exception is raised. (OMG Issue on inconsistency with the POA.IDL.

**Parameters:**reference - object reference for which the servant is returned. **Returns:**servant associated with the reference. **Throws:** WrongPolicy - if neither the RETAIN policy or the USE\_DEFAULT\_SERVANT policy is present. ObjectNotActive - if the servant is not present in the Active Object Map (for RETAIN) or no default servant is registered (for USE\_DEFAULT\_POLICY). WrongAdapter - if reference was not created by this POA instance.

### reference\_to\_id

byte[] **reference\_to\_id**([Object](http://docs.google.com/org/omg/CORBA/Object.html) reference)  
 throws [WrongAdapter](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongAdapter.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

This operation returns the Object Id value encapsulated by the specified reference. This operation is valid only if the reference was created by the POA on which the operation is being performed. The object denoted by the reference does not have to be active for this operation to succeed.

**Parameters:**reference - the object reference from which the object id needs to be returned. **Returns:**object id encapsulated in the reference. **Throws:** WrongAdapter - if the reference was not created by the POA specified in the reference. WrongPolicy - declared to allow future extensions.

### id\_to\_servant

[Servant](http://docs.google.com/org/omg/PortableServer/Servant.html) **id\_to\_servant**(byte[] oid)  
 throws [ObjectNotActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ObjectNotActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

If the POA has the RETAIN policy and the specified ObjectId is in the Active Object Map, this operation returns the servant associated with that object in the Active Object Map. Otherwise, if the POA has the USE\_DEFAULT\_SERVANT policy and a default servant has been registered with the POA, this operation returns the default servant.

**Parameters:**oid - object id for the which the servant is returned. **Returns:**servant associated with oid. **Throws:** ObjectNotActive - is raised if ObjectId is is not in the Active Object Map (for RETAIN policy), or no default servant is registered (for USE\_DEFAULT\_SERVANT policy). WrongPolicy - is raised if the RETAIN policy or the USE\_DEFAULT\_SERVANT policy is not present.

### id\_to\_reference

[Object](http://docs.google.com/org/omg/CORBA/Object.html) **id\_to\_reference**(byte[] oid)  
 throws [ObjectNotActive](http://docs.google.com/org/omg/PortableServer/POAPackage/ObjectNotActive.html),  
 [WrongPolicy](http://docs.google.com/org/omg/PortableServer/POAPackage/WrongPolicy.html)

If an object with the specified Object Id value is currently active, a reference encapsulating the information used to activate the object is returned.

**Parameters:**oid - id of the object for which the reference is returned. **Returns:**the object reference **Throws:** ObjectNotActive - if the Object Id value is not active in the POA. WrongPolicy - if the RETAIN policy is not present.

### id

byte[] **id**()

This returns the unique id of the POA in the process in which it is created. It is for use by portable interceptors.

This id is guaranteed unique for the life span of the POA in the process. For persistent POAs, this means that if a POA is created in the same path with the same name as another POA, these POAs are identical and, therefore, have the same id. For transient POAs, each POA is unique.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/POAOperations.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/omg/PortableServer/POAManagerOperations.html)   [**NEXT CLASS**](http://docs.google.com/org/omg/PortableServer/REQUEST_PROCESSING_POLICY_ID.html) | [**FRAMES**](http://docs.google.com/index.html?org/omg/PortableServer/POAOperations.html)    [**NO FRAMES**](http://docs.google.com/POAOperations.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).