



1 Installing kdb+

The workshop uses the kdb+ 32-bit version. It is a complete system and ideal for learning kdb+.

You can get the latest version from <http://kx.com/software-download.php>.

It is best to install kdb+ and the workshop material in the default directory given in the instructions. If you cannot do this, then define environment variable QHOME to point to the new directory. Note that some workshop material assumes the default installation.

Console and Studio for kdb+

When kdb+ is run, it displays a console where you can enter commands and see the results. This is all you need to follow the workshop, and if you just want to learn a little about kdb+, then it is easiest to work in the console.

In practice, kdb+ developers use one of the IDEs that connect to the console. A popular one is *Studio for kdb+* which is described below. If you expect to work more with kdb+, it is worth downloading and becoming familiar with this IDE.

Windows Path Separator

Windows users: please note that in a q session, you should always use / (forward slash) as the path separator, not \ (back slash).

Kx wiki

The workshop comes with a local copy of the Kx wiki, which includes the kdb+ reference. Open `Workshop/Wiki/wiki.html` in your browser to bring up the wiki. You might want to make a shortcut to this on your desktop.

The latest version of this local copy is always available at:
<http://code.kx.com/files/kxwiki.zip>.



1.1 Windows Install

1.1 Windows Install

Unzip qwindows.zip, taking care to retain the folder structure. Move the contents to directory `c:\q`. You should have:

```
C:\q          / main directory
C:\q\w32\q.exe / location of w32 executable
C:\q\Workshop / workshop files
```

First of all set the QHOME environmental variable and then add q to your PATH variable from the command line:

```
set QHOME=C:\q
setx QHOME "C:\q"

set PATH=%PATH%;%QHOME%\w32
setx PATH "%PATH%;%QHOME%\w32"
```

Now you should be able to simply type "q" in order to start a q session.

If q loads correctly, the console is displayed and prompts for input. You can type in expressions at the prompt and press Enter to run them. Try:

```
q)2 + 3 5
```

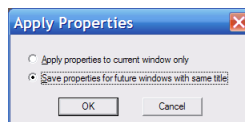
Enter `\\` to exit q:

```
q)\\          / exit q
```

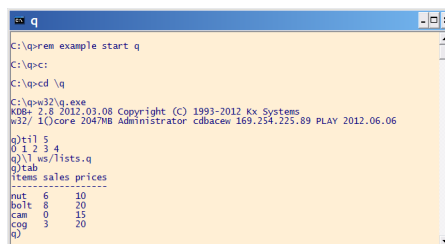
In the console, the up and down arrows scroll through and recall previous entries.

You can customize the Windows colors and font sizes by loading the console, then right-clicking the title bar and selecting Properties. After changing the properties, select *Save properties for future windows with same title*.

Since q source is in plain text files, it is worthwhile installing a good text editor such as notepad++ or notepad2.



For example:





1.2 Mac Install

Installing rlwrap

The Mac console works best if you have `rlwrap`, which allows line recall and editing in the terminal. To check for it, just enter `rlwrap` in a command terminal.

It can be installed with MacPorts or Homebrew:

```
sudo port install rlwrap    / MacPorts
brew install rlwrap        / Homebrew
```

The workshop Mac applications assume that `rlwrap` has been installed.

Installing kdb+

Unzip `qmac.zip` into your home directory, `~/q` (e.g. `/Users/elmo/q`).

You should have:

```
~/q          / main directory
~/q/m32/q     / location of m32 executable
~/q/Workshop  / workshop files
```

Loading q

It is possible to update your `.profile` (or `.bash_profile`) with the following lines in order to excute `q` by simply typing `"q"`:

```
export QHOME=~/q
export PATH=$PATH:$QHOME/m32
alias q="rlwrap q"
```

After editing type this command:

```
source ~/.profile
or
source ~/.bash_profile
```

Finally simply type `"q"` in order to start a `q` session.

If `q` loads correctly, it displays a console and issues a prompt, and you can type in expressions at the prompt and press Enter to run them. Try:

```
q)2 + 3 5
```

Enter `\\` to exit `q`:

```
q)\\          / exit q
```



1.3 Linux Install

1.3 Linux Install

Installing rlwrap

The Linux console works best if you have `rlwrap`, which allows line recall and editing in the terminal. To check for it, just enter `rlwrap` in a command terminal.

It can be installed with your usual Linux installer, e.g. using `aptitude`:

```
apt-get install rlwrap
```

The workshop shell scripts assume that `rlwrap` has been installed.

Installing kdb+

Unzip `qlinux.zip` into your home directory, i.e. `~/q`.

You should have:

```
~/q          / main directory
~/q/l32/q    / location of l32 executable
~/q/Workshop / workshop files
```

Loading q

It is possible to update your `.bashrc` (or `.bash_profile`) with the following lines in order to excute `q` by simply typing `"q"`:

```
export QHOME=~/q
export PATH=$PATH:$QHOME/m32
alias q="rlwrap q"
```

After editing type this command:

```
source ~/.bashrc
or
source ~/.bash_profile
```

Finally simply type `"q"` in order to start a `q` session.

If `q` loads correctly, it displays a console and issues a prompt, and you can type in expressions at the prompt and press Enter to run them. Try:

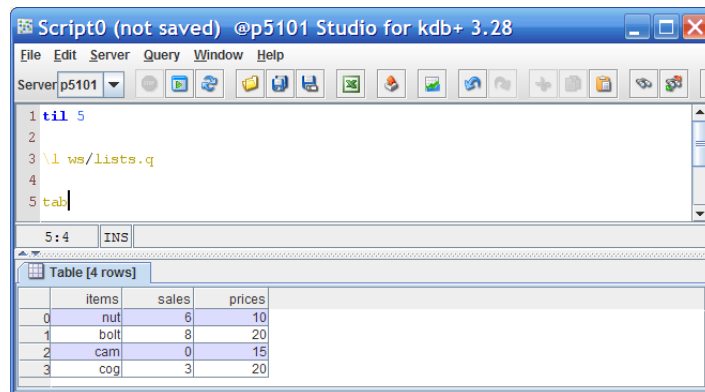
```
q)2 + 3 5
```

Enter `\\` to exit `q`:

```
q)\\          / exit q
```

1.4 Studio for kdb+

Once you are familiar with the console, you can try *Studio for kdb+*, which is at `Workshop/Studio/studio.jar`. This lets you open scripts and run them line by line. Tables are shown in a grid, as below.



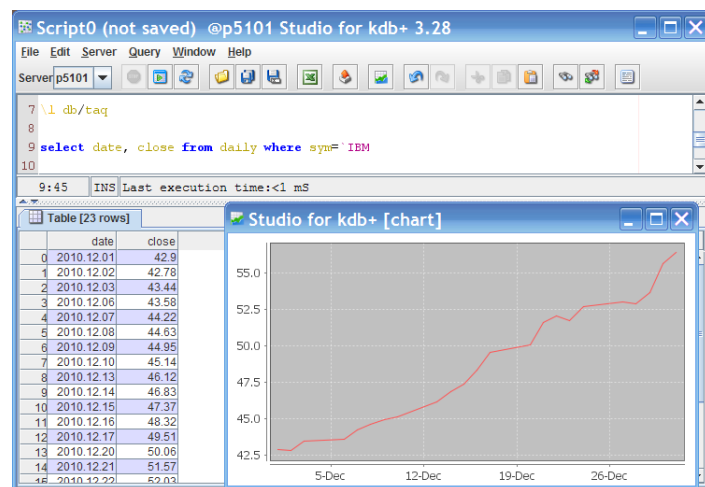
The screenshot shows the Studio for kdb+ 3.28 interface. The script editor contains the following code:

```
1 til 5
2
3 \l ws/lists.q
4
5 tab
```

The table displayed below the script is:

	items	sales	prices
0	nut	6	10
1	bolt	8	20
2	cam	0	15
3	cog	3	20

Data can also be plotted:



Studio for kdb+ connects to a q server which can be anywhere on the network, for example, a remote server. To use it, first start a kdb+ console, and set it to listen on a given port, e.g. `load kdb+, then listen on port 5000:`

```
q)\p 5000
```

You can then start Studio for kdb+ and connect to the server on port 5000.