

This is a quick start guide for ASObj-C Database. For more detailed information please check the help file in the database (see image 2).

The basic idea behind this database, as pretty much all relational databases is searching. Most relational databases also have lots of reports but this database does not. I couldn't think of any helpful reports to do.

All the windows in the database have some type of page search except one. All but one window have live searches (the list of found items changes as the user types.) The different window has the old style search where you enter something and then search, or search again.

There are 2 main areas to work with when using the database. The windows in the database would be the most common place to work in with but there is also a ASObj-C menu (see image 2).

The class item window

The "Class items" window is where nearly all the work in the database is done. The classes and frameworks are only visible so the user can see what the classes and frameworks are connected to the "Class items" and to get a short description of them as well as any notes that would be helpful about classes or frameworks.

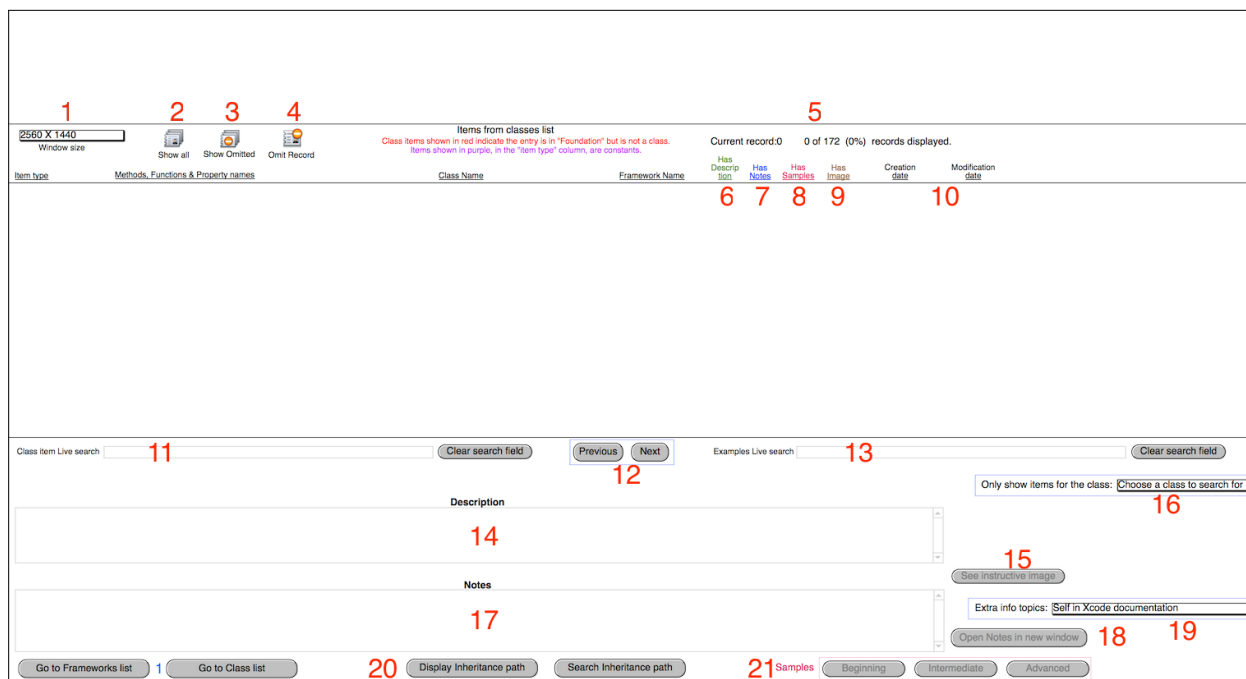


Figure 1: The "Class items" window

The red numbers in "figure 1" are labels for something in the "Class items" window. To find out what something does in the window look for the item in "figure 1," get the number, and use "List 1" to find out what the item does.

Note: After searches are preformed all records that match the search are displayed in the database. These records are referred to as the "found set" of records. The records that did not match the search requirements are what is called "Omitted" from the "found set" of records.

List 1: description of things in the window

Parts of the class item window (see figure 1)

- 1: Allows the screen size to be set for the database.
- 2: When clicked all the class item records will be displayed (showing what is called the "found set").
- 3: When clicked all the omitted records are displayed and the "found set" is hidden.
- 4: Causes the selected record to be removed from the "found set" and changed to an "omitted" record.
- 5: Displays the record status showing the current record number, how many records are being display out of the total number of records and show the percentage of records displayed.
- 6) Displays a **green** checkmark when the record has something entered in Description field.
- 7) Displays a **blue** checkmark when the record has something entered in Notes field.
- 8) Displays a **red** checkmark when the record has any samples.
- 9) Displays a **brown** checkmark when the record has has an image associated with the record. To display the image click the "See instructive image" button (labeled 15 in "Image 1").
- 10) This show the creation and modification date so the user can tell when the record was created and when it was last updated. This allows the user to quickly check if any of the information about a "Class item" have been changed recently.
- 11) Typing text into this field will cause the database to search for "Class items" that match the text typed so far. Each time a new character is typed, deleted or changed a new search is preformed.
- 12) This allows the currently selected record to be changed to the previous or next record depending on which button is clicked.
- 13) Typing text into this field will cause the database to search all the three types of samples (Beginning, Intermediate and Advanced) for what has been text typed so far. Each time a new character is typed, deleted or changed a new search is preformed.
- 14) This contains descriptive information about the class item.
- 15) Some records have a image associated with them that helps to make what is written in the notes or description fields more clear. Clicking this button will cause the image to be displayed. If the text in the button is gray then there is no image associated with the record. If the text is black then there is an image associated with the record.
- 16) This popup menu causes only the records that are about the selected class to be displayed.
- 17) This field is where helpful notes and other information is stored that did not fit into the description of the item.
- 18) Some notes fields contain a lot of information that is difficult to read in the tiny notes box. Clicking this button will cause a new window to be opened that contains the contents of the notes field. If there are no notes entered for the class item then the text in this button is gray. The text is black when there is information in the notes field.
- 19) There are some topics that are helpful to know about but do not fit directly into descriptions or notes about "Class items," classes or frameworks. This popup allows the user to see what topics are available and select a topic to view.
- 20) These 2 buttons help the user to understand class hierarchy in the frameworks better. This is useful to understand what classes a particular class inherits from. This is a more advanced operation and may not be of interested to people new to ASObj-C.
- 21) The three buttons "Beginning," "Intermediate" and "Advanced" allows the user to choose what type of sample to view. If a button has gray text then there are no samples for the type of sample that correspond to the button. If the text is black then there are samples of the desired type.

The ASObj-C menu

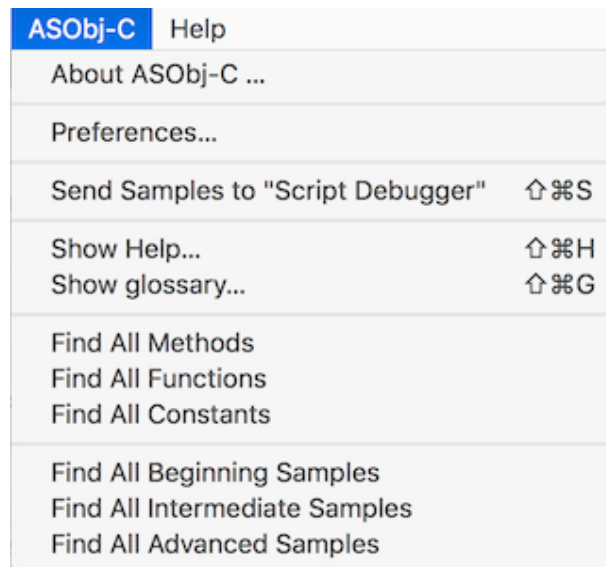


Image 2: The AsObj-C menu

When working with the database the "ASObj-C menu" has three key commands set up to:

- 1) Open the currently displayed example in the selected script editor (the default is Script Debugger).
- 2) Show the database help.
- 3) Show a glossary for words used in the database.

The menu items in the bottom of the menu can be used to execute preset searches in the database.

The preferences menu item can be used to set user preferences. Currently there is only 1 preference for the database and it determines which script editor (Script Debugger or Script Editor) is opened when you type shift-command-s (or select Send samples to" menu item) which allows the sample to be sent to to the chosen editor. After it opens in the editor it then automatically compiles the script sample.