

# Checking element editing permissions

It is a programmer responsibility to check model modification permissions before doing model changes. Some Elements in model can be read-only, but API **does not prevent** modifications of these elements.

The Element is read-only if:

- The Element belongs to the read-only used project (module)
- The Element is not locked for editing in a teamwork environment
- Some other restrictions are added in some specific modeling case

## Checking editing permission

Use `com.nomagic.magicdraw.uml.BaseElement.isEditable()` to check if Element's properties can be changed.

Editing permissions should be checked if primitive Element properties are modified (like a name, for example) or references are modified (like a Property type).

Check the adding permission if you are going to modify some containment property (like *Element.ownedElement*).

## Checking adding permission

Use `BaseElement.canAddChild()` to check if children can be added into the Element.

Use `BaseElement.canAdd(com.nomagic.magicdraw.uml.BaseElement)` to check if a parent can own a given element and a parent has permissions to own. This method will fail, for example, if you will try to add an Operation into a Package.

## Checking removing permission

Use `BaseElement.isEditable()` to check if an Element can be removed (disposed) from the model.

## Checking moving permission

The moving permission is similar to the adding permission, but checks more conditions.

## Plugging custom permissions handler

Use `com.nomagic.magicdraw.uml.permissions.ElementPermissionsManager.addPermissionsHandler(ElementPermissions)` to register a custom element permissions handler. Using this API you can handle custom rules for putting a permission on the top of some specific elements.