



# Getting Started With Protégé

---

This tutorial gives a brief introduction to creating, viewing, and saving a Protégé-2000 project. You will create a project called "tutorial," which contains a few of the classes and slots from the newspaper example that accompanies Protégé. The tutorial covers the following:

- [Creating a project](#)
- [Saving and naming a project](#)
- [Creating classes](#)
- [Creating slots](#)
- [Entering instances](#)
- [Customizing a form](#)
- [Creating and saving a query](#)

Once you have read the tutorial, you will be ready to explore Protégé on your own.

---

For documentation of the Protégé-2000 features, see the [Protege-2000 Documentation Home Page](#).

For specialized tutorials, see the [Tutorial Home Page](#).



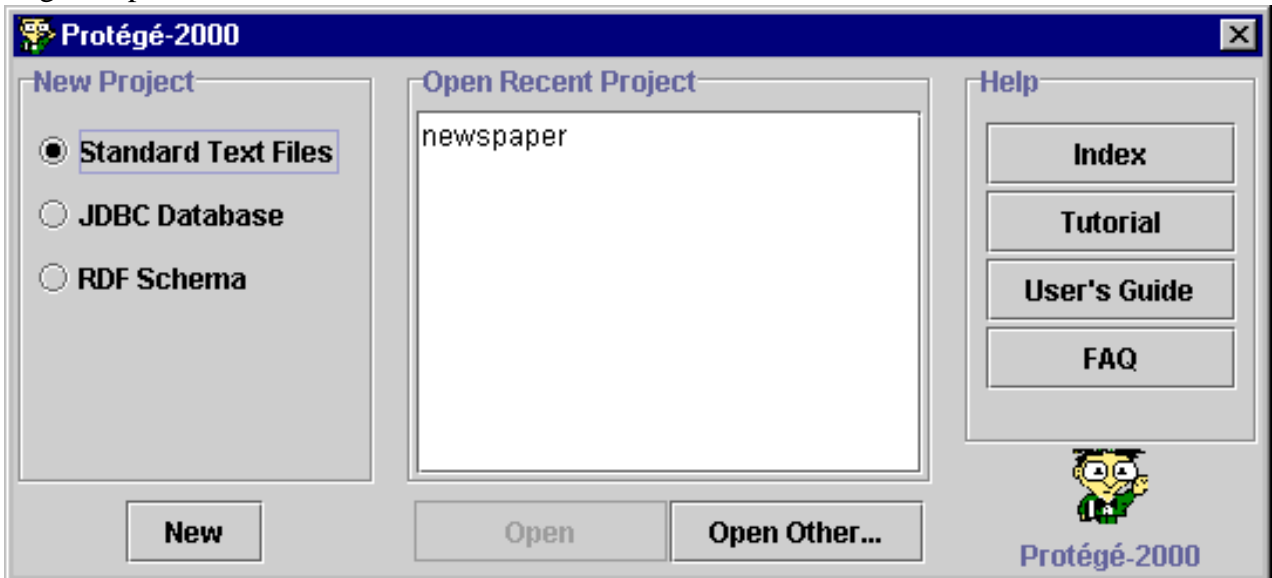
# Creating the Sample Project

First, you must create a new project in Protégé format.

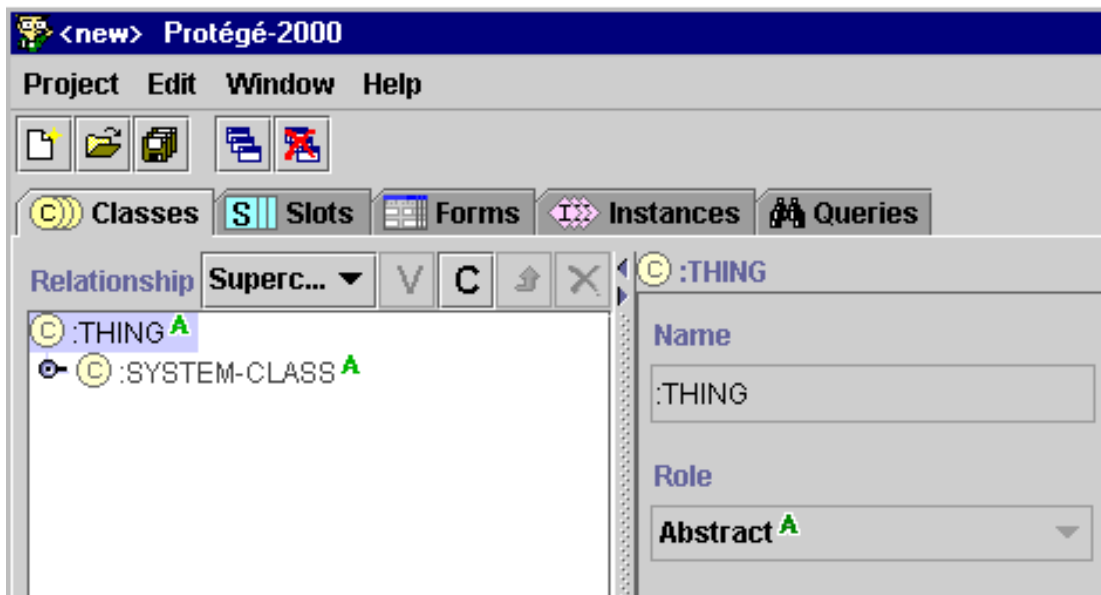
To create the sample project when you start Protégé:

1. Start Protégé. If you have a Protégé project already open, simply exit and restart the program. (For information on creating a project from within Protégé, see [Creating a Project](#).)

When you start Protégé, a dialog box prompts you to create a new project, open a recent project, or get help.



2. In the New Project field at the left, make sure **Standard Text Files** is selected. This will create a project file in Protégé-2000 format.
3. Click **New**.  
The Protégé window opens and the standard tabs become visible. A new project always opens at the Classes view. The internal Protégé classes THING and SYSTEM-CLASS are all that is visible. No instances will be created.



You are now ready to save and name your Protégé project.

---

Next: [Saving and Naming the Project](#)


[Getting Started With Protégé](#)

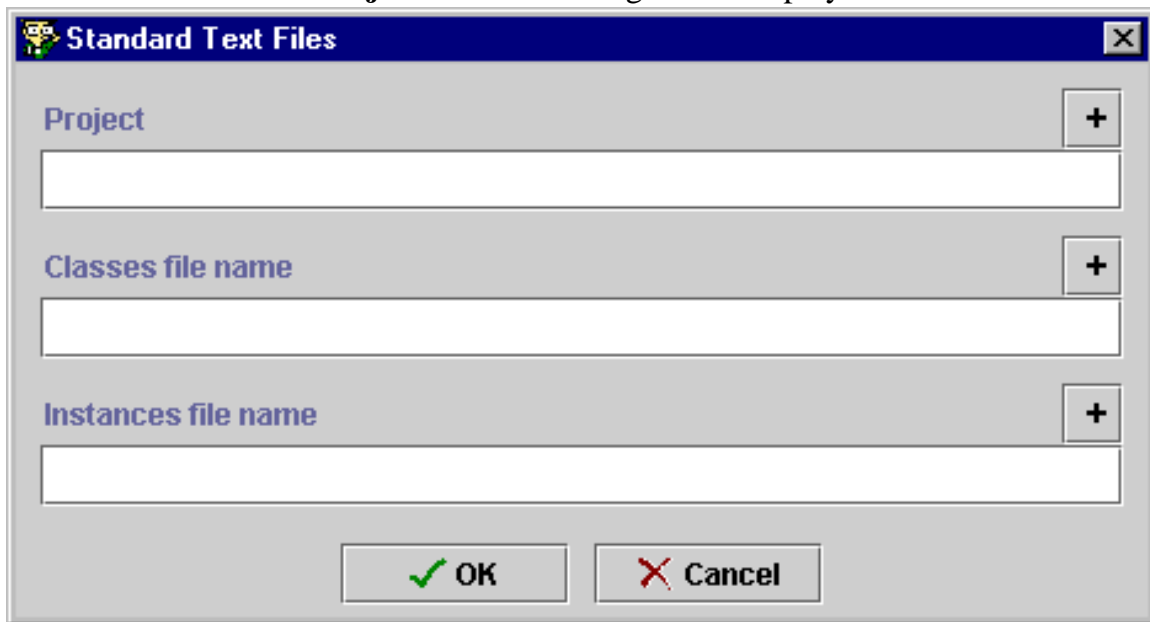


# Creating the Sample Project

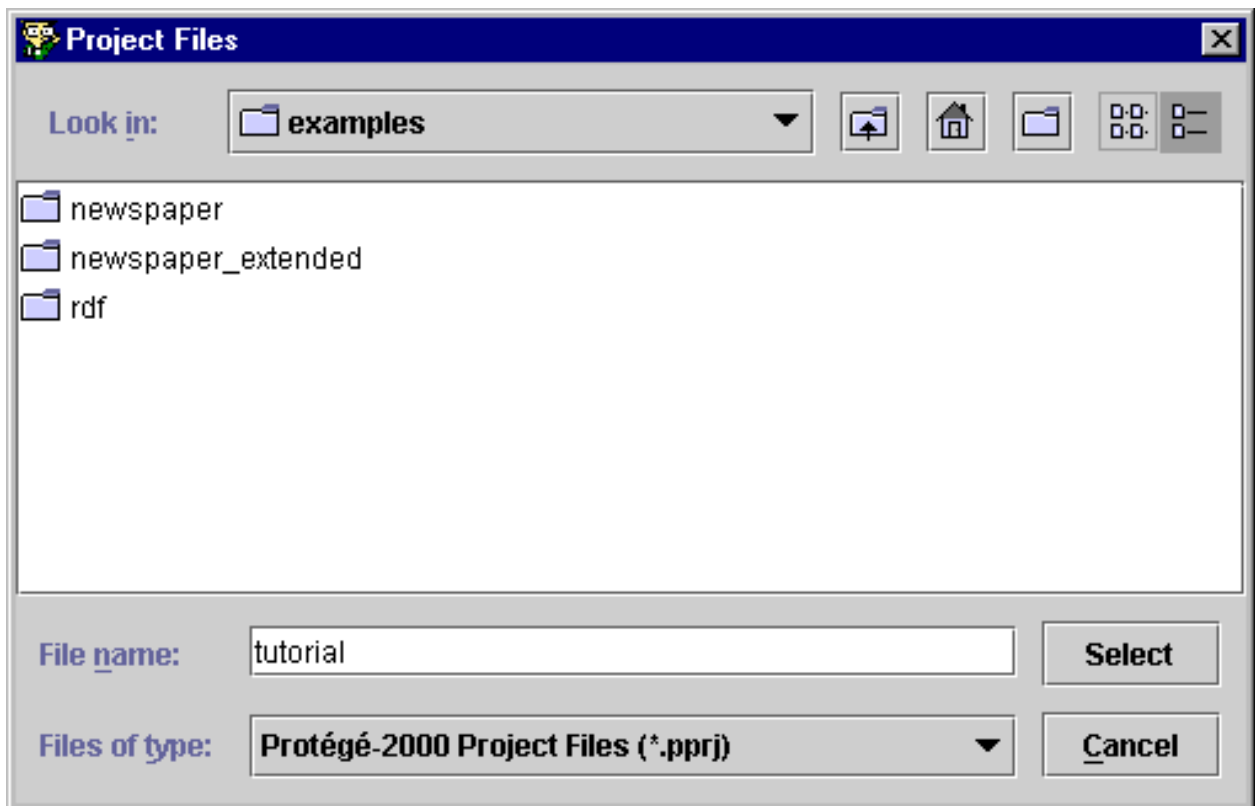
Although the tutorial is not very long, you may find you want to exit the project before you have finished. If you save and name it now, you will be ready to exit whenever is convenient.

To save the project:

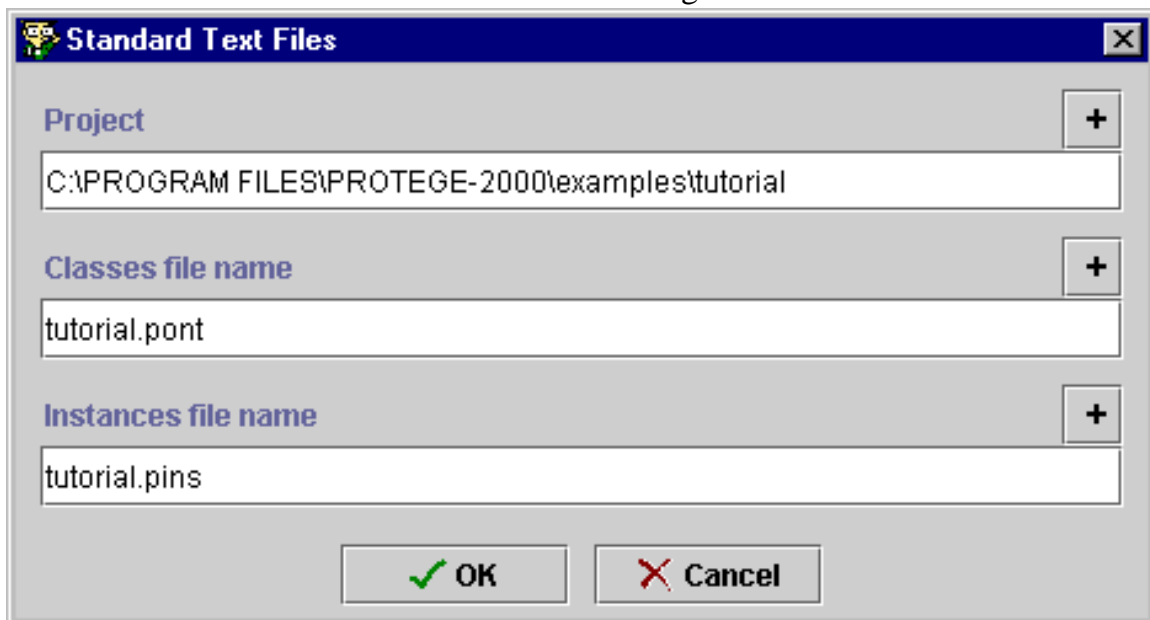
1. Click the Save Project button  at the top left of the Protégé-2000 window. You can also choose **Save** from the **Project** menu. A dialog box is displayed.



2. To choose the location where you want the project to be saved, click the top **+** button at the right of the Project line. Then double-click on the examples folder to open it.



3. Type the name for the project, tutorial, in the **File name** entry field.
4. Click Select.
5. You are returned to the Standard Text Files dialog box. Click OK to save.



Note: You can also choose a location by typing the full path name in the **Project** line. The names of the other files will be filled in automatically.

---

Next: [Creating and Naming Classes](#)

[Getting Started With Protégé](#)



# Creating and Naming Classes

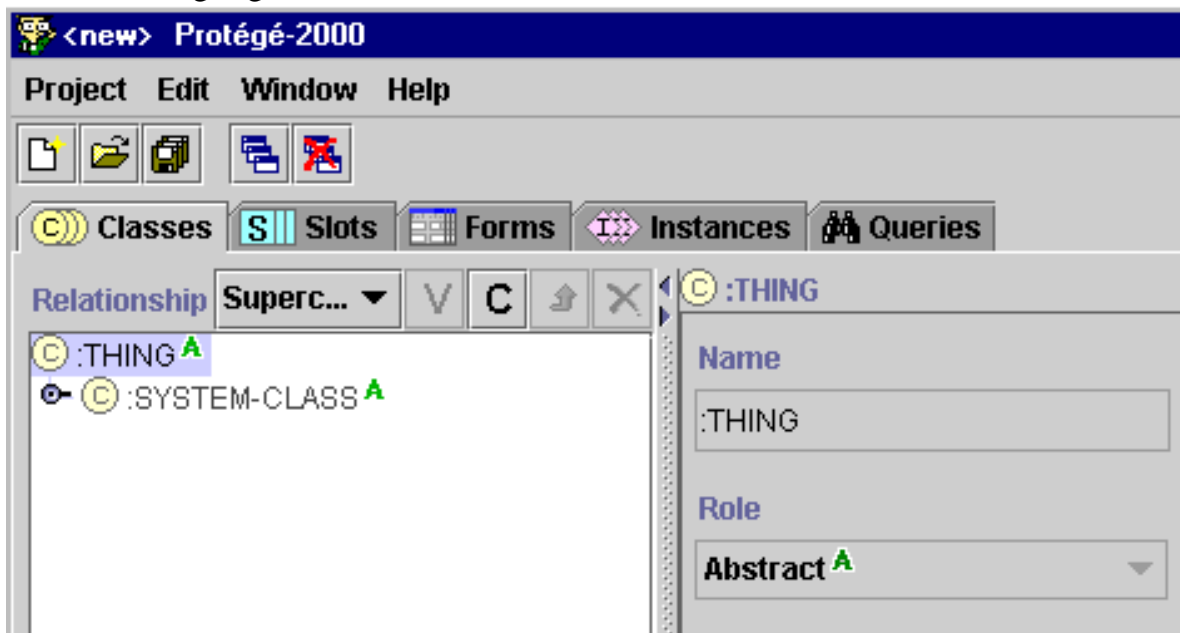
Once you have created the project, the first thing you need to do to start constructing your ontology is to add some classes and subclasses. In this section, you will:

- [Create a class, Author](#)
- [Create subclasses of the Author class](#)
- [Create the Person class and some subclasses](#)
- [Add a new superclass to an existing class](#)


## Creating the Author Class

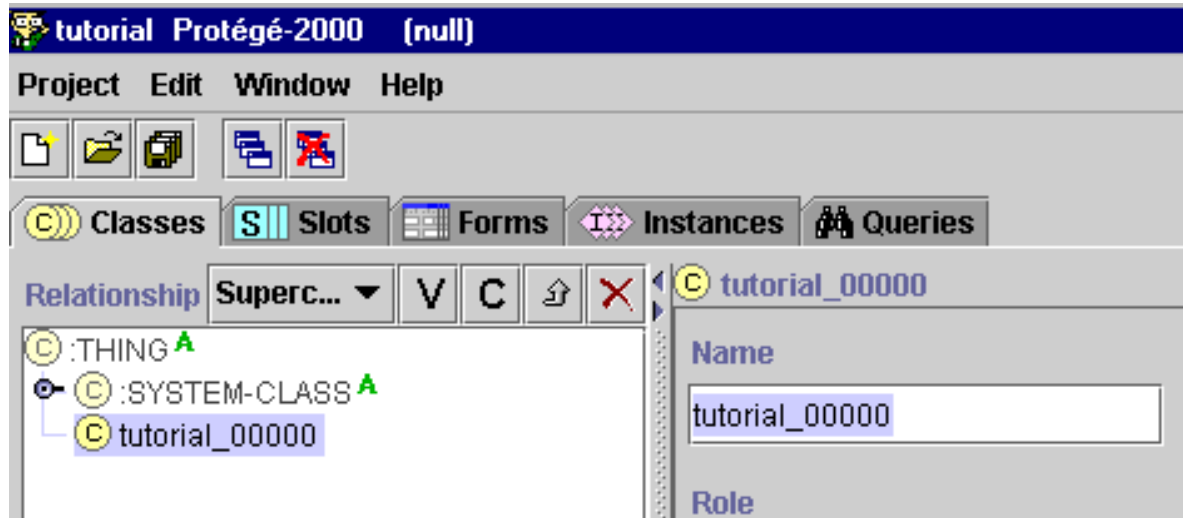
First we will create a new class called Author. An Author is any possible source of an article, such as a news service or columnist. To create the Author class:

1. First, if you have clicked on any buttons, or explored the Protégé window, make sure the Classes tab is selected.
2. Locate the Class Relationship Pane at the left of the Protégé window. This pane shows the hierarchy of classes, with the current class highlighted. Make sure that class :THING is highlighted.

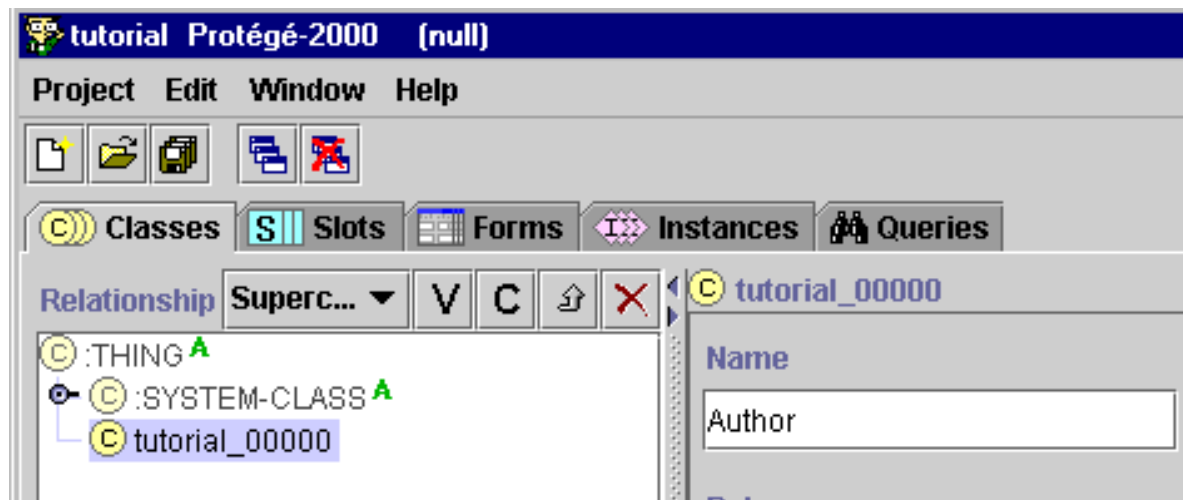


Almost all the classes you create will be created subordinate to :THING. SYSTEM\_CLASS is used by Protégé for defining the structure of various Protégé forms.

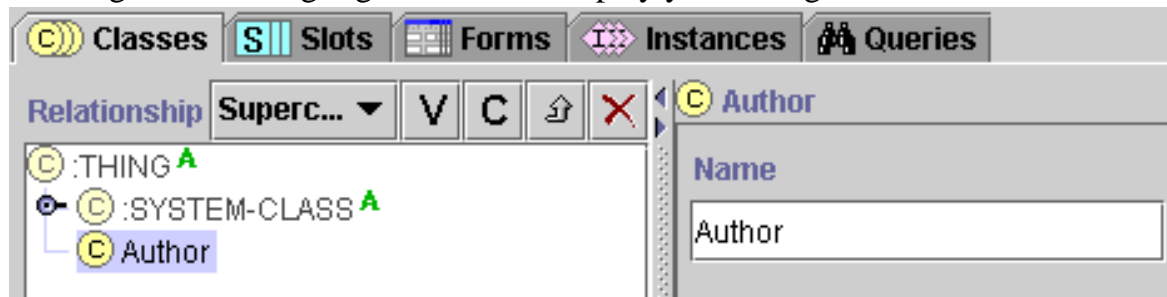
- Click on the Create  button at the top right of the Class Relationship Pane. A class is created with a generic name based on the name of the project, such as tutorial\_00000. You can see the name in the Class Relationship Pane. The class is highlighted to show it is selected.



- Type Author. A recommended Protégé convention is to make the first character of each word in a class name uppercase and the rest lowercase, and to separate words with an underscore.



- Click again on the highlighted class to display your change.



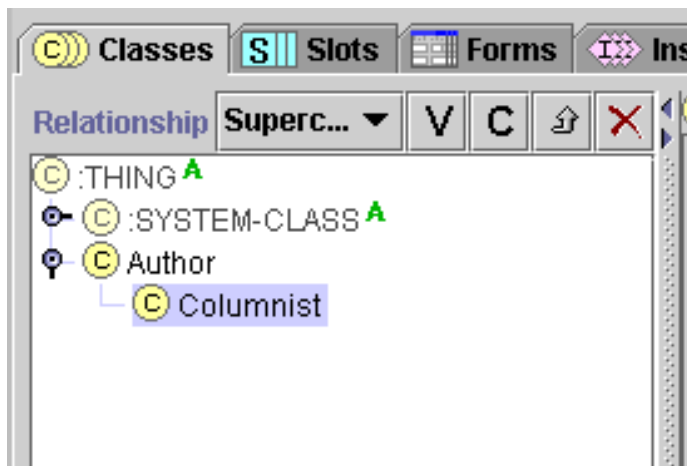
If you have trouble renaming the class, look at the Class Form at the right of the Protégé window. The generic name of the class you just created should be



displayed in the Name field and highlighted. If the correct generic name is displayed, but not highlighted, simply click in the Name field to edit the name. If the incorrect name is displayed, then the wrong class is selected in the Class Relationship Pane. Click on the class you want to rename.

## Creating Subclasses of Author

Now we want to create some subclasses of Author:

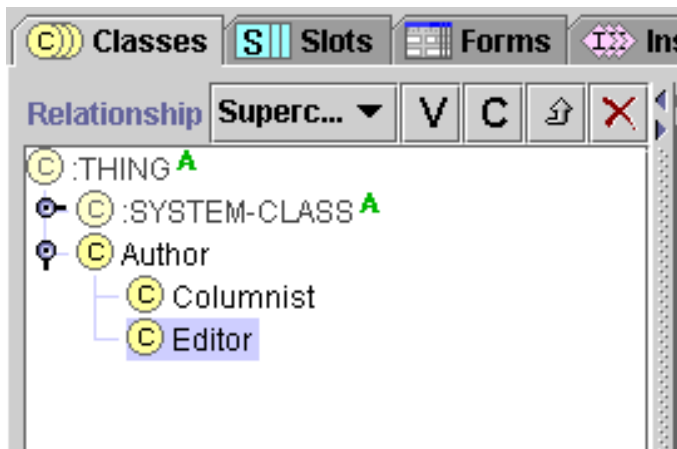
1. Make sure Author is selected in the Class Relationship Pane.
2. Click the Create **C** button.
3. Rename the class to Columnist.



Notice that when you create the first subclass of a class, a  or  icon appears to its left. You can use this icon to display or hide the subclasses of a class.

To create another subclass of Author:

1. Select Author in the Class Relationship Pane. If you do not remember to select Author, you will create a subclass of Columnist.
2. Click the Create **C** button.
3. Rename the class to Editor.



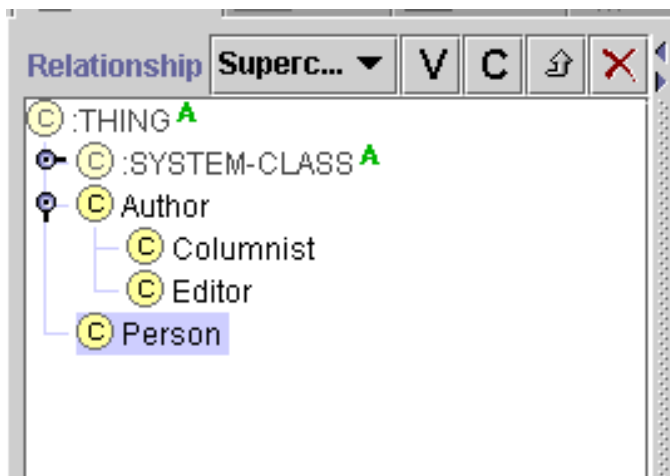
## Creating the Person Class

Now we want to create another class, called Person. A Person is any individual, whether or not they are an Author.

1. Select :THING in the Class Relationship Pane. This ensures that the new class will be at the right level in the class hierarchy.



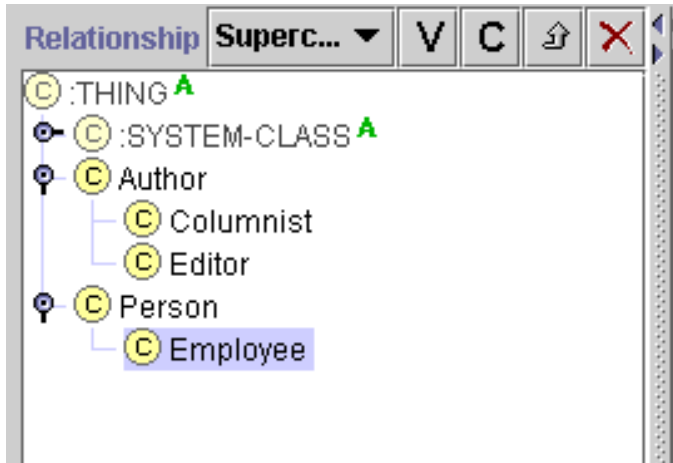
2. Click the Create **C** button.
3. Rename the class to Person.



## Creating a Subclass of Person

Now create a subclass of Person, Employee:

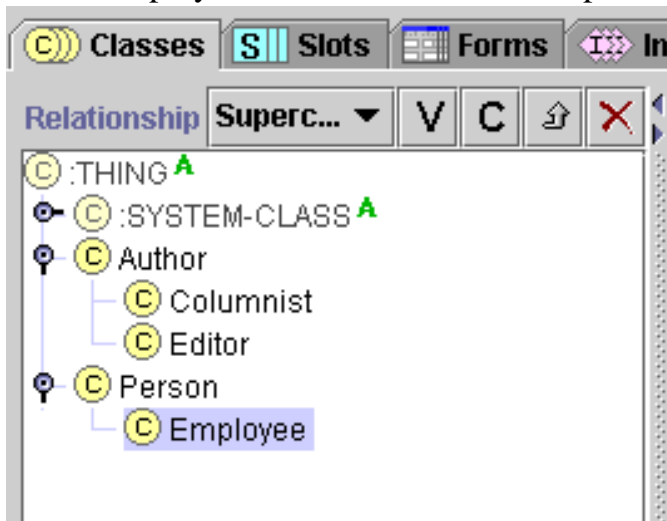
1. Select Person in the Class Relationship Pane.
2. Click the Create **C** button.
3. Rename the new class to Employee.



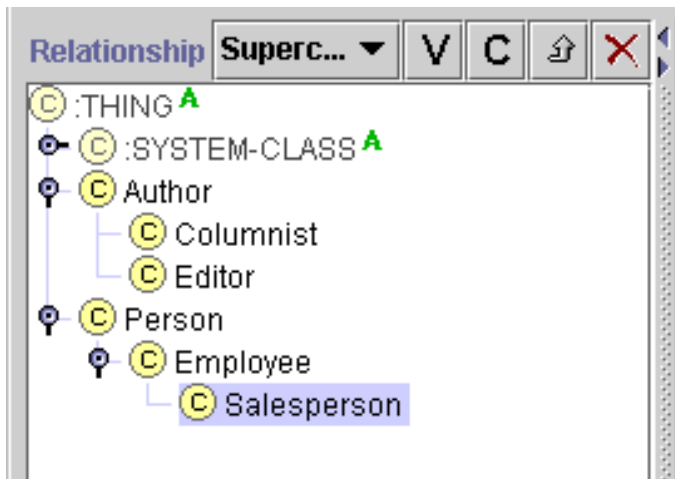
## Creating a Subclass of a Subclass

We are now ready to create some subclasses of Employee. First, let us create an employee type who is not an Author:

1. Select Employee in the Class Relationship Pane.



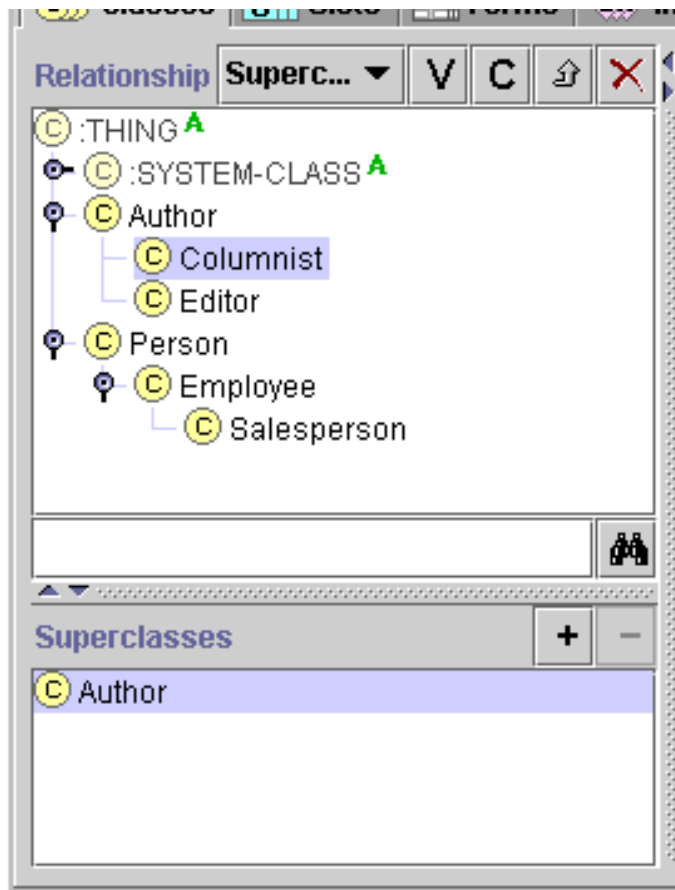
2. Click the Create **C** button.
3. Rename the class to Salesperson.



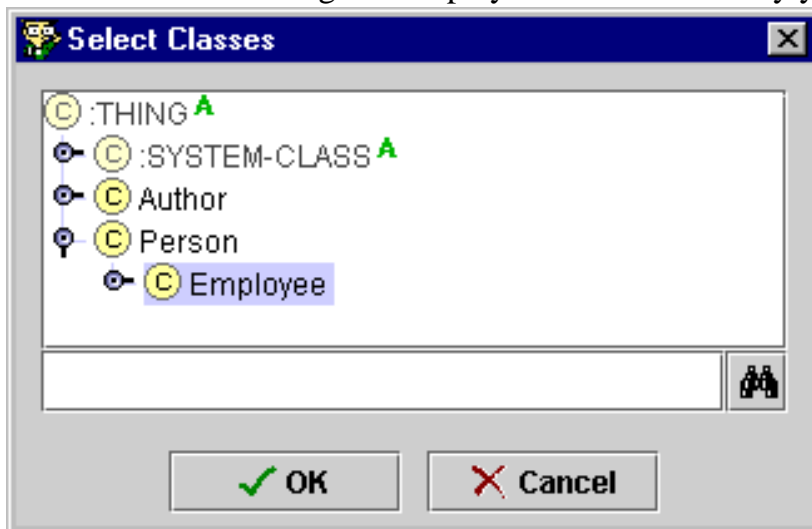
## Adding an Additional Superclass to an Existing Class

We also want a Columnist to be an Employee. Since we have already created the Columnist class, we do not create it again. Instead, make the existing Columnist class a subclass of Employee as follows:

1. Select Columnist in the Class Relationship Pane. The current superclass of **Columnist**, **Author**, is displayed in the Superclasses pane just below the Class Relationship Pane.

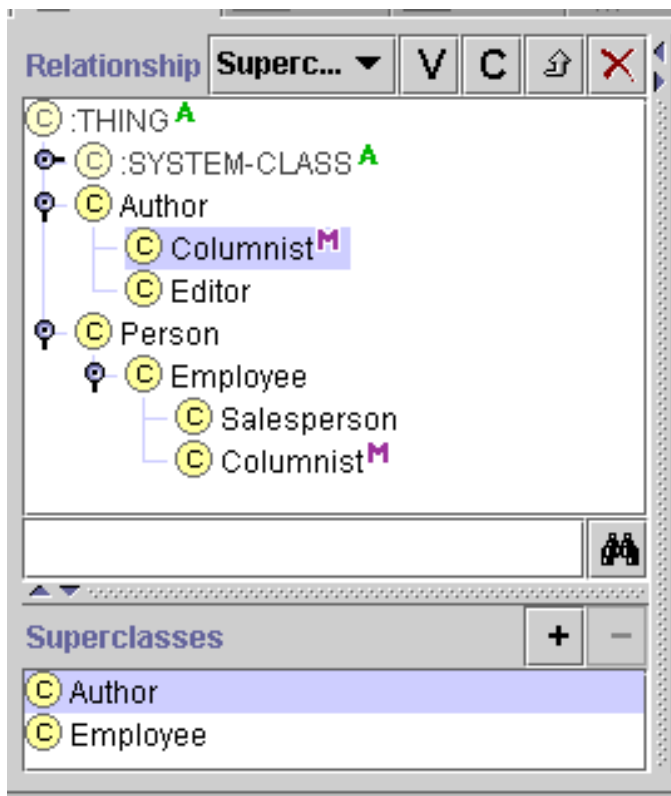


2. Click the Add **+** button in the Superclasses pane.  
A Select Classes dialog box displays the class hierarchy you have developed so far.



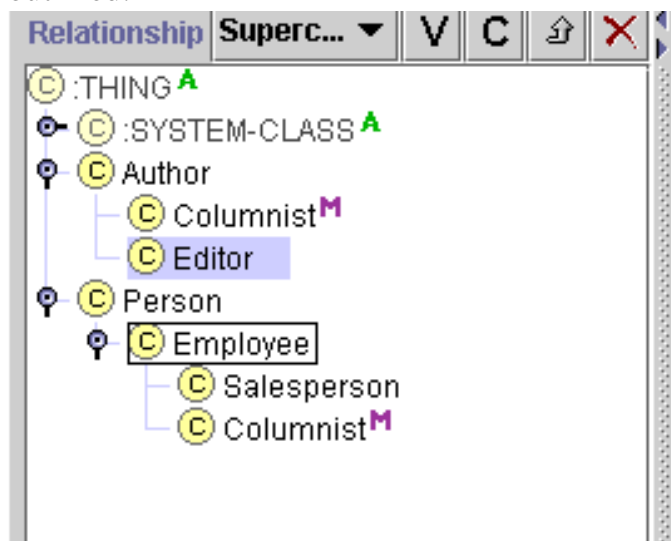
3. Click the **+** icon next to **Person** to display the subclasses of **Person**.
4. Highlight **Employee**.
5. Click OK.

**Columnist** now appears in two places in the Class Window: once under **Author** and again under **Employee**. These superclasses are listed in the Superclasses pane. In addition, an **M** icon appears after each instantiation of **Columnist**, to show that it has multiple parents.

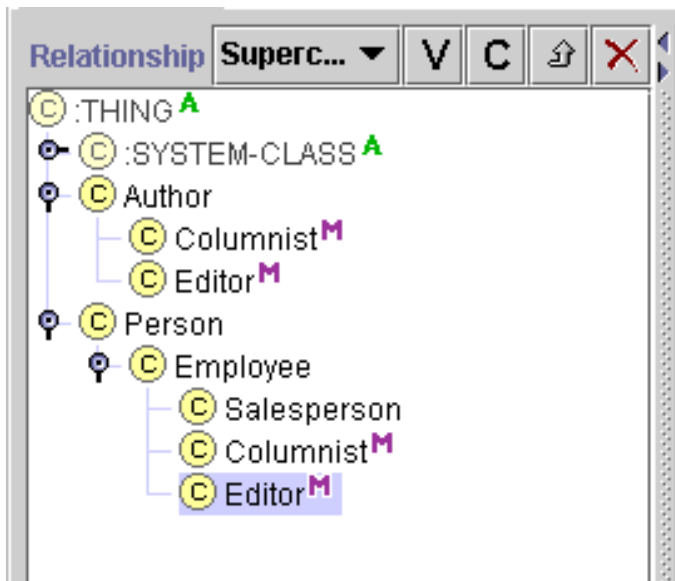



You can also add a superclass using drag and drop:

1. Select Editor in the Class Relationship Pane.
2. Hold down the mouse button and drag the **Editor** class over **Employee**. **Employee** is outlined.



3. Hold down the **Ctrl** key and release the mouse button to drop the subclass.



4. If **Editor** appears under **Person**, you did not hold down the **Ctrl** key long enough while dropping. Delete the superclass by clicking on the **Remove**  button in the Superclasses Pane while **Person** is highlighted, then try again.

You have now created several classes and given two classes multiple superclasses. You are now ready to assign some attributes to the classes you have created, by creating slots. This is covered in the next section.

---

Next: [Creating and Naming Slots](#)

[Getting Started With Protégé](#)



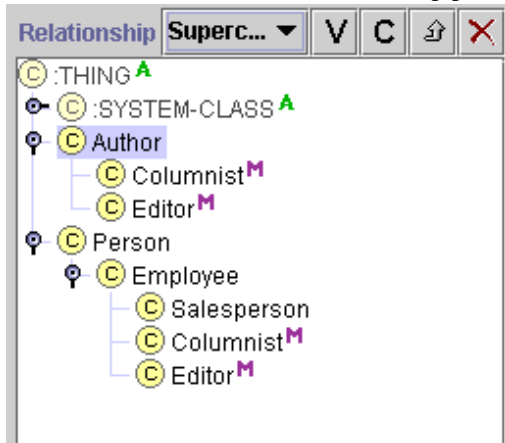
# Creating and Naming Slots

You now have several classes in your project. You can assign attributes to these classes by creating slots. In this section you will:

1. [Create a basic slot for the \*\*Author\*\* class.](#)
2. [Create a basic slot for the \*\*Person\*\* class.](#)
3. [Create a Float slot for the \*\*Person\*\* class.](#)
4. [View how a class with multiple superclasses inherits from both of them.](#)

To create a slot for the class **Author**:

1. Select **Author** in the Class Relationship pane.



2. Click the Create **C** button at the far right of the Template Slots pane.

The screenshot shows the 'Template Slots' pane with a table. The table has the following columns: Name, Type, Cardinality, and Other Facets. The 'Create' button (C) is visible at the top right of the pane.

Name	Type	Cardinality	Other Facets
------	------	-------------	--------------

3. The Slot Form is displayed.

The screenshot shows a window titled "tutorial\_00007" with the following fields and controls:

- Name:** A text box containing "tutorial\_00007".
- Value Type:** A dropdown menu currently showing "String".
- Documentation:** An empty text area.
- Template Values:** An empty text area with control buttons (V, C, +, -).
- Cardinality:**
  - required
  - multiple
  - at least:
  - at most:
- Default:** An empty text area with control buttons (V, C, +, -).
- Minimum:** An empty text box.
- Maximum:** An empty text box.
- Inverse Slot:** An empty text box with control buttons (V, C, +, -).

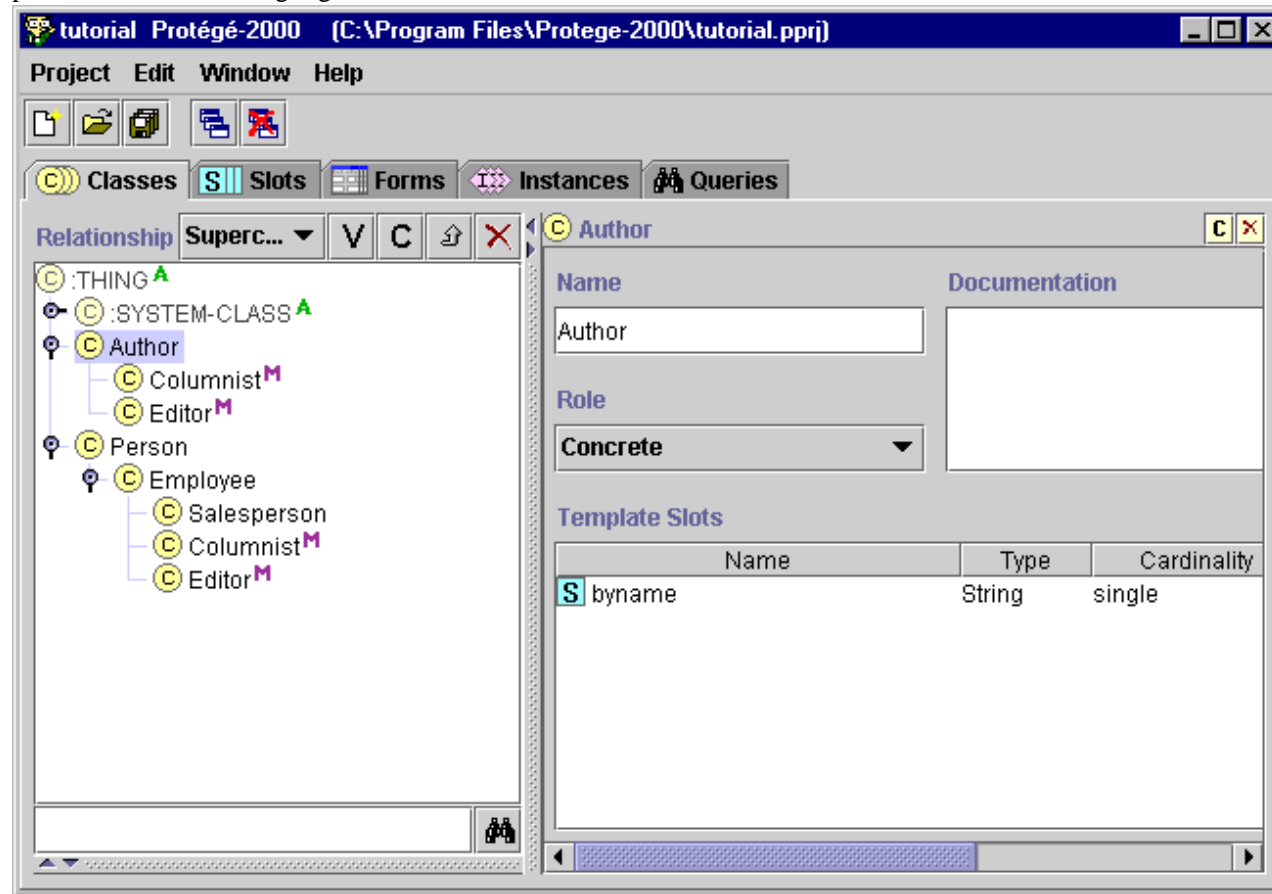
4. To name the slot, highlight the generic name in the Name field. Type the name for the slot, **byname**. A recommended convention to make slot names lowercase, with words separated with an underscore. Making slots names lowercase, while making classes start with an uppercase letter, helps you distinguish them in your ontology.

The screenshot shows the same window as above, but with the following changes:

- Name:** The text box now contains "byname".
- All other fields and controls remain identical to the previous screenshot.

5. This slot will be a string slot, that is, a slot that takes values that are any string of alphanumeric characters, possibly including spaces. For this simple slot, we will not change any of the other fields in the Slot Form. Simply close the form and return to the main window. Notice that your slot now appears in the Template Slots

pane when **Author** is highlighted.



The blue slot icon indicates that the slot is "direct", that is, it was created directly on the selected class. You can edit a direct slot by double-clicking on it to open the slot form.

We will now create two slots for the **Person** class. To create a string slot for **Person**:

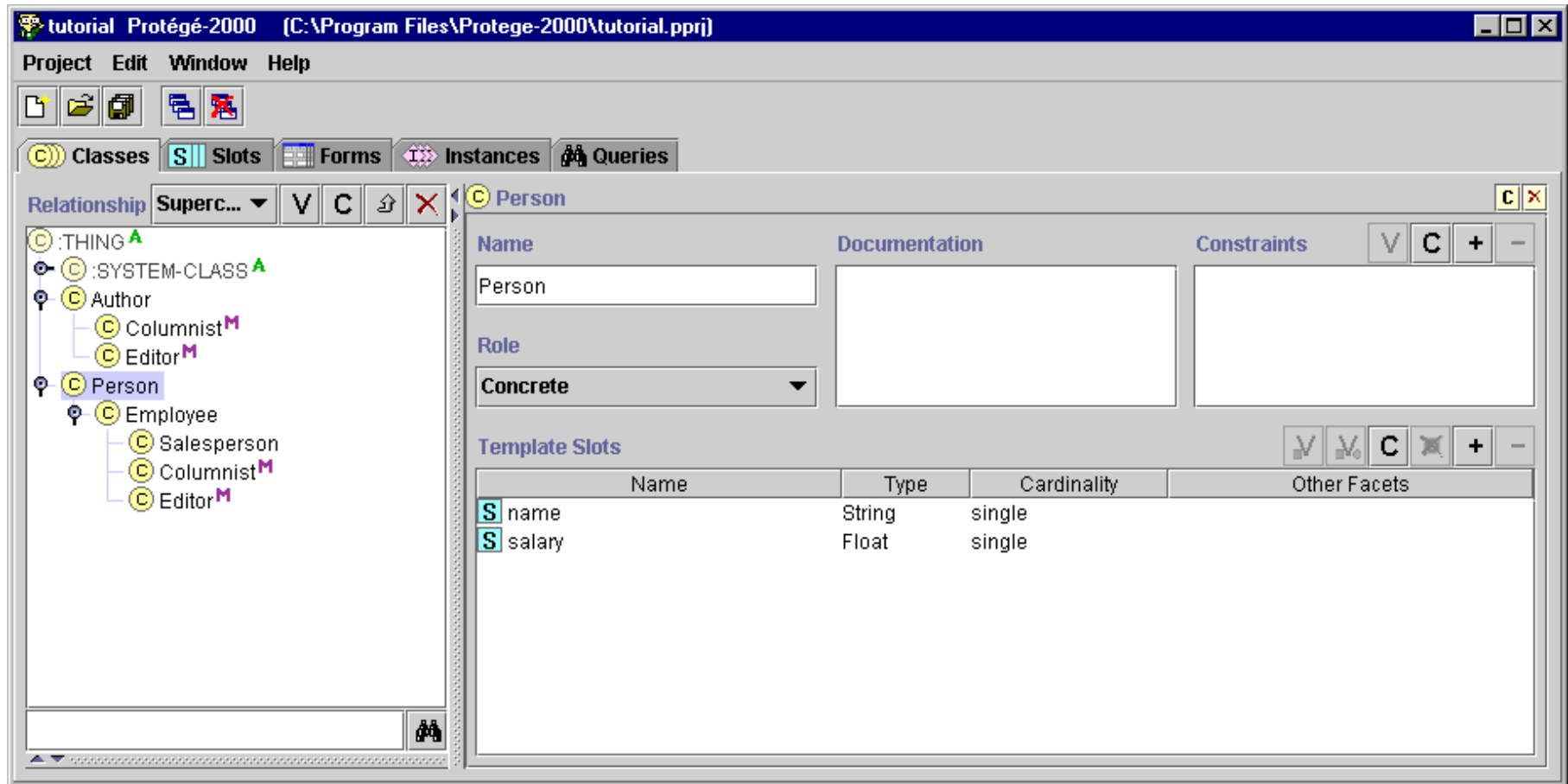
1. Click on **Person** in the Class Relationship Pane.
2. Click the Create **C** button at the far right of the Template Slots pane.
3. Change the slot name to **name** in the Slot Form.
4. Exit the Slot Form.

To create the next slot, which will be of Type Float:

1. Make sure **Person** is still selected in the Class Relationship Pane.
2. Click the Create **C** button at the far right of the Template Slots pane.
3. Change the slot name to **salary** in the Slot Form.
4. Select **Float** from the **Value Type** menu.

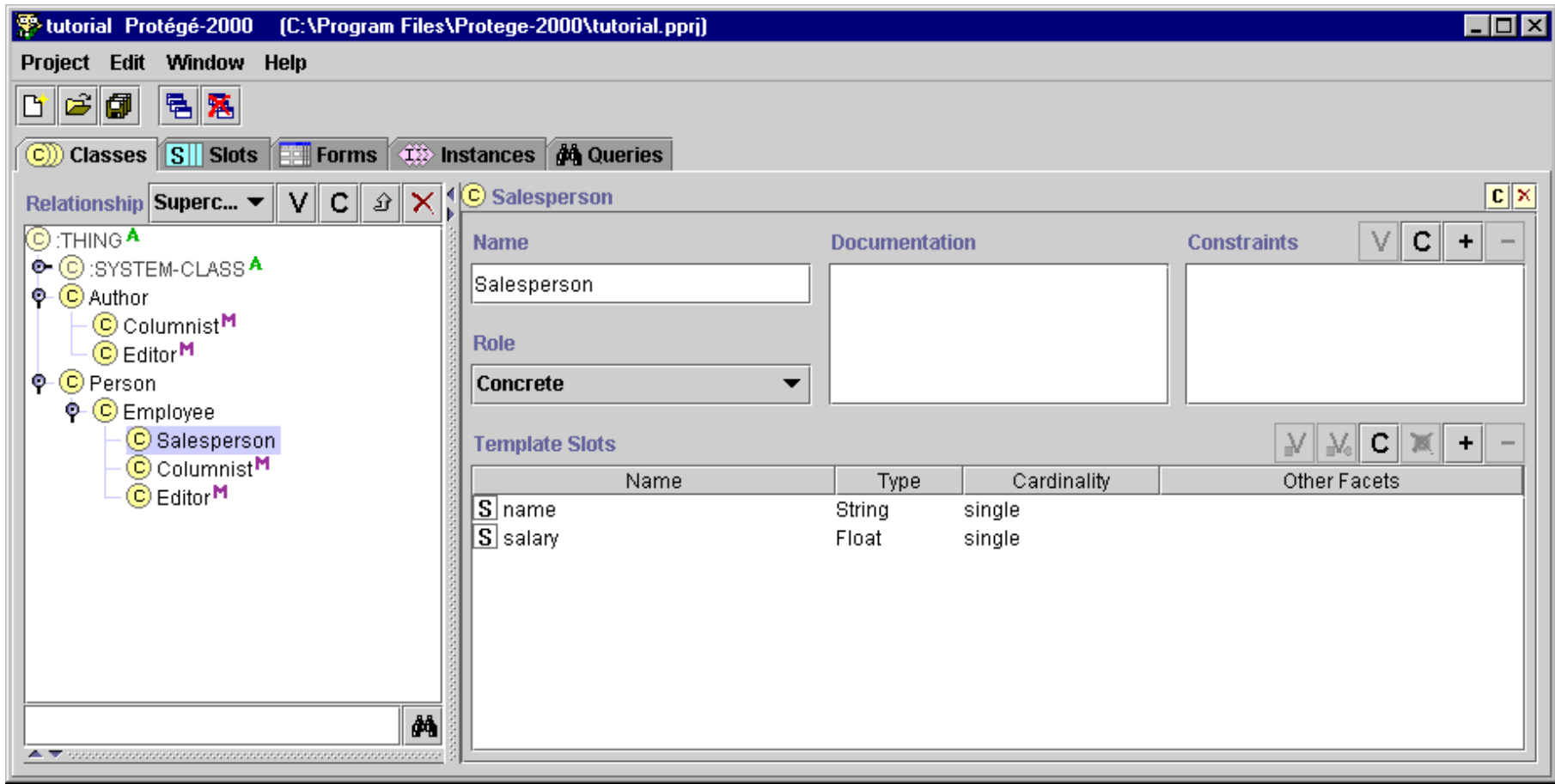
A Float is a floating point integer. Values that are entered for this slot must be valid floating point numbers.

5. Exit the slot form. You now see two direct slots for **Person**.



## Slots and Inheritance

The slots you create at a class are inherited by all their classes. To see this, click on **Salesperson**. You will see the slots you created for **Person**. Notice that the slot icons are now white, to indicate that these slots are inherited.



Now, click on **Columnist**. You see that **Columnist** inherits slots from both **Person** (via **Employee**) and **Author**. This multiple inheritance is an essential part of Protégé.

You have now created some simple slots. This is all you need to enter some instances and begin creating your knowledge base.

---

Next: [Entering Instances](#)

[Getting Started With Protégé](#)



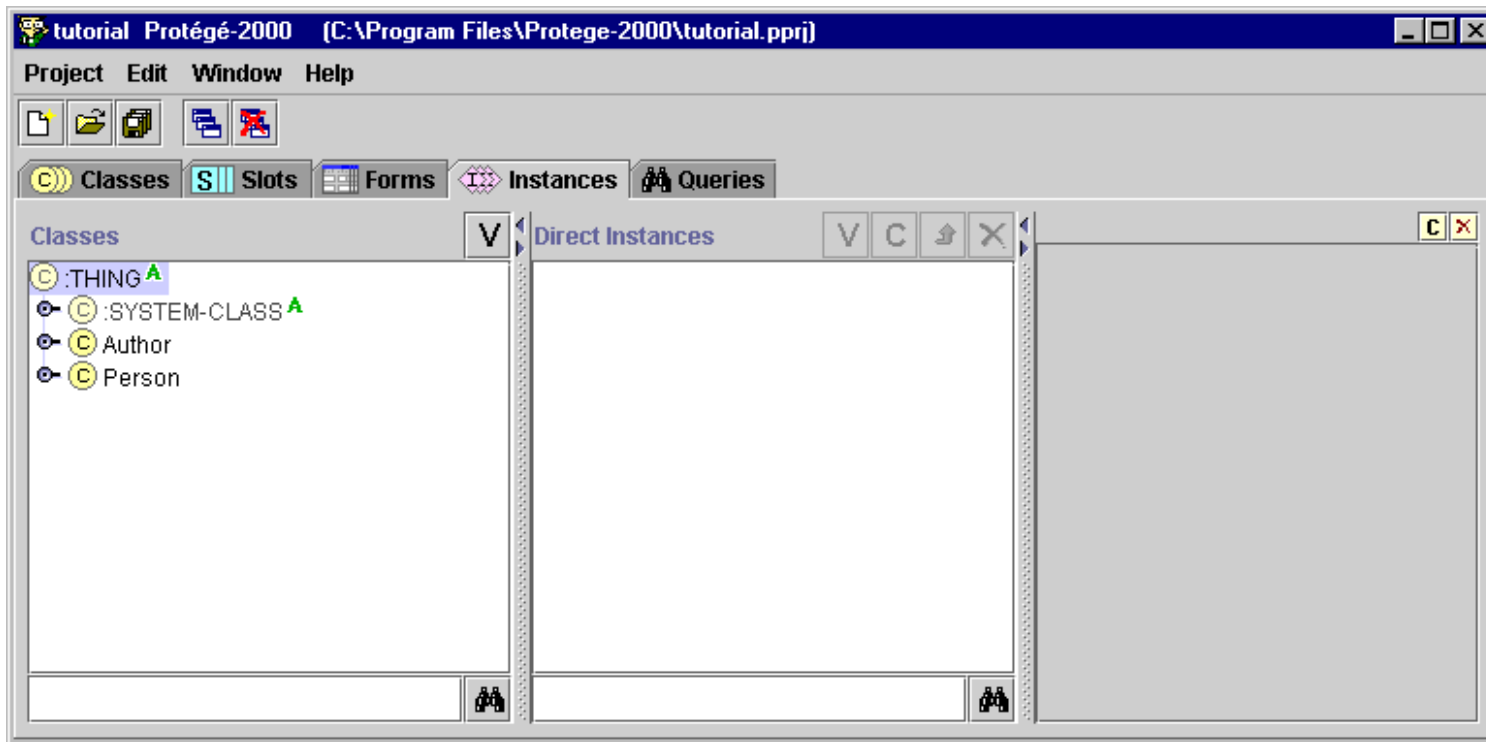
# Entering Instances

Instances are the actual data in your knowledge base. In general, it is a good idea to make sure you have structured your project as well as you can before entering extensive instances. If you have to make changes to your class or slot structure after instances have been entered, you may lose some information. In addition, if you add slots, you will not have that information for your new instances.

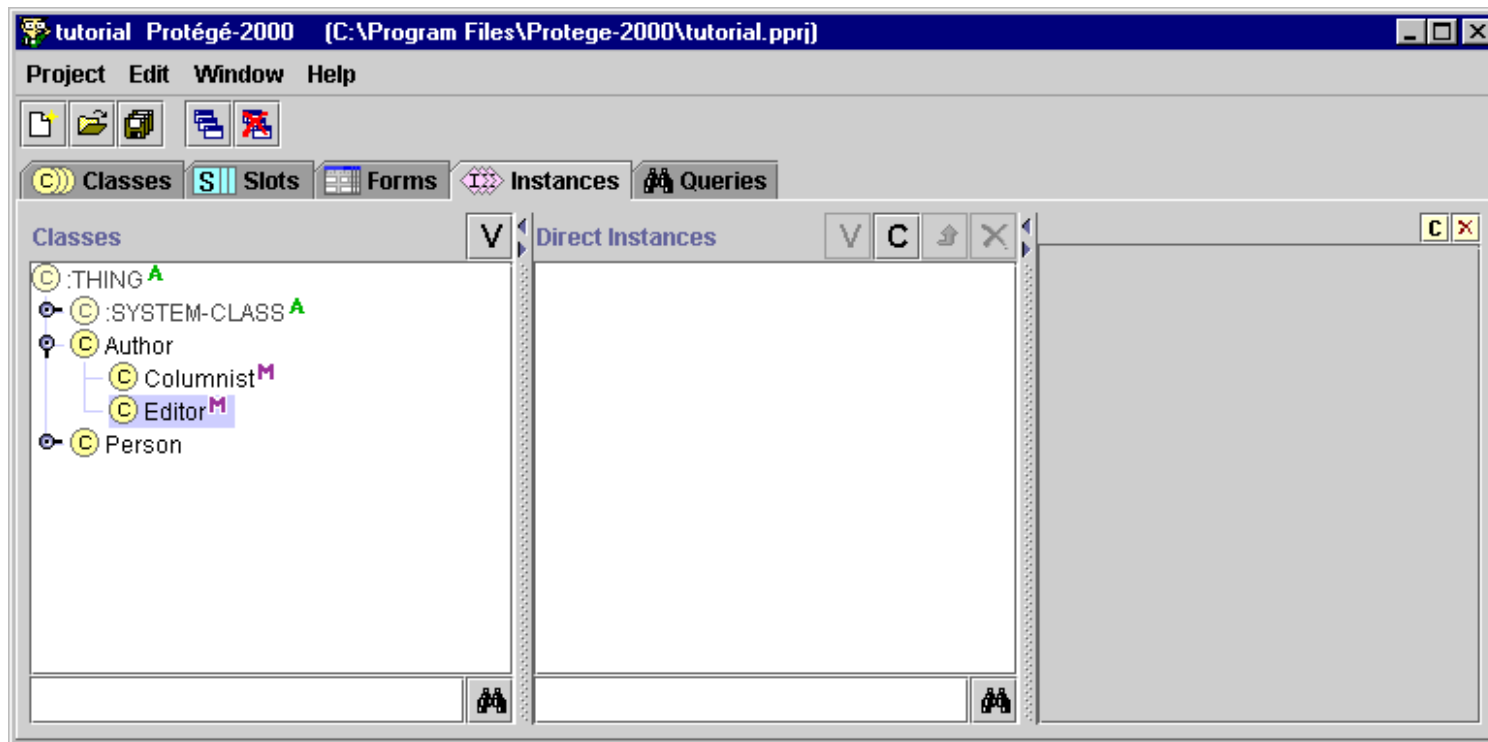
In this section, you will enter two instances for the **Editor** class.

To enter an instance for **Editor**:

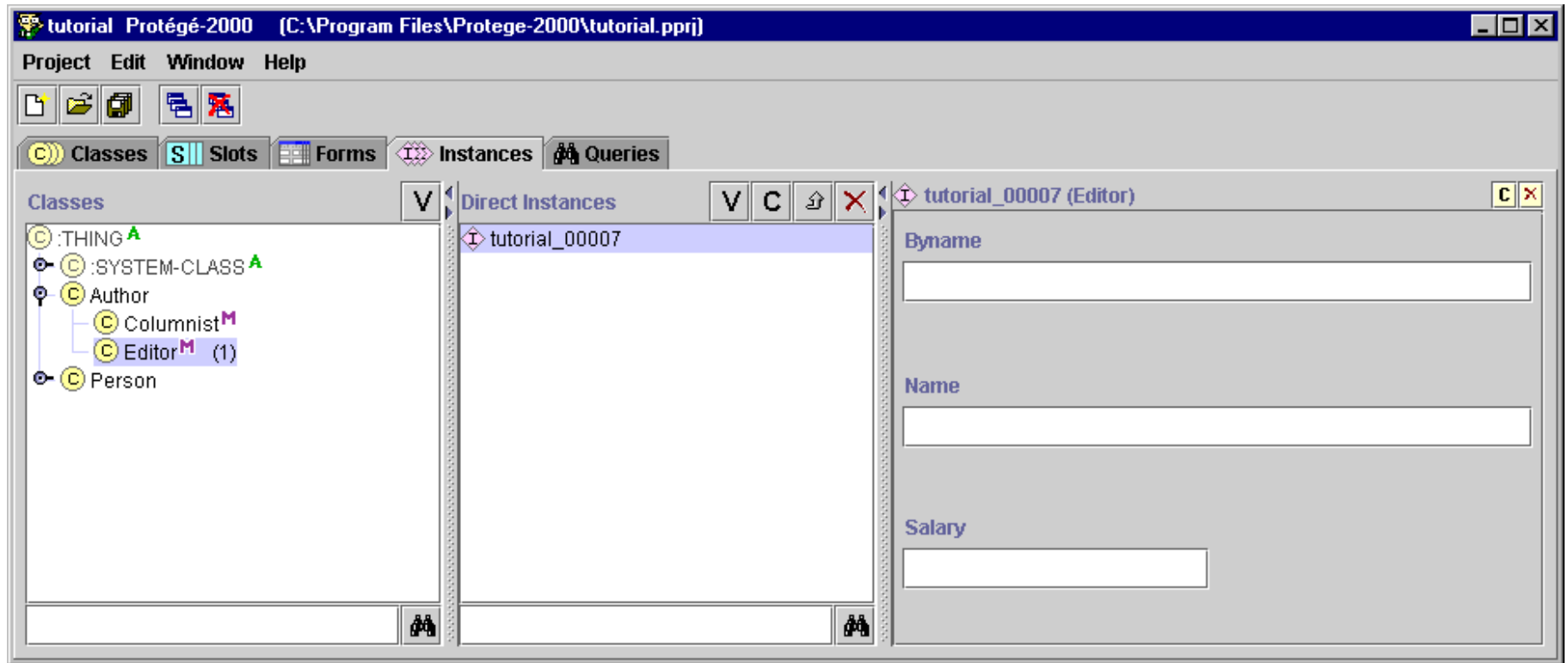
1. Click on the Instances tab.



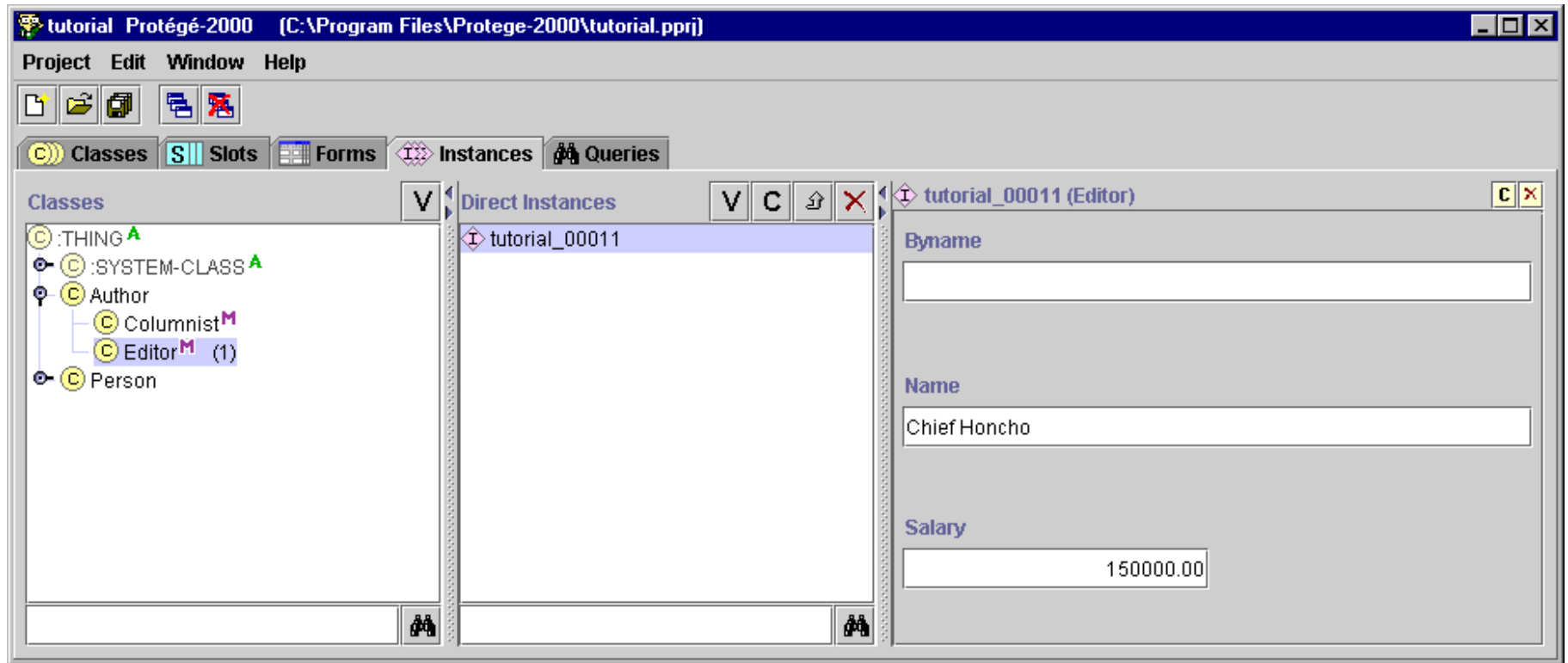
2. Click on **Editor**. The Create **C** button in the Direct Instances Pane becomes active, indicating you can now create an instance.



3. Click on the Create button in the Direct Instances Pane to create an instance.



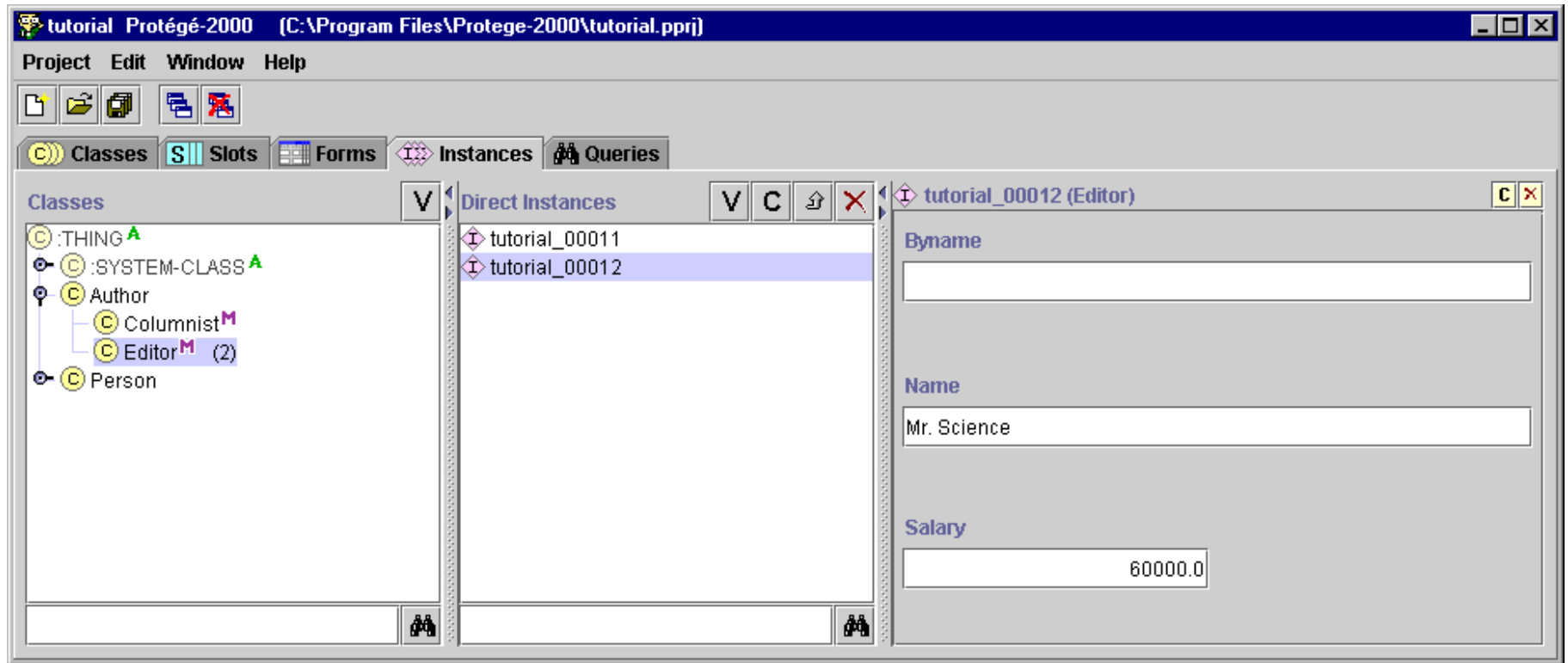
4. Enter **Chief Honcho** in the **Name** field.
5. Enter 150000 in the **Salary** field. Notice that this field will not accept commas as part of the entry.
6. **Chief Honcho** does not do any writing herself, so do not enter a byname.



Notice that the information for the **Editor** class changed when you created the instance. The (1) in parentheses after **Editor** indicates that this class has 1 instance.

To create another instance:

1. Click the Create **C** button in the Direct Instances pane again.
2. Type Mr. Science in the **Name** field.
3. Type 60000 in the **Salary** field.



You have now learned to create instances. However, the default form can be difficult to deal with. Most obviously, the names you see in the Direct Instances pane are extremely unilluminating. In addition, no **Editors** have bynames, so the **Byname** field is not useful in this particular form. The next section shows how to edit forms.

---

Next: [Customizing Forms](#)

[Getting Started With Protégé](#)



# Customizing a Form

Forms determine how your instance information is presented and entered. By changing a form, you can:

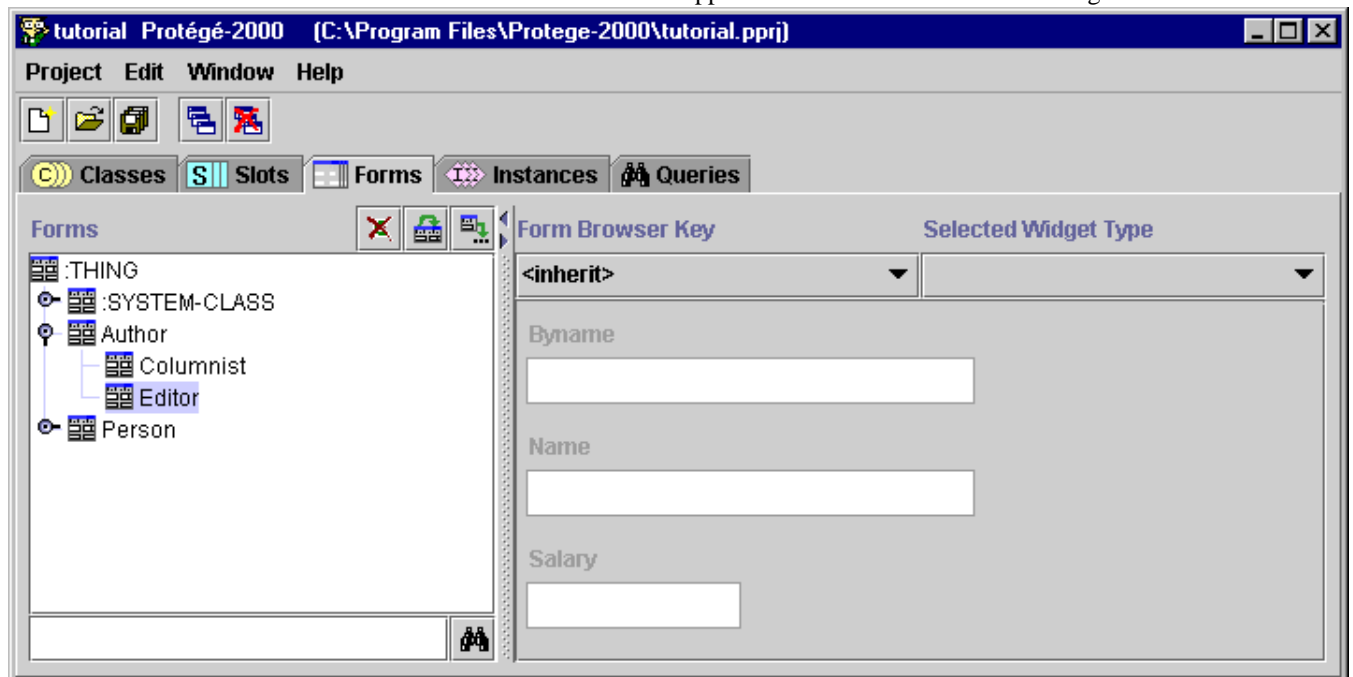
- [change the browser key](#), used to list your instances
- [reposition fields](#)
- [hide a field](#)

## Changing the Browser Key

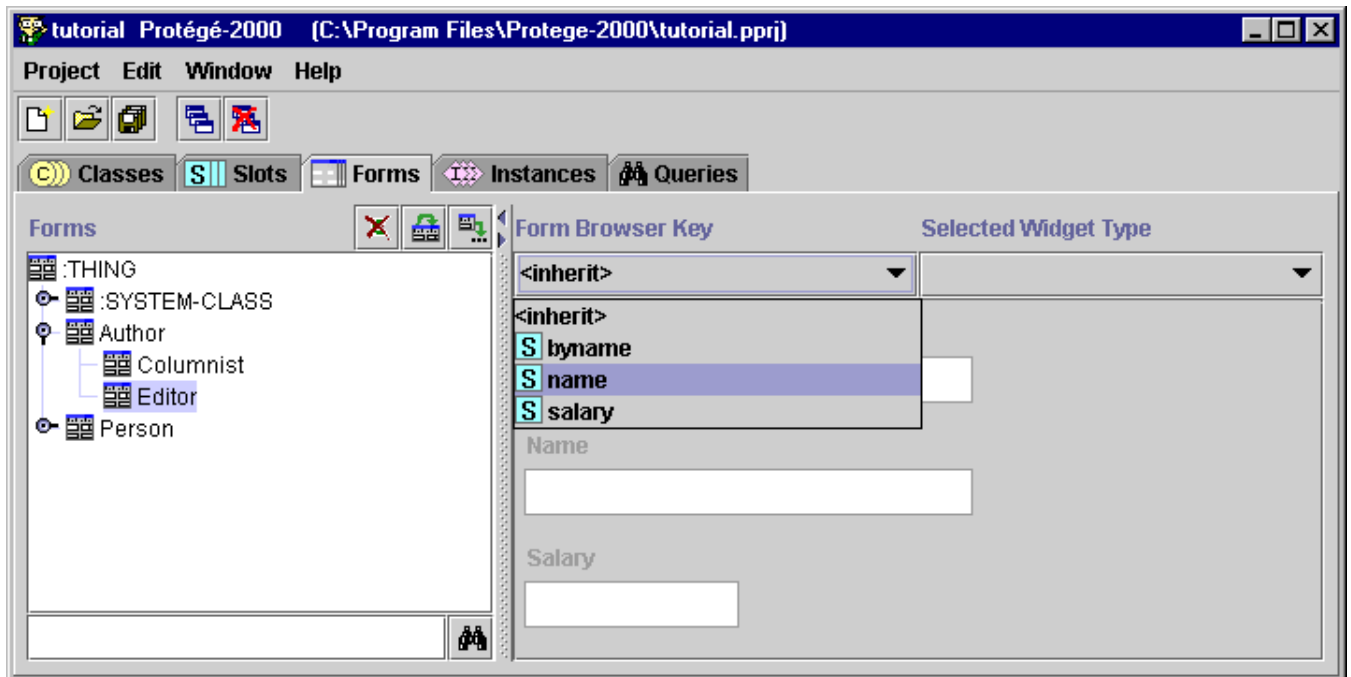
The *browser key* is the slot that is used to identify the different instances when they are displayed in a list. If you do not select a slot to use as a browser key, Protégé-2000 uses a default key, `<inherit>`, such as `newspaper_0017`. It is usually helpful to set a browser key for all classes that will contain instances. In fact, you may choose to set the browser key before you even start entering instances.

To set the browser key for the **Editor** slot:

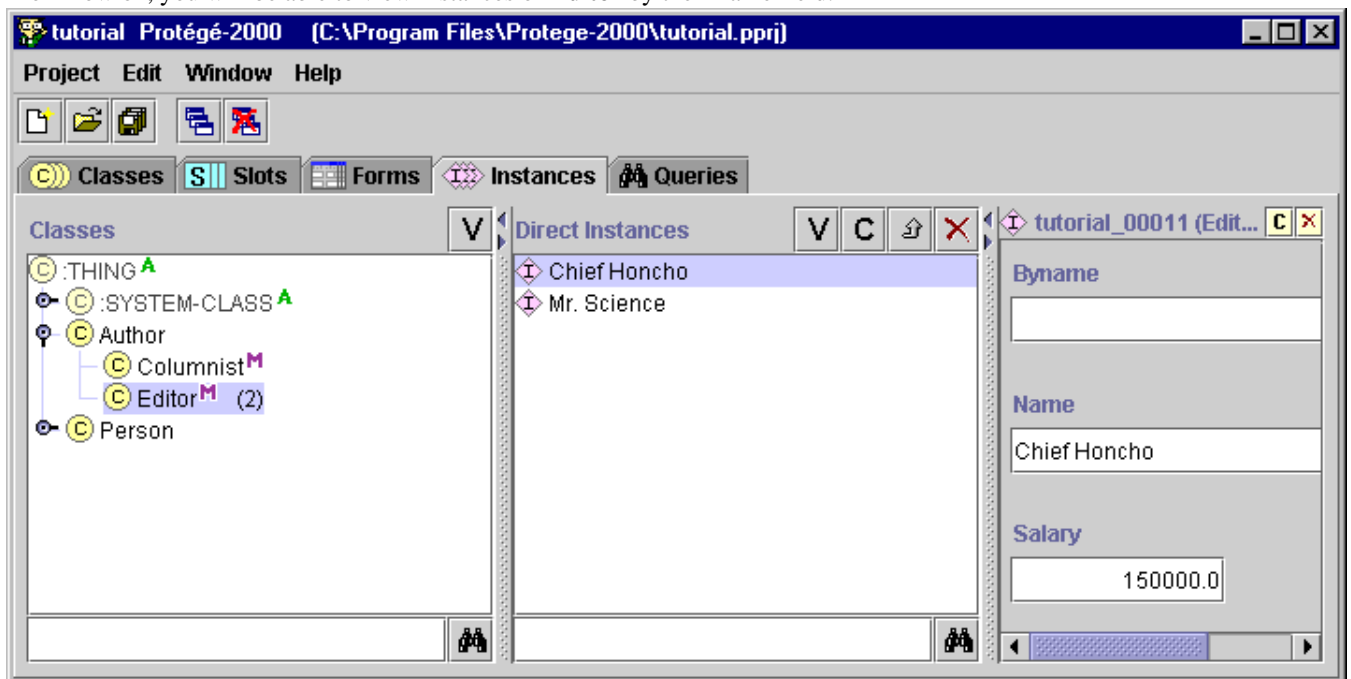
1. Click on the Forms Tab.
2. Click on **Editor** in the Forms Pane to the left.. The default form appears in the Form Edit Pane to the right.



3. Select **name** from the **Form Browser Key** menu.



4. If you wish, view the Instances Tab to see that the name of the instances you entered now appears in the Direct Instances Pane. From now on, you will be able to view instances of **Editor** by their name field.

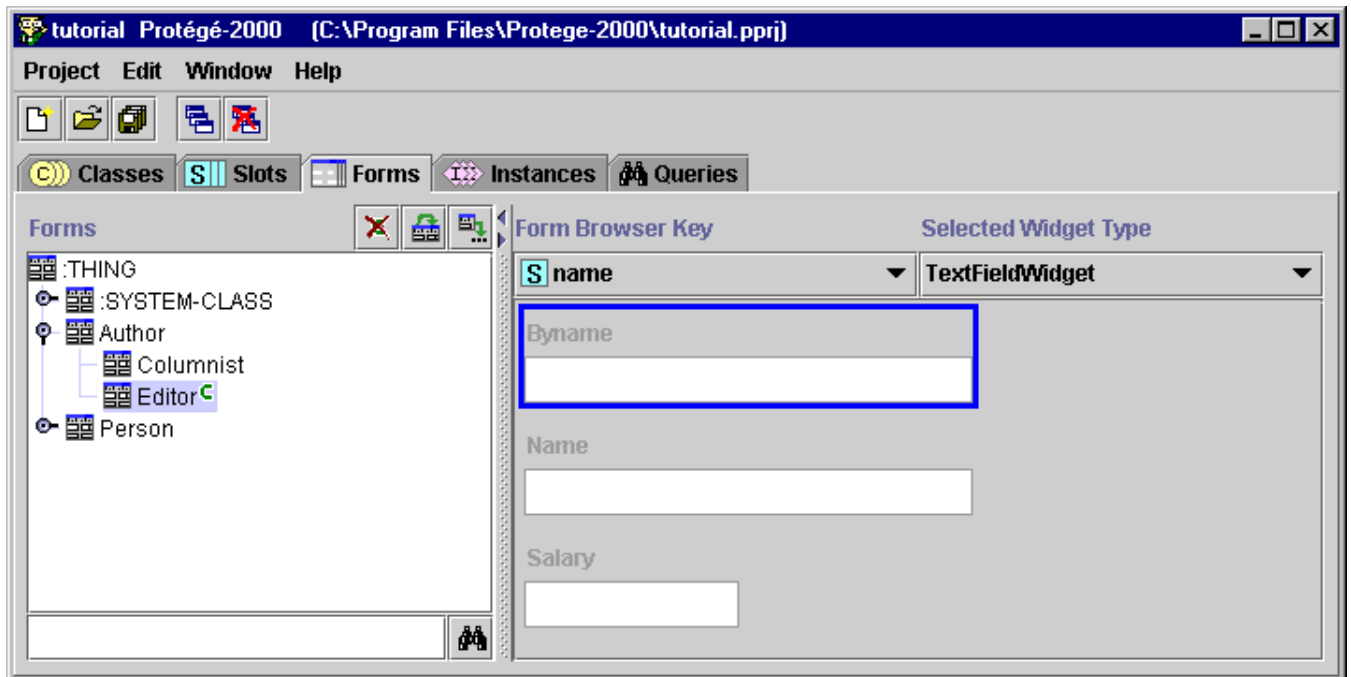


Now return to the Forms Tab to continue customizing the Editor form.

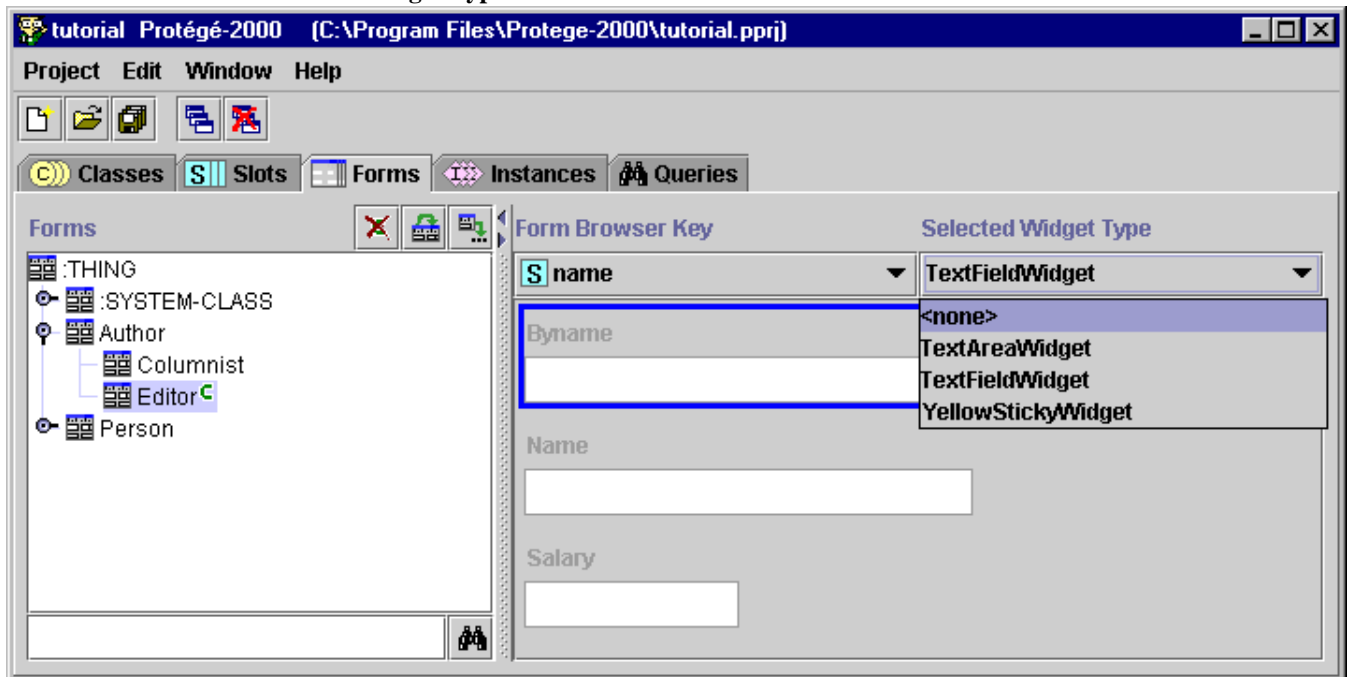
## Hiding a Field

To hide the **Byname** field:

1. Click on the **Byname** field to select it. The **Selected Widget Type** menu becomes active, which allows you to customize the widget.



2. Select <none> from the Selected Widget Type menu.

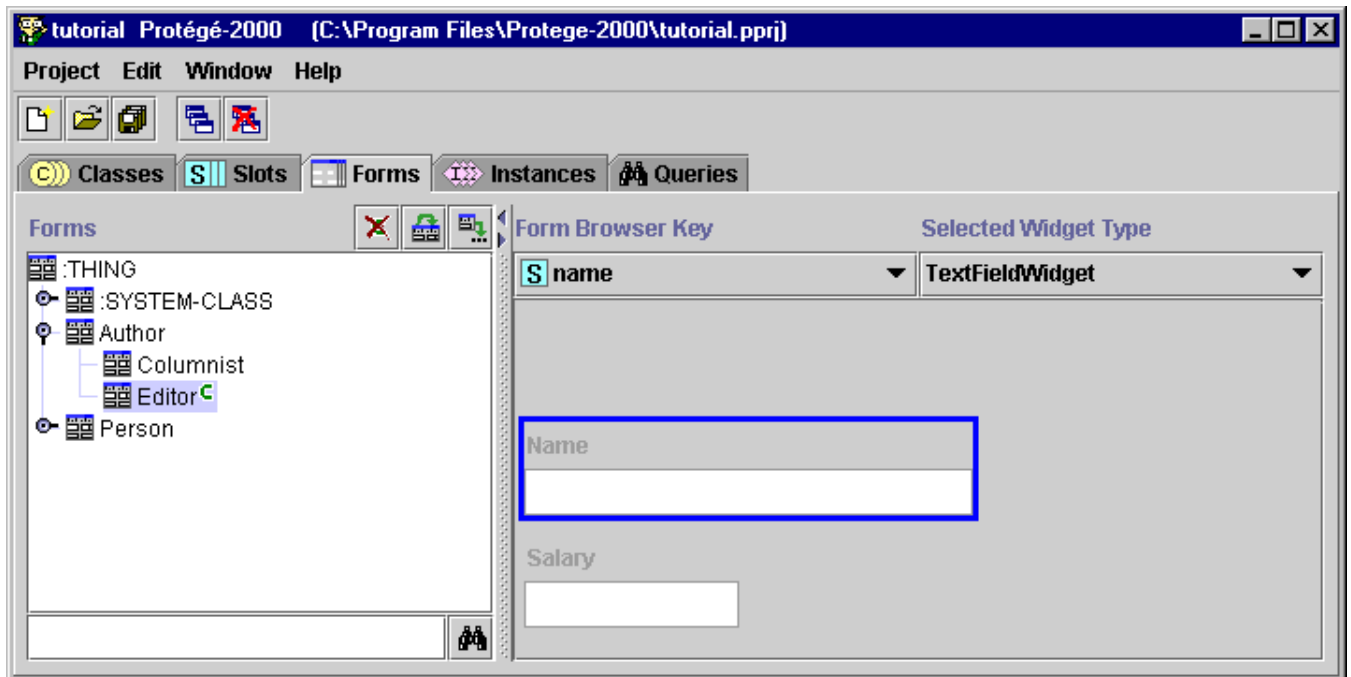


3. The field is removed from the form.

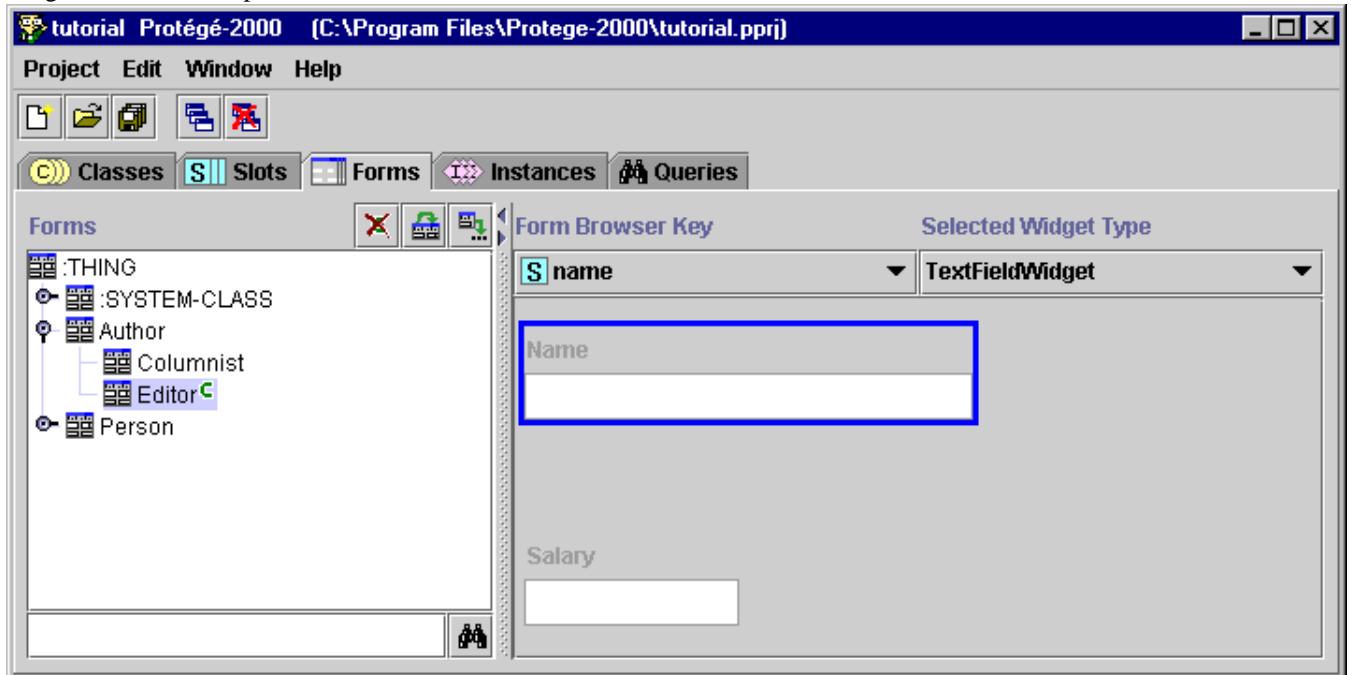
## Repositioning Fields

Now, you may no longer like the layout of the form. To reposition a field by dragging:

1. Select the **Name** field by clicking on it. It is outlined in blue to show it is selected.



2. Drag the field to the top of the Form Edit Pane.

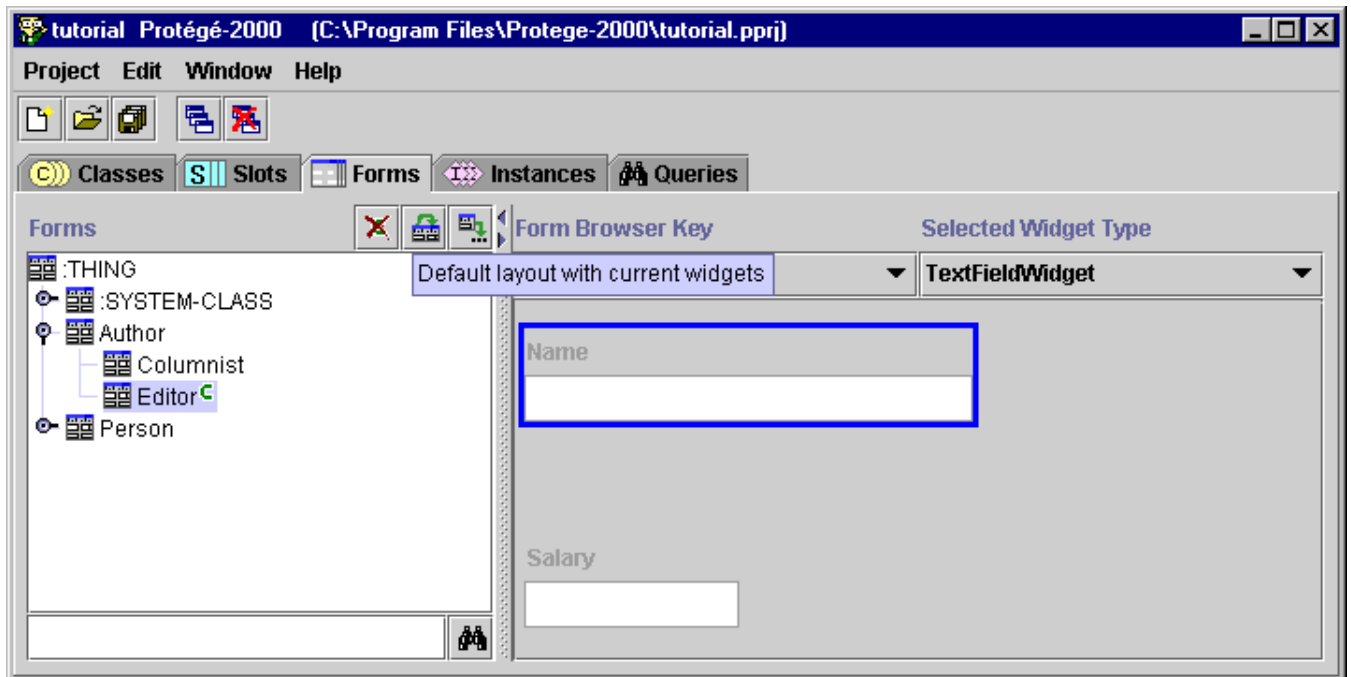


You can also resize a selected form by dragging the edges, just like any window.

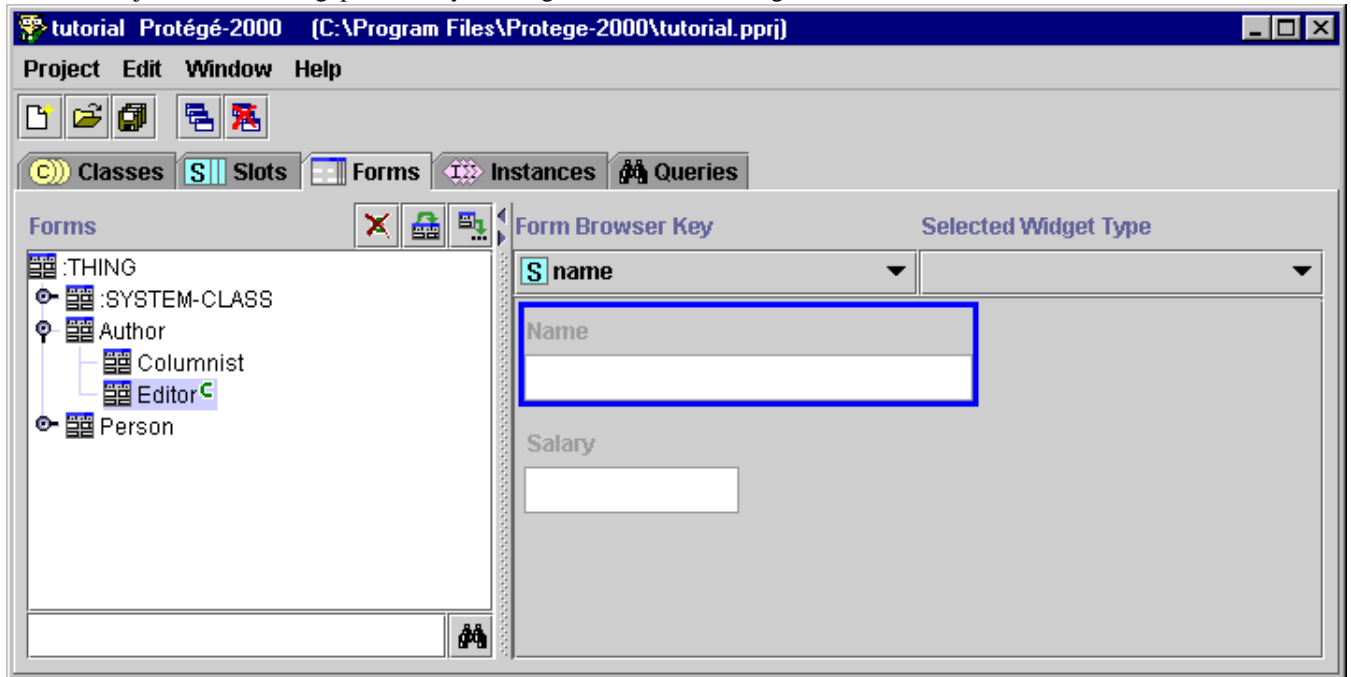
When you have a lot of fields, and you have deleted or resized several of them, it can be time-consuming to move all the fields by hand. Protégé-2000 provides a short-cut to rearrange all the visible fields on a form into a standard order, accommodating their current size. You can then use this as a basis to further customize the form.

To have Protégé-2000 arrange the fields for you:

1. Click the Default Layout button , the center button in the Forms Pane.



2. The form adjusts to close the gap caused by deleting one field and moving another.



Congratulations. You have now created a simple project with Protégé-2000. You can now explore the program on your own, using it to represent your own ontology. For more information about the capabilities of Protégé-2000, see the [Protége-2000 Documentation Home Page](#).

Next: [Creating and Saving a Query](#)

[Getting Started With Protégé](#)



# Creating and Saving a Query

---

The Queries tab allows you to query your project and locate all instances that match criteria you specify. To create a query, you must select one or more classes, and one or more slots within that class. You can also save queries in the Query Library for future recall.

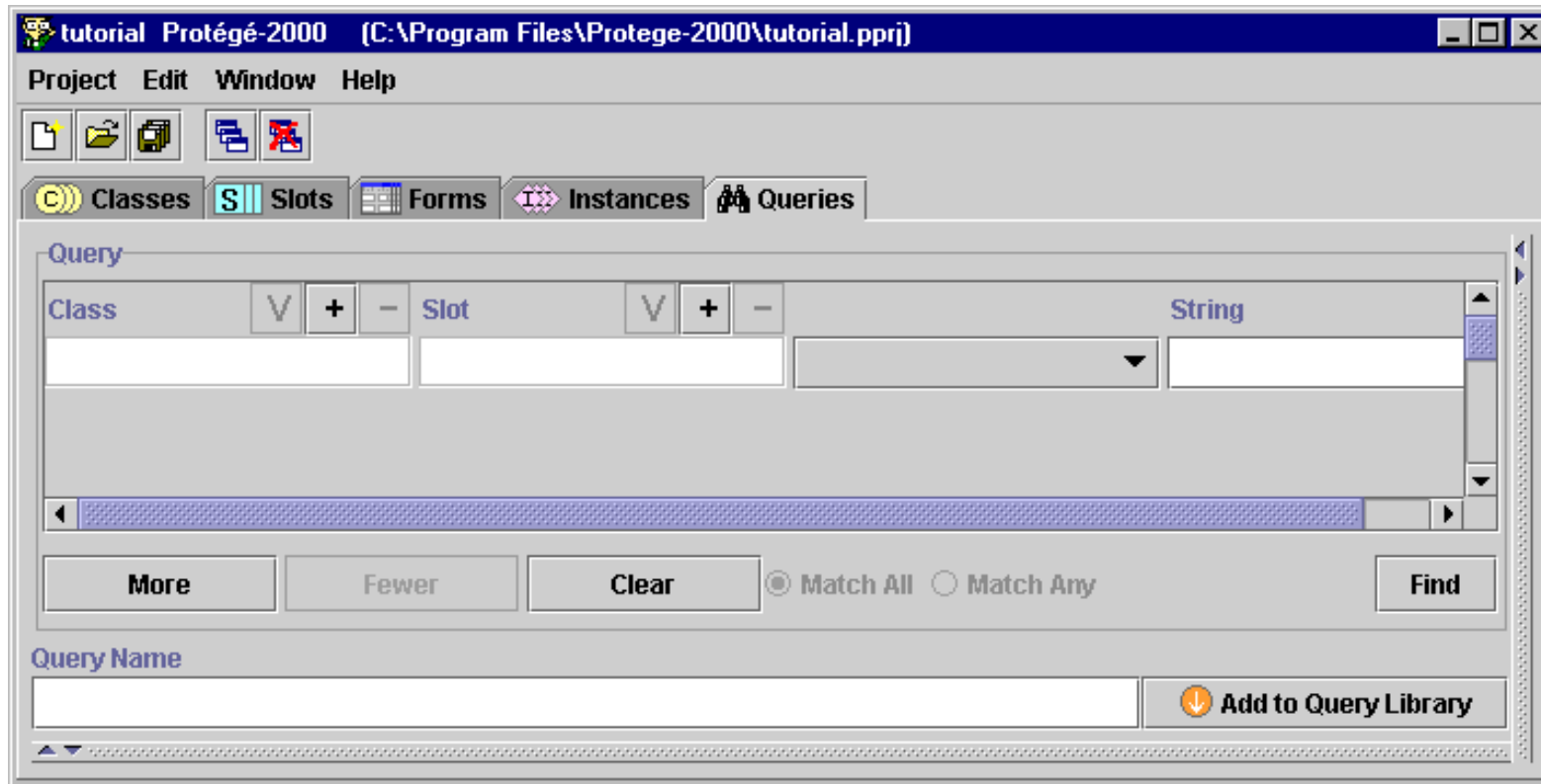
In this section, you will:

1. [Create](#) a query.
2. [Run](#) the query and view the results.
3. [Save](#) the query.
4. [Retrieve](#) the query and rerun it.

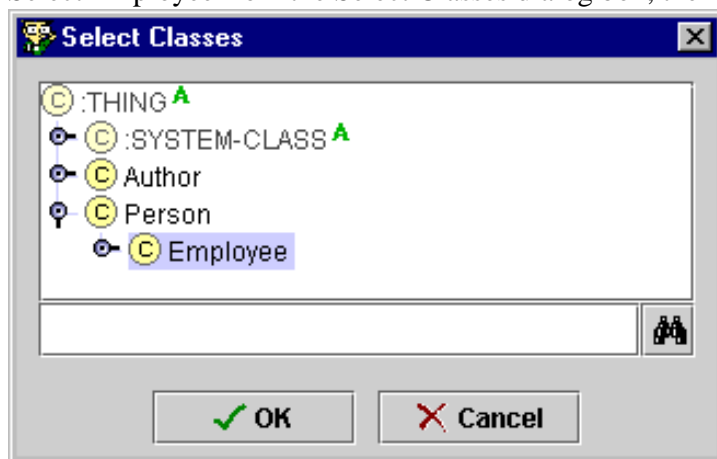
## Creating a Query

Suppose you are interested in locating all Employees who who have a salary greater than \$40,000 dollars a year. To create the query:

1. Click on the Queries tab.

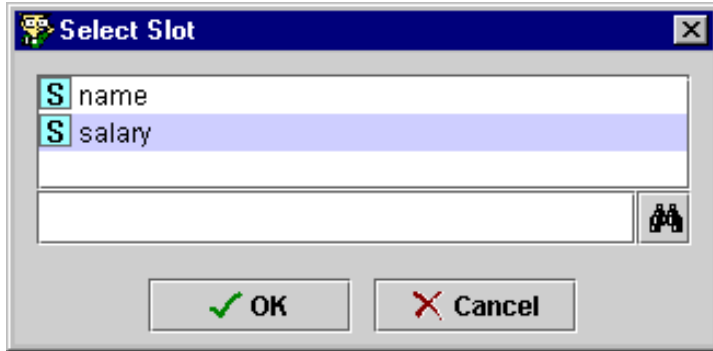


2. First, click the Select Class **+** button above the Class entry bar.
3. Select Employee from the Select Classes dialog box, then click OK.

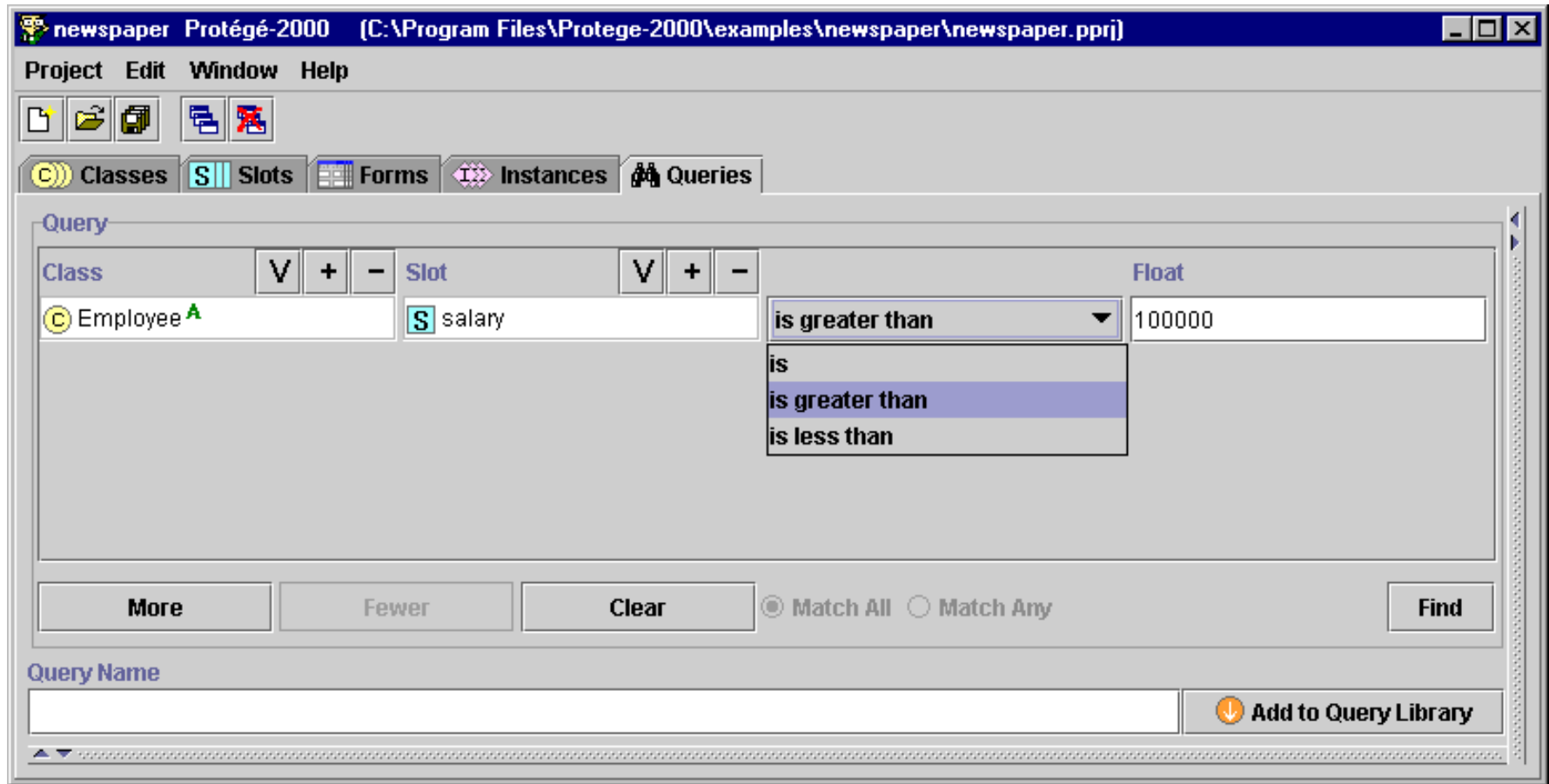


Employee is now displayed in the Class entry bar.

4. With Employee showing in the Class entry bar, click the Select Slot **+** button above the Slot entry bar.
5. Select Salary from the Select Slot dialog box and click OK.



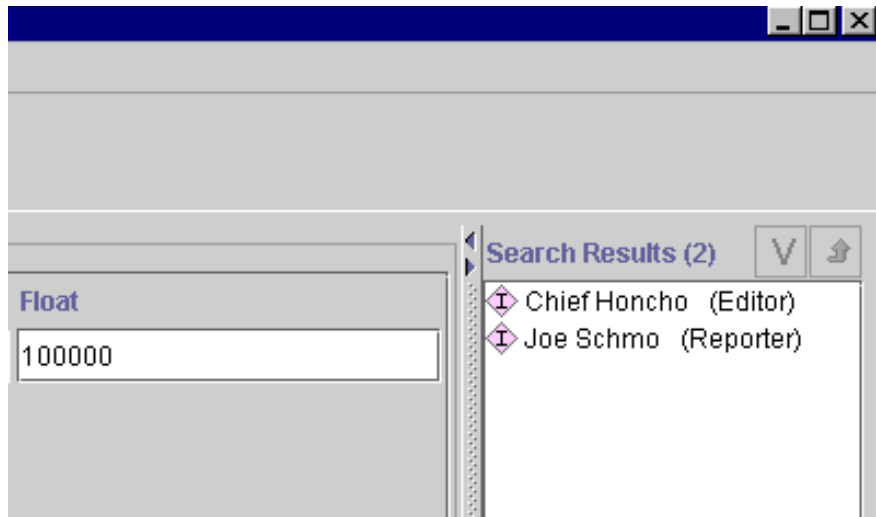
6. The menu to the right of the Slot entry bar is now active, and the entry bar at the far right of the window reminds you that the slot you have selected is a float. Select "**is greater than**" from the menu. Next, enter 100000 in the Float entry bar.



## Running the Query

Now you have set up your query, you can run it and view the results.

1. To run the query, first click the Find button near the bottom right of the window.
2. The results are shown in the Search Results window at the far right. If you cannot see the results, you may need to enlarge your window or move the slider bar.

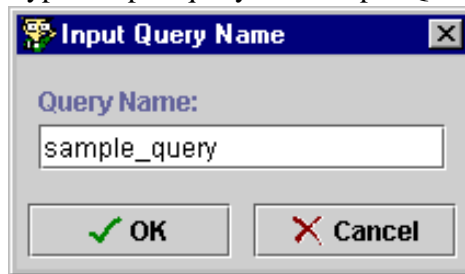


You can double-click on any instance to view details.

## Saving the Query

You can save any query before or after you run it. To save the query in the Query Library:

1. Click the Add to Query Library  button at the lower right of the screen.
2. Type sample\_query in the Input Query Name dialog box.

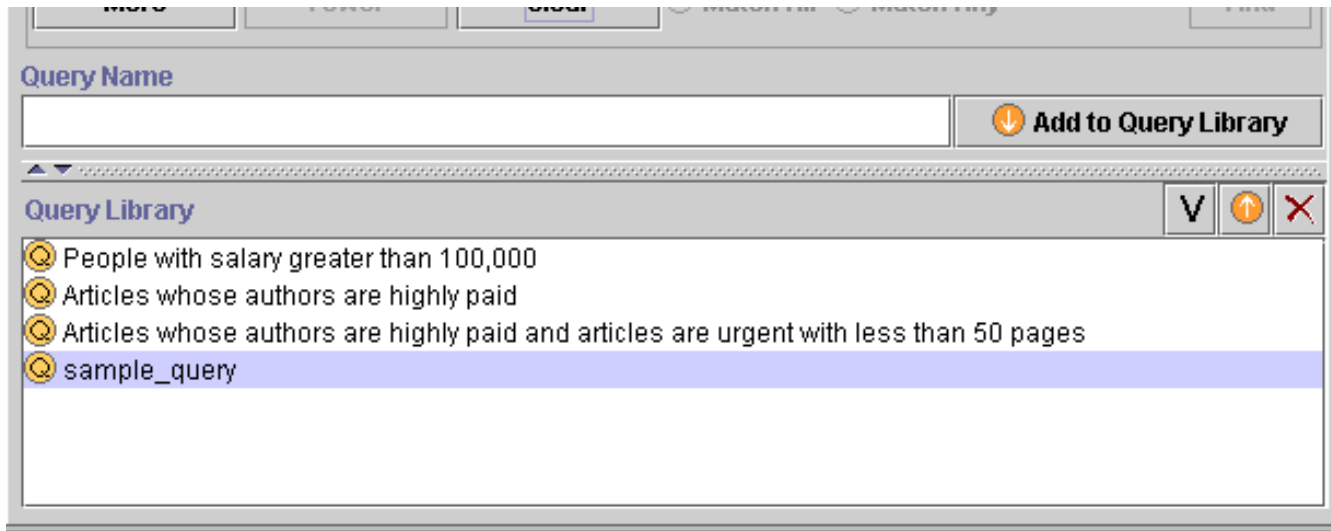



3. Click OK.

## Retrieving the Query

To retrieve a saved query, you can select it from the Query Library. To do this:

1. First, click the Clear button to empty the screen, so that you can see the query operation.
2. Select the query in the Query Library at the bottom of the screen. If you cannot see the Query Library, you may need to enlarge your window or move the slider bar.



3. Click the Retrieve Query  button to retrieve the query.
4. The saved query is now displayed at the top of the window. If you wish, you can now alter it or combine it with another saved query.
5. Click the Find button.