Wonsole

The new web console for power users

User Manual

NTNU – Trondheim
Norwegian University of
Science and Technology

■netlight

CUSTOMER DRIVEN PROJECT
November 19, 2012
Team: Ivo Dlouhy, Martin Havig, Øystein Heimark, Oddvar Hungnes

Contents

List of Figures

Chapter 1

Introduction

This document describes how to use the Wonsole application. It contains a user tutorial as well as command refence.

Chapter 2

Basics

User interface for wonsole is minimal a simple, so that the funcionality gets presented in a clear way. If you see the main screen in the picture 2.1, on the left hand side, there is a GUI - Graphical User Interface - this is where data get presented as graphics. On the right hand side, there is a console window. This is a place where user can input the commands and send the commands to the system by pressing enter.

By default the list of available databases is displayed in GUI and command prompt in console.

authors

books

Figure 2.1: Main screen

Chapter 3

Tutorial

3.1 Database

The most basic command in Wonsole is the command to open, or switch database. It has this format:

```
db DATABASE
```

Where DATABASE is the name of the database to open. Databases are listed in GUI and they can be opened by clicking at the name too.

After issuing the command, the database contents are displayed in GUI, see picture 3.1. Wonsole uses document oriented database, so it is list of documents indexed by numbers. From each document the title attribute is displayed as preview.

Same command can be used to switch the database anywhere in the workflow as in picture 3.2.

Note: All uncommited changes are lost when switching the database.

```
0: title:"Romeo and Juliet"

1: title:"Julius Caesar"

2: title:"Hamlet"

3: title:"Macbeth"

4: title:"A Doll's House"

5: title:"Peer Gynt"
```

```
wonsole2> db books
wonsole2> |
```

Figure 3.1: Database open

```
0: title:"Henrik Ibsen"

1: title:"William Shakespeare"
```

```
wonsole2> db books
wonsole2> db authors
wonsole2>
```
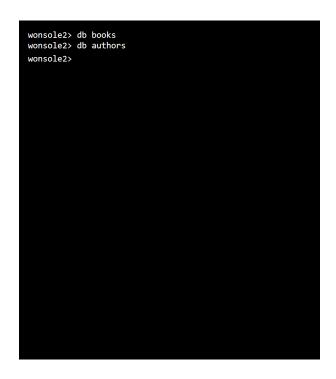
Figure 3.2: Database switch

3.2 Log

Note: Developers only.

3.3 Print

Command print is available to display data in plaintext instead of using GUI. It's syntax is very simple - it has one attribute: variable or data to print.

```
print DATA
```

Command loads the data, converts it into text and prints in the console as ssen in picture 3.3.

```
                                        wonsole2> print 'Hello world!'
          authors                       "Hello world!"
                                        wonsole2>
          books
```

Figure 3.3: Print

3.4 View

Oposite of the print command there is view. It displays the complex variable in the GUI.

```
view VARIABLE
```

3.5 JavaScript evaluatuion

Wonsole includes complete JavaScript language to the command set. If user issues acommand, that is not recognized as the Wonsole command, this input gets evaluated as JavaScript. You can for example issue a helper command, that does some required functionality not supported by Wonsole and than continue using default Wonsole commands, JavaScript language is 100

```
var new = "New variable" //Javascript
print variable
"New variable"
```

This is an example of creating a new JavaScript variable and using it in simple Wonsole command, demonstration on picture 3.4.

Note: This is advanced functionality. JavaScript language is very complex and it is the user full responsibility to use it in a safe, suitable way.

```
authors

books
```

```
wonsole2> var variable = "New variable"
wonsole2> print variable
"New variable"
wonsole2>
```

Figure 3.4: Javascript integration

3.6   Quiet

When Wonsole displays a list of documents saved in the database (see command db) by default it uses quiet mode. In quiet mode, it displays only object index and value of the title attribute as a object identification as you can see in picture 3.5.

```
quiet
```

You can toggle preview mode by using command quiet. When toggled off, all attributes of the object are displayed - but only the simple data types. If the atribute is complex, the Object keyword is displayed instead, see picture 3.6.

```
0: title:"Romeo and Juliet"

1: title:"Julius Caesar"

2: title:"Hamlet"

3: title:"Macbeth"

4: title:"A Doll's House"

5: title:"Peer Gynt"
```

```
wonsole2> db books
wonsole2>
```

Figure 3.5: Quiet mode on

```
0: title:"Romeo and Juliet",

  price:100,

  language:"en",

  tags:Object,

  stock:5

1: title:"Julius Caesar",

  price:100,

  language:"en",

  tags:Object,

  stock:5

2: title:"Hamlet",

  price:170,

  language:"en",

  tags:Object,

  stock:7

3: title:"Macbeth",
```

```
wonsole2> db books
wonsole2> quiet
wonsole2>
```

Figure 3.6: Quiet mode off

3.7 Navigtion in list of documents

```
0: title:"Romeo and Juliet"

1: title:"Julius Caesar"

2: title:"Hamlet"

3: title:"Macbeth"

4: title:"A Doll's House"

5: title:"Peer Gynt"
```

```
wonsole2> db books
wonsole2> print docs
[{"_id":"0b432ce4fb079967bbe5320f33003c50","_rev":"19-
4f3cffa172aca6b413c39865c838bf9c","title":"Romeo and
Juliet","price":100,"language":"en","tags":
["tragedy","pre1800","play"],"stock":5},
{"_id":"0b432ce4fb079967bbe5320f33003fa3","_rev":"19-
c55a2b1dffdeb8426a2fe1079bb06b3c","title":"Julius
Caesar","price":100,"language":"en","tags":
["tragedy","pre1800","play"],"stock":5},
{"_id":"0b432ce4fb079967bbe5320f330049a5","_rev":"17-
f964f928dcafca2802195fc2868c3611","title":"Hamlet","price":1
["tragedy","pre1800","play"],"stock":7},
{"_id":"0b432ce4fb079967bbe5320f330055c0","_rev":"17-
de1c60db46d267b3538c4eac5e33c8e5","title":"Macbeth","price":
["tragedy","pre1800","play","onsale"],"stock":0},
{"_id":"0b432ce4fb079967bbe5320f33006ce0","_rev":"12-
6086f6f8fcff5337083a78839a6fe957","title":"A Doll's
House","year":1879,"tags":
["boring","toolong","post1800","nora"],"language":"no","pric
{"_id":"0b432ce4fb079967bbe5320f33006e34","_rev":"11-
b05670b2552056cdf5eef59afd38d8dd","title":"Peer
Gynt","year":1876,"tags":
["mediumboring","post1800"],"language":"no","price":299,"stc
wonsole2>
```

Figure 3.7: Docs array

```
0: title:"Romeo and Juliet"

1: title:"Julius Caesar"

2: title:"Hamlet"

3: title:"Macbeth"

4: title:"A Doll's House"

5: title:"Peer Gynt"
```

```
wonsole2> db books
wonsole2> print docs
[{"_id":"0b432ce4fb079967bbe5320f33003c50","_rev":"19-
4f3cffa172aca6b413c39865c838bf9c","title":"Romeo and
Juliet","price":100,"language":"en","tags":
["tragedy","pre1800","play"],"stock":5},
{"_id":"0b432ce4fb079967bbe5320f33003fa3","_rev":"19-
c55a2b1dffdeb8426a2fe1079bb06b3c","title":"Julius
Caesar","price":100,"language":"en","tags":
["tragedy","pre1800","play"],"stock":5},
{"_id":"0b432ce4fb079967bbe5320f330049a5","_rev":"17-
f964f928dcafca2802195fc2868c3611","title":"Hamlet","price":1
["tragedy","pre1800","play"],"stock":7},
{"_id":"0b432ce4fb079967bbe5320f330055c0","_rev":"17-
de1c60db46d267b3538c4eac5e33c8e5","title":"Macbeth","price":
["tragedy","pre1800","play","onsale"],"stock":0},
{"_id":"0b432ce4fb079967bbe5320f33006ce0","_rev":"12-
6086f6f8fcff5337083a78839a6fe957","title":"A Doll's
House","year":1879,"tags":
["boring","toolong","post1800","nora"],"language":"no","pric
{"_id":"0b432ce4fb079967bbe5320f33006e34","_rev":"11-
b05670b2552056cdf5eef59afd38d8dd","title":"Peer
Gynt","year":1876,"tags":
["mediumboring","post1800"],"language":"no","price":299,"stc
wonsole2> print docs[0]
{"_id":"0b432ce4fb079967bbe5320f33003c50","_rev":"19-
4f3cffa172aca6b413c39865c838bf9c","title":"Romeo and
Juliet","price":100,"language":"en","tags":
["tragedy","pre1800","play"],"stock":5}
wonsole2>
```

Figure 3.8: Docs indexing

Object

title: Julius Caesar

price: 100

language: en

   tags:

      0: tragedy

      1: pre1800

      2: play

stock: 5

```
wonsole2> db books
wonsole2> doc 0
wonsole2> doc 1
wonsole2>
```

Figure 3.9: Document opening and switching

Object

title: Julius Caesar

price: 100

language: en

   tags:

      0: tragedy

      1: pre1800

      2: play

stock: 5

```
wonsole2> db books
wonsole2> doc 0
wonsole2> doc 1
wonsole2> print doc
{"_id":"0b432ce4fb079967bbe5320f33003fa3","_rev":"19-
c55a2b1dffdeb8426a2fe1079bb06b3c","title":"Julius
Caesar","price":100,"language":"en","tags":
["tragedy","pre1800","play"],"stock":5}
wonsole2>
```

Figure 3.10: Doc object

Object

```
title: Romeo and Juliet
price: 100
language: en
    tags:
        0: tragedy
        1: pre1800
        2: play
stock: 4
available: true
```

```
wonsole2> db books
wonsole2> doc 0
wonsole2> doc.stock = 4
wonsole2> doc.available = true
wonsole2> |
```

Figure 3.11: Modification of doc

3.8   Transactions: Commit and Rollback

Object

title: Romeo and Juliet

price: 100

language: en

   tags:

      0: tragedy

      1: pre1800

      2: play

stock: 4

available: true

```
wonsole2> db books
wonsole2> doc 0
wonsole2> doc.stock = 4
wonsole2> doc.available = true
wonsole2> commit
```

Figure 3.12: Commit

Object

title: Romeo and Juliet

price: 100

language: en

   tags:

      0: tragedy

      1: pre1800

      2: play

stock: 5

```
wonsole2> db books
wonsole2> doc 0
wonsole2> doc.stock = 4
wonsole2> doc.available = true
wonsole2> rollback
wonsole2> doc 0
wonsole2>
```

Figure 3.13: Rollback

3.9 Operating with objects: Add and Remove

0: title: "Romeo and Juliet"

1: title: "Julius Caesar"

2: title: "Hamlet"

3: title: "Macbeth"

4: title: "A Doll's House"

5: title: "Peer Gynt"

6:

7: title: "New book"

```
wonsole2> db books
wonsole2> add {}
wonsole2> add {"title":"New book"}
wonsole2>
```

Figure 3.14: Add

Object

title: New book

note: Not yet available

```
wonsole2> db books
wonsole2> add {"title" : "New book"}
wonsole2> doc 6
wonsole2> doc.note = "Not yet available"
wonsole2>
```

Figure 3.15: Add and modify

Chapter 4

Appendices

4.1   Command reference

command	description	example
db DATABASE	opens database DATABASE	db books
docs	switches view to documents	docs
doc INDEX	opens document at INDEX	doc 0
commit	commits the changes to database	commit
rollback	rollbacks the changes	rollback
refresh	refreshes the UI	refresh
clear	clears the console	clear
print	VARIABLE prints the variable VARIABLE	print docs
quiet	toggles list verbosity, default: on	quiet
log	toggles log visibility, default: off	log
remove INDEX	removes object at index INDEX	remove 0
remove JSON	removes object specified by JSON	remove {"id":1231232}
add JSON	adds object specified by JSON	add {"title" : "New book"}
seteach ARRAY ATTRIBUTE VALUE	sets (creates if necessary) ATTRIBUTE in object in ARRAY to VALUE	seteach docs author "Unknown"
foreach ARRAY CODE	executes the CODE for each item of ARRAY	foreach docs price=price*10
filter ARRAY ATTRIBUTE VALUE	filters the ARRAY with VALUE of ATTRIBUTE	filter docs title Report
script ARRAY	executes the ARRAY of commands specified by string	script ["clear", "db books"]

Table 4.1: Command reference