

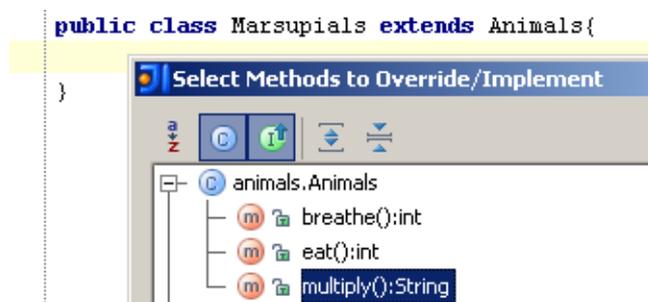
Overriding Methods of a Superclass

You can override any method of a parent class, using the code generation facility. IntelliJ IDEA creates a stub that contains a call to the method of the superclass, leaving the developer with the task of providing some meaningful source code.

To override methods of a superclass

1. With the class in question having the focus, invoke the **Override Method** command in one of the following ways:
 - Press **Ctrl+O**.
 - On the main menu, choose **Code | Generate**.
 - Right-click the editor and choose **Generate** on the context menu.
 - Press **Alt+Insert**.
2. Choose **Override Methods** on the pop-up menu.

The **Select Methods To Override** dialog box displays the list of methods that can be overridden. The list does not include the methods that are already overridden, or cannot be accessed from the current subclass.



3. Select one or more methods to override.
4. If necessary, select the following options:
 - **Insert @Override** to add the `@Override` annotation.
 - **Copy JavaDoc** to insert JavaDoc comments from the overridden methods (if any).
5. Having generated the overriding method, create the required source code. Note the  icon that marks the overriding method in the left gutter. Use this icon to view the name of the base class, and [navigate to the overridden method](#).

```
@Override
public String multiply() {
    return super.multiply();
}
```

File template responsible for overriding a method (*Overridden method body*) accepts predefined template variables from "File Header" (**File | Settings - IDE Settings - File Templates - Code - File Header**), e.g. `${USER}`, `${DATE}`, etc.

For example, consider the following file template:

```
#if ( $RETURN_TYPE != "void" )return $DEFAULT_RETURN_VALUE;#end // TODO ($USER, $DATE):To change
```

Provided that an overridden class contains two methods, this template expands into the following code:

```
public void breathe() {
// TODO (mio, 9/21/12): To change body of implemented methods use File | Settings | File Template
}
public void eat() {
// TODO (mio, 9/21/12): To change body of implemented methods use File | Settings | File Template
}
```

See Also

Procedures:

- [Implementing Methods of an Interface](#)
- [Creating Listeners](#)

Reference:

- [File and Code Templates](#)

Web Resources:

- [Developer Community](#) 