



Rexx L.A. 2020 Online Symposium NetRexx and BSF4ooRexx Code Examples for Redis (Using the Jedis Java Client)

by Tony Dycks - **Revised**: September 24, 2020



Overview of Presentation - I

- What is Redis DB?
- Advantages of Redis DB over DBM Data Stores & Memcached
- Tech Stack Applications of Redis in the Industry
- Installing Redis Db on a Linux OS Platform
- Installing Redis Db on a Windows Workstation
- Starting Up the Redis Server
- Using the Redis CLI Client
- Documentation Sources



Overview of Presentation - II

- Redis Clients for Programming Languages
- Udemy Online Courses on Redis
- Other Web Resources for Learning Redis
- Installing the Jedis Java Client for Redis
- Java Based Jedis Tutorials
- Net Rexx Code Examples using the Jedis Client
- BSF4ooRexx Code Examples using Jedis Client
- Findings and Recommendations

What is Redis DB?

- Open Source In-memory Data Structure Store
- Can be used as a Data store, Cache or Message Broker
- Basically, a No SQL Key / Value Store with a variety of ways to store data values
- Improvement over DBMs in that it handles a number of different structures vs. String based Key/Value DBMs
- Home Page for Redis DB
 - <https://redis.io/>

Advantages of Redis Over DBMs

- Goes beyond the scope of a DBM Structure such as GDBM or QDBM
- Server that is More Sophisticated than a DBM Server such as Tokyo Tyrant
- Data Replication is Supported; Configurable
- Logging is Supported; also Configurable
- In memory caching makes it faster to retrieve and store data
- Storage Indexing for Data Repository (From 0 to 15); 16 Numbered Repositories on one Redis Db Server

Advantages of Redis Over Memcached

- Both Redis and Memcached Store Data In Memory
- Memcached storage is volatile; complete data loss if the software is restarted or if a machine loses power
- Redis will persist its memory data to the hard disk or storage device periodically
 - Configurable, but it does this automatically even with a basic default installation
- Memcached is not available for native Windows OS
- Redis available in binary packages for Windows, Linux & Mac
- Super Easy to Install; Memcached requires Source Code Extraction, Configure and Make to Build Environment

Disadvantages of Redis Db

- Scaling is difficult when compared to other No SQL Databases such as MongoDB or Apache Cassandra
- Index Values are the Root Level of DB Store Categorization; Developer needs to keep track of what index stores what type of data and the data structure(s) used
- Native Windows Binary Installation is a bit out of date; Newer functionality would require a Source Code build of the Redis Db software
- Other No SQL databases such as MongoDB & Cassandra do not have the index storage limitations
- MongoDB can use Collections within Multiple Databases; Redis Db does not Support that

Data Contents Structures for Redis

- Goes beyond the scope of a DBM Structure such as GDBM or QDBM
- Data Value Structures Supported
 - Strings
 - Hashes with Multiple Keys and Values
 - Lists
 - Sets and Sorted Sets with Range Queries
 - Bitmaps
 - Hyperlogs
 - Geospatial Indexing (Map Location Applications)
 - Streams (with Redis v5.0 and up)

Applications of Redis Db

- **Who uses Redis?**
 - Uber, Airbnb, Pinterest, Shopify, Twitter,
 - Reddit, Udemy, Instagram, Instacart
 - Over 4600 Companies in their tech stacks
- **Source:** Redis Story – Reviews, Pros & Cons
<https://stackshare.io/ampstories/redis>

Applications of Redis Db +

- **Cloud Storage Integrations:**
 - Presto, Clever Cloud, Google Cloud, Spring Data, Net Data, Heroku Redis, Cloud 66, Clever Cloud, Boundary, plus more
- **Docker Container:**
 - Docker Container of Redis Db Available on Docker Hub (Docker Account required)
 - **URL:** <https://hub.docker.com/>

Installing Redis Db on Linux OS Platforms

- For Most Linux Distros, Redis Db can be installed from the Command Shell with your Package Manager
 - **Fedora / CentOS**
 - `sudo yum install redis`
 - **Debian** Family Distros (Debian, Ubuntu, Mint)
 - `sudo apt install redis`
 - Verify that both the `redis-server` and `redis-tools` are installed as pre-requisites
 - **Arch Linux** Family Distros (Arch, Manjaro)
 - `sudo pacman -S redis`

Installing Redis Db on Windows

- If you are not concerned about the whether you have a current Version of Redis, you can install a Redis binary on Windows 7 or 10 from the following web link:
 - **URL:**
<https://github.com/microsoftarchive/redis/releases/tag/win-3.2.100>
 - This Version is v3.2 which is far behind the current Linux Stable Version of 6.0
 - Use only if you do not need the newer functionality of Redis Db.

Starting Up The Redis Db Server

- Depending on the install the Redis Db Server may or may not be Installed as a Service
- If the Redis Db Server service is not running use the following command to start the Redis Db Server:
 - **redis-server** <Enter>
 - The Redis Db Server will default to Port 6379 as the port of connection without any customization
- **Source:** Redis Quickstart Guide
- **URL:** <https://redis.io/topics/quickstart>

Testing The Availability of the Redis DB Server

- Pinging the Redis DB Server; issue the Shell Command:
 - **redis-cli ping** <Enter>
 - Response should be:
 - **PONG**
- To invoke the Redis Db Client:
 - **redis-cli** <Enter>

Using the Redis Db CLI Client

- To start up of the Redis Db Command Line Client:
 - **redis-cli** <Enter>
- A prompt reflecting your localhost address and default port should be displayed
 - **127.0.0.1:6379>**
- To Select a Given Database Index:
 - select <index-value> {value from 0 to 15}
 - **select 15**
- Prompt will reflect the index in Square Brackets
 - **127.0.0.1:6379[15]>**





Documentation on Redis Db

- Redis Website - Base URL for Documentation:
 - <https://redis.io/documentation>
- Full List of Redis CLI Commands:
 - <https://redis.io/commands>
- Tutorials and FAQs:
 - Introduction to Redis Db Data Types
 - <https://redis.io/topics/data-types-intro>

Redis Clients for Programming Languages - I.

- Redis Client APIs are Available for The Following Key Programming Languages
 - C#, Go, **Java**, Lua, Node.js, Perl, PHP, and Python
- Names of Some of the Better Redis API Clients
 - C# => ServiceStack.Redis
 - Go => Radix
 - **Java** => **Jedis**, lettuce, Redisson
 - Lua => redis-lua
 - Node.js => node_redis
 - Perl => Redis
 - PHP => phpredis, Predis
 - Python => redis-py

Redis Clients for Programming Languages - II.

- Redis Website Page with List of Redis APIs
 - **URL:** <https://redis.io/clients> (Source for Quotations below)
 -  - Entries flagged with a “**Star**” are Recommended Clients for use
 -  - Entries flagged with a “**Smiley Face**” are Clients APIs that have had development activity within the last 6 months
- Recommended Java Based Redis Clients
 - **Jedis** - “A blazingly small and and sane redis java client”
 - **lettuce** - “Advanced Redis client for thread-safe sync, async, and reactive usage. Supports Cluster, Sentinel, Pipelining, and codecs.”
 - **Redisson** - “distributed and scalable Java data structures on top of Redis server”

Courses on Redis



- **Learn Redis**

- Jesse Boyer – Rating of 2.9
- Course covered Client APIs for the Programming Languages PHP and Python
- <https://www.udemy.com/course/learn-redis/>

- **Learn Redis from Scratch** (Recommended)

- Eduonix – Brad Traversy – Rating of 4.3
- Better Coverage of Redis Server and CLI vs. “Learn Redis”
- Web Based Node.js Programming Example of Contact Info Data Entry & Display Web Application
- <https://www.udemy.com/course/learn-redis-from-scratch/>

Other Web Resources for Learning Redis

- **Command Reference - redis**

- Web page with references to CLI Commands
- Search can be Filtered by Command Group
- <https://redis.io/commands>

- **The Little Redis Book**

- **Author:** Karl Seguin
- Open Source PDF Book
- **Download Site**
- <https://github.com/karlseguin/the-little-redis-book>

Installing the Jedis Java Client for Redis

- **Java** Installation Used for Code Examples
 - Open JDK 1.8 (Linux 32/64Bit Intel & RPi4)
 - Oracle Java SE 8 (Windows 10 32-Bit)
- **NetRexx** Versions: v3.03 and v3.08
- **ooRexx** Versions: v4.2 and v5.0 Releases
- **BSF4ooRexx**: BSF 641.20200130
- **Windows** Versions: 7 &10 Pro 32 & 64 Bit
- **Linux Distros**: CentOS, Debian, Ubuntu, Mint, Manjaro, Sparky, Raspbian

Installing the Jedis Java Client for Redis +

- **Jedis** Jar File Version (Currently 3.3.0)
- Maven Repository Website
- <https://mvnrepository.com/artifact/redis.clients/jedis/3.3.0>
- Copy **jedis-3.3.0.jar** File to the `$JAVA_HOME/jre/lib/ext` Directory
- **\$JAVA_HOME** should point to the JDK that your NetRexx and BSF4ooRexx Installs Reference
- For Java Developers using Maven ...
 - Setup JAVA Classpath to Reference Jar File
 - Add Dependency to the Maven POM File

Java Based Jedis Tutorials



- **Baeldung.com - Intro to Jedis - the Java Redis Client Library**
- <https://www.baeldung.com/jedis-java-redis-client-library>
- Covers Methods for Strings, Lists, Sets, Sorted Sets and Hashes (Sections 5.1 thru 5.5)
- Connection Pooling & Clusters (Sections 9 & 10)
- Code Samples and Relevant Articles:
- <https://github.com/eugenp/tutorials/tree/master/persistence-modules/redis>

Java Based Jedis Tutorials+

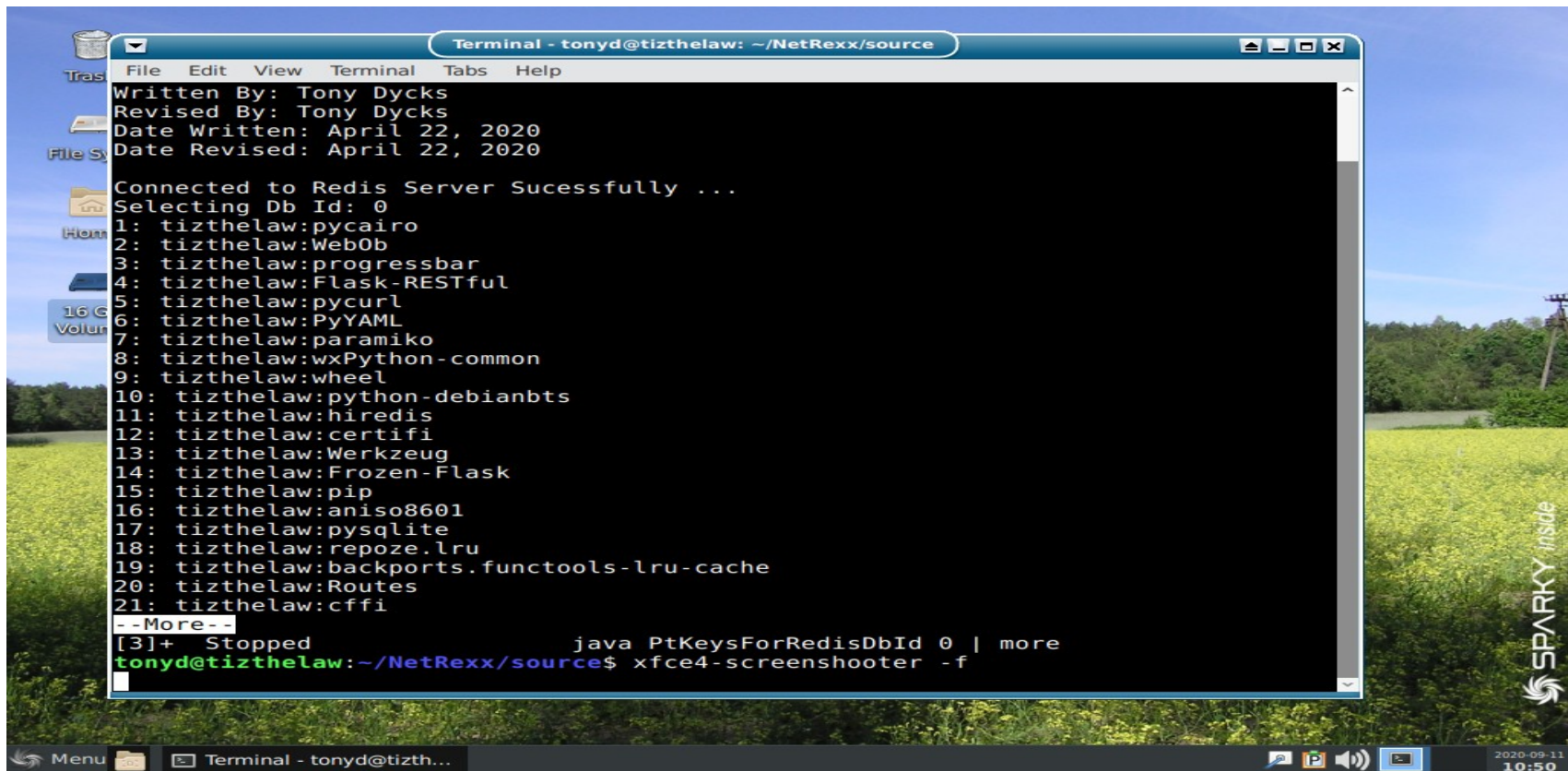
- **Baeldung.com - List All Available Redis Keys**
- Section 4.2 covers a Java Jedis Client Code Solution to Storing Set Data using the Irange Method of the Jedis Client
- Section 7 covers Scanning the Redis Db data using Java
- Solution uses a Hash Structure to Store The Data and an Iterator to Read The Data
- <https://www.baeldung.com/redis-list-available-keys>

NetRexx Code Examples using the Jedis Client - I

- **Program:** PtKeysForRedisDbId.nrx
- Display List of Redis DB Keys on the Console Shell for a Numeric Database Id Specified on the Command Line
- Command Line Parameter Value from 0 to 15
- In this example Db Id 0 contains a Repository of Installed Python 2.7 PIP Modules with Version Info
- **Key Convention:** <hostname>:<pip-package>
- Syntax to Run the Compiled NetRexx Program:
 - **java PtKeysForRedisDbId 0 <Enter>**

NetRexx Code Examples using the Jedis Client - II

- Console Shell Output of Run:

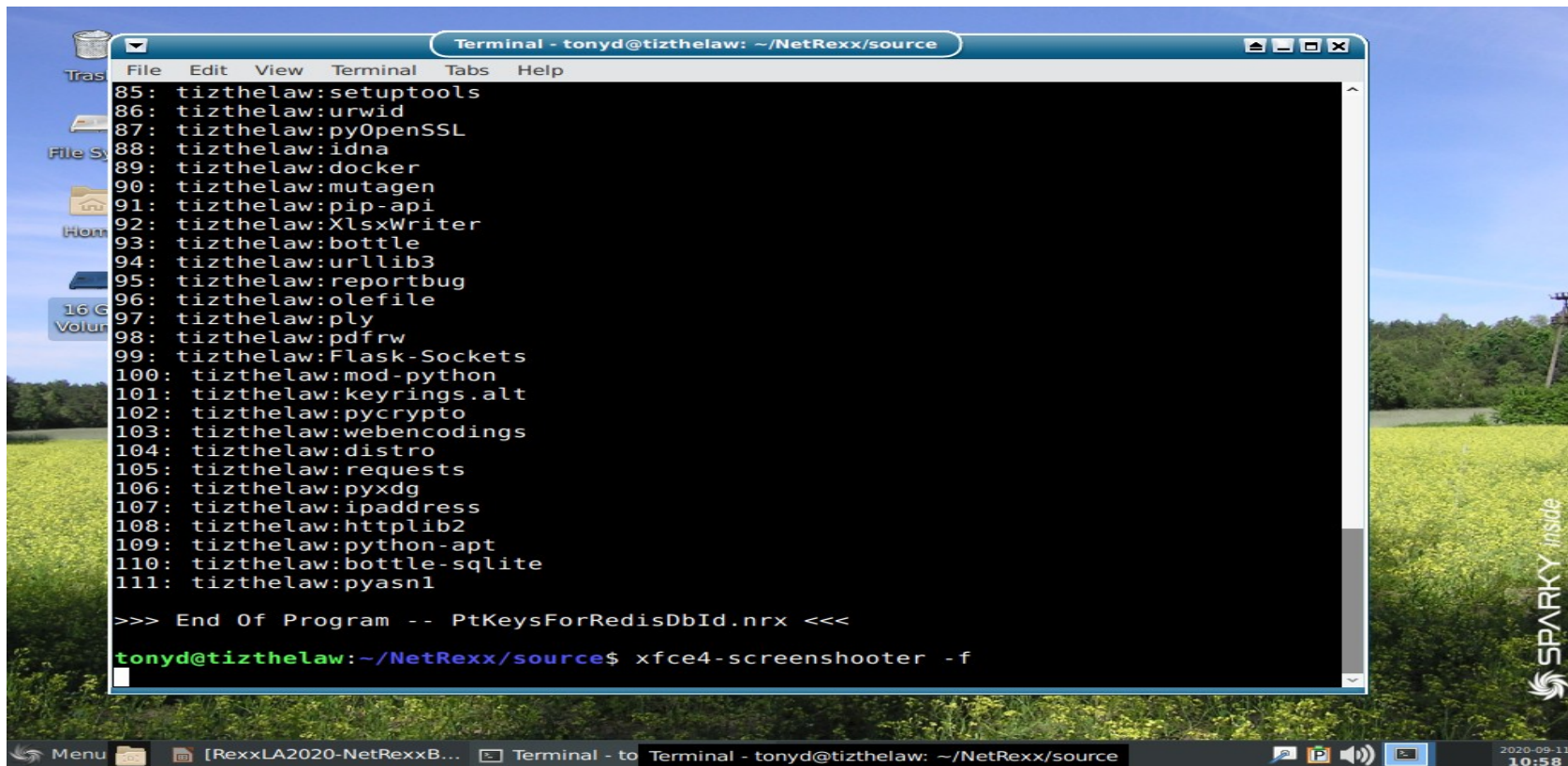


```
Terminal - tonyd@tizthelaw: ~/NetRexx/source
File Edit View Terminal Tabs Help
Written By: Tony Dycks
Revised By: Tony Dycks
Date Written: April 22, 2020
Date Revised: April 22, 2020
Connected to Redis Server Successfully ...
Selecting Db Id: 0
1: tizthelaw:pycairo
2: tizthelaw:Web0b
3: tizthelaw:progressbar
4: tizthelaw:Flask-RESTful
5: tizthelaw:pycurl
6: tizthelaw:PyYAML
7: tizthelaw:paramiko
8: tizthelaw:wxPython-common
9: tizthelaw:wheel
10: tizthelaw:python-debianbts
11: tizthelaw:hiredis
12: tizthelaw:certifi
13: tizthelaw:Werkzeug
14: tizthelaw:Frozen-Flask
15: tizthelaw:pip
16: tizthelaw:aniso8601
17: tizthelaw:pysqlite
18: tizthelaw:repoze.lru
19: tizthelaw:backports.functools-lru-cache
20: tizthelaw:Routes
21: tizthelaw:cffi
--More--
[3]+ Stopped java PtKeysForRedisDbId 0 | more
tonyd@tizthelaw:~/NetRexx/source$ xfce4-screenshooter -f
```

SPARKY inside

NetRexx Code Examples using the Jedis Client - III

- Console Shell Display of Program Run End



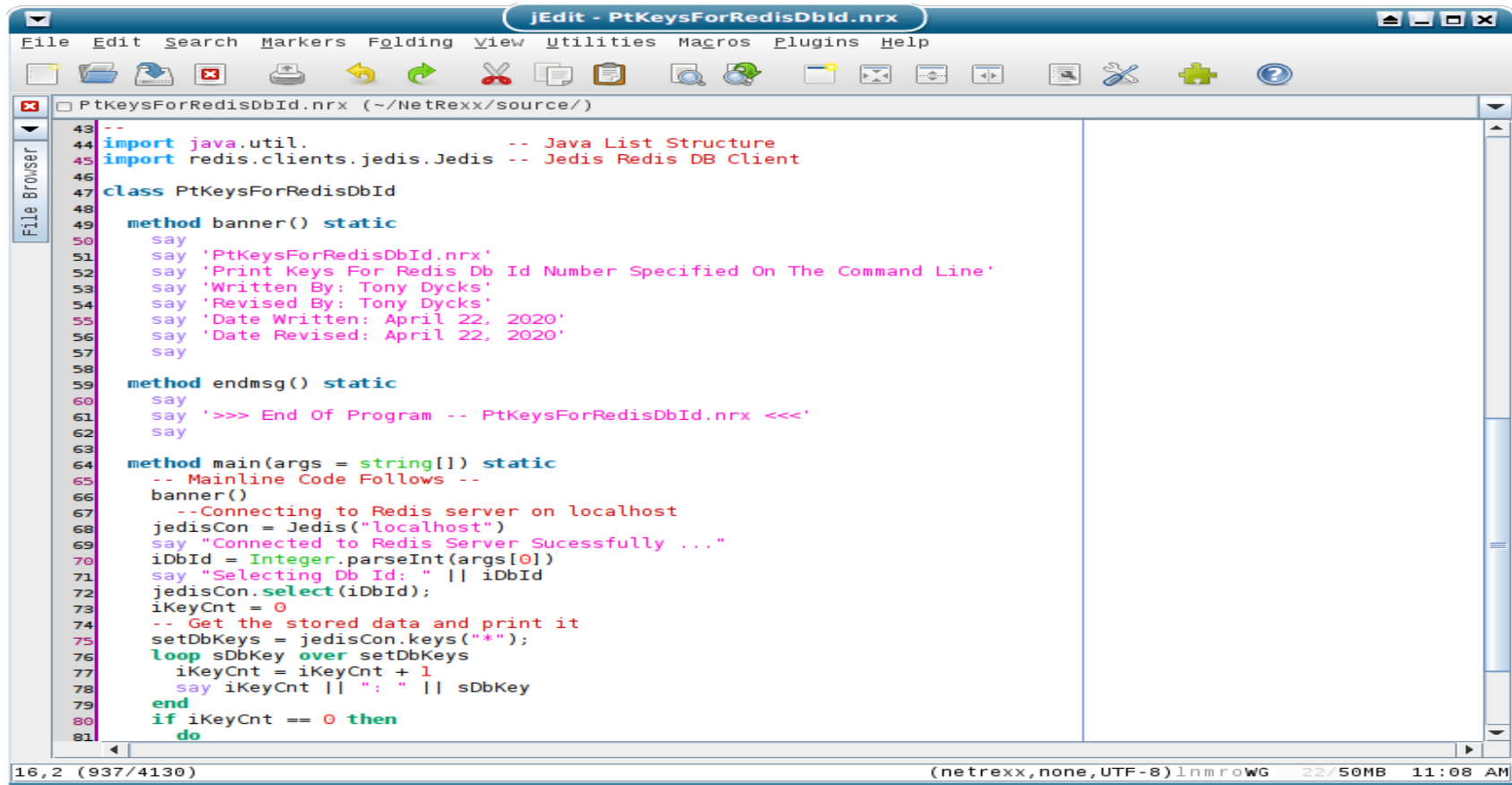
The image shows a terminal window titled "Terminal - tonyd@tizthelaw: ~/NetRexx/source". The terminal displays a list of 111 NetRexx code examples, each on a new line, starting with a line number (85-111) and the code name. The list includes: setuptools, urwid, pyOpenSSL, idna, docker, mutagen, pip-api, XlsxWriter, bottle, urllib3, reportbug, olefile, ply, pdfwr, Flask-Sockets, mod-python, keyrings.alt, pycrypto, webencodings, distro, requests, pyxdg, ipaddress, httplib2, python-apt, bottle-sqlite, and pyasn1. Below the list, the terminal shows the message ">>> End Of Program -- PtKeysForRedisDbId.nrx <<<". The prompt is "tonyd@tizthelaw:~/NetRexx/source\$". The user has entered the command "xfce4-screenshooter -f". The terminal window is overlaid on a desktop background of a green field under a blue sky. The desktop environment includes a menu bar at the top, a sidebar on the left with icons for Trash, File System, Home, and 16 GB Volume, and a taskbar at the bottom with a menu icon, window icons, and system tray icons. The system tray shows the date "2020-09-11" and time "10:58". A vertical watermark "SPARKY inside" is visible on the right side of the terminal window.

```
Terminal - tonyd@tizthelaw: ~/NetRexx/source
File Edit View Terminal Tabs Help
85: tizthelaw:setuptools
86: tizthelaw:urwid
87: tizthelaw:pyOpenSSL
88: tizthelaw:idna
89: tizthelaw:docker
90: tizthelaw:mutagen
91: tizthelaw:pip-api
92: tizthelaw:XlsxWriter
93: tizthelaw:bottle
94: tizthelaw:urllib3
95: tizthelaw:reportbug
96: tizthelaw:olefile
97: tizthelaw:ply
98: tizthelaw:pdfwr
99: tizthelaw:Flask-Sockets
100: tizthelaw:mod-python
101: tizthelaw:keyrings.alt
102: tizthelaw:pycrypto
103: tizthelaw:webencodings
104: tizthelaw:distro
105: tizthelaw:requests
106: tizthelaw:pyxdg
107: tizthelaw:ipaddress
108: tizthelaw:httplib2
109: tizthelaw:python-apt
110: tizthelaw:bottle-sqlite
111: tizthelaw:pyasn1

>>> End Of Program -- PtKeysForRedisDbId.nrx <<<
tonyd@tizthelaw:~/NetRexx/source$ xfce4-screenshooter -f
```

NetRexx Code Examples using the Jedis Client - IV

- NetRexx Code Snippet



```
jEdit - PtKeysForRedisDbId.nrx
File Edit Search Markers Folding View Utilities Macros Plugins Help
PtKeysForRedisDbId.nrx (~/NetRexx/source/)
43 --
44 import java.util.           -- Java List Structure
45 import redis.clients.jedis.Jedis -- Jedis Redis DB Client
46
47 class PtKeysForRedisDbId
48
49 method banner() static
50 say
51 say 'PtKeysForRedisDbId.nrx'
52 say 'Print Keys For Redis Db Id Number Specified On The Command Line'
53 say 'Written By: Tony Dycks'
54 say 'Revised By: Tony Dycks'
55 say 'Date Written: April 22, 2020'
56 say 'Date Revised: April 22, 2020'
57 say
58
59 method endmsg() static
60 say
61 say '>>> End Of Program -- PtKeysForRedisDbId.nrx <<<'
62 say
63
64 method main(args = string[]) static
65 -- Mainline Code Follows --
66 banner()
67 --Connecting to Redis server on localhost
68 jedisCon = Jedis("localhost")
69 say "Connected to Redis Server Successfully ..."
70 iDbId = Integer.parseInt(args[0])
71 say "Selecting Db Id: " || iDbId
72 jedisCon.select(iDbId);
73 iKeyCnt = 0
74 -- Get the stored data and print it
75 setDbKeys = jedisCon.keys("*");
76 loop sDbKey over setDbKeys
77 iKeyCnt = iKeyCnt + 1
78 say iKeyCnt || ": " || sDbKey
79 end
80 if iKeyCnt == 0 then
81 do
16,2 (937/4130) (netrexx,none,UTF-8)lnmrowg 22/50MB 11:08 AM
```

NetRexx Code Examples using the Jedis Client - V

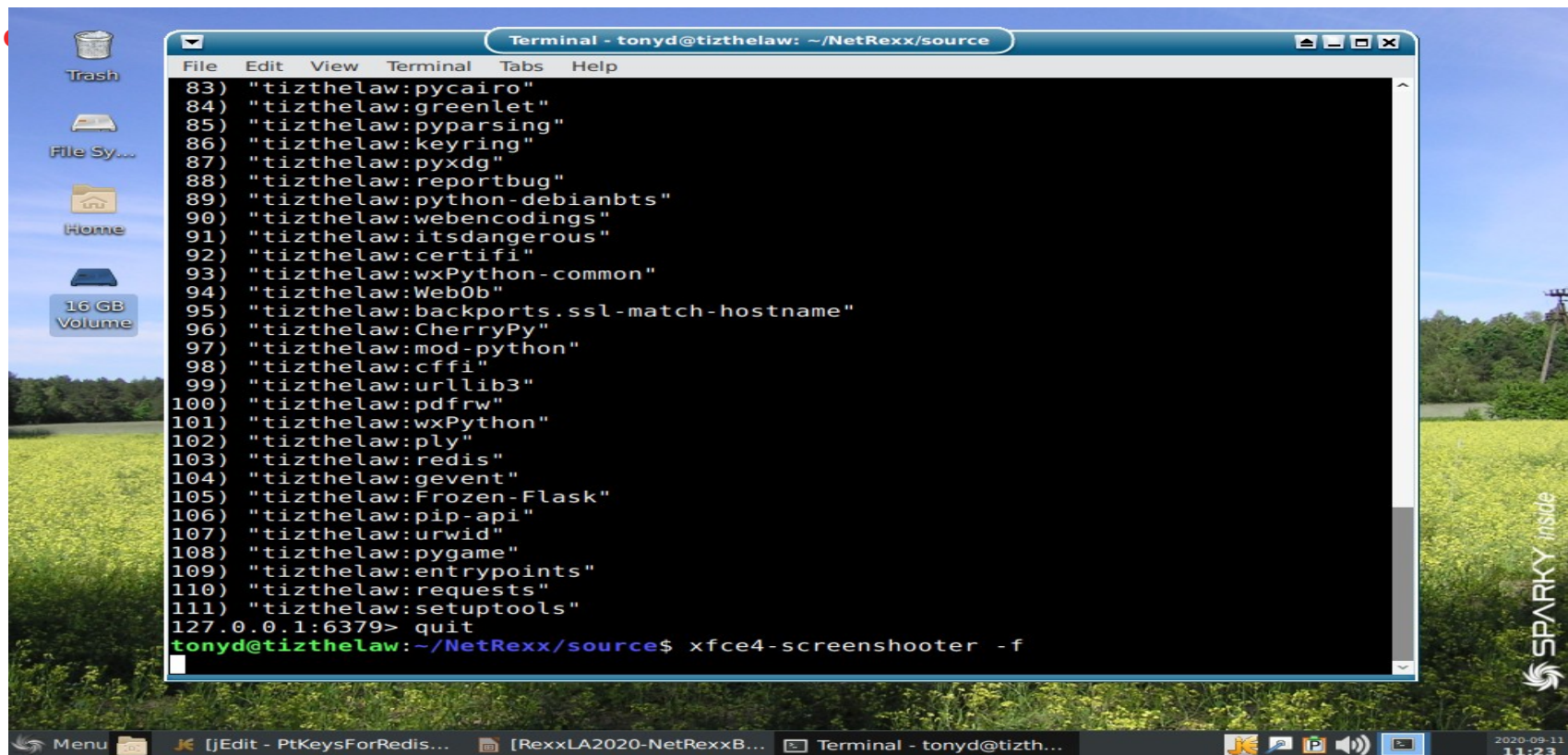
- Website Reference for Java Code Example used in this NetRexx Program
- TutorialsPoint.com -- Redis - Java
 - Redis Java Keys Example
- **Link:**
- https://www.tutorialspoint.com/redis/redis_java.htm

NetRexx Code Examples using the Jedis Client - VI

- Validating The Program Output via the Redis CLI Client
- From a Command Shell Prompt:
 - **redis-cli** <Enter>
- From Redis CLI Prompt (127.0.0.1:6379>):
 - **keys *** <Enter>
- To Exit the Redis CLI Client:
 - **quit** <Enter>

NetRexx Code Examples using the Jedis Client - VII

- Screenshot of Keys Listing End from Redis CLI



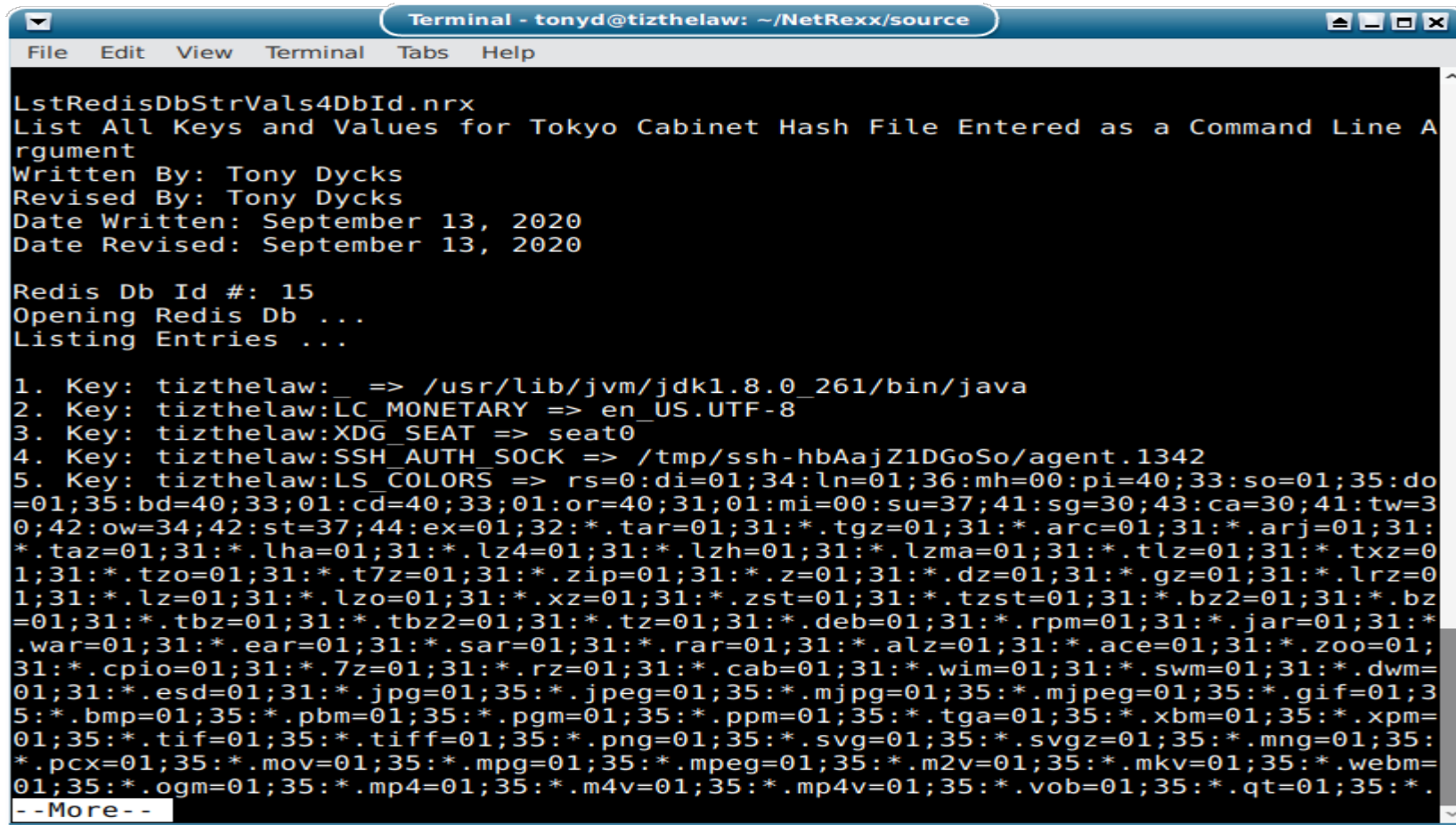
```
Terminal - tonyd@tizthelaw: ~/NetRexx/source
File Edit View Terminal Tabs Help
83) "tizthelaw:pycairo"
84) "tizthelaw:greenlet"
85) "tizthelaw:pyparsing"
86) "tizthelaw:keyring"
87) "tizthelaw:pyxdg"
88) "tizthelaw:reportbug"
89) "tizthelaw:python-debianbts"
90) "tizthelaw:webencodings"
91) "tizthelaw:itsdangerous"
92) "tizthelaw:certifi"
93) "tizthelaw:wxPython-common"
94) "tizthelaw:WebOb"
95) "tizthelaw:backports.ssl-match-hostname"
96) "tizthelaw:CherryPy"
97) "tizthelaw:mod-python"
98) "tizthelaw:cffi"
99) "tizthelaw:urllib3"
100) "tizthelaw:pdfw"
101) "tizthelaw:wxPython"
102) "tizthelaw:ply"
103) "tizthelaw:redis"
104) "tizthelaw:gevent"
105) "tizthelaw:Frozen-Flask"
106) "tizthelaw:pip-api"
107) "tizthelaw:urwid"
108) "tizthelaw:pygame"
109) "tizthelaw:entrypoints"
110) "tizthelaw:requests"
111) "tizthelaw:setuptools"
127.0.0.1:6379> quit
tonyd@tizthelaw:~/NetRexx/source$ xfce4-screenshooter -f
```

NetRexx Code Examples using the Jedis Client - VIII

- **Program:** LstRedisDbStrVals4DbId.nrx
- List String Values for All Db Keys for a String Key / Value Redis Db. Db Id to list is Entered as a Command Line Argument from **0** to **15**
- Redis CLI Command Demo'd: **get** <dbkey>
- **Input:** Redis Db Id previously loaded with **set** commands (Linux Environment Variables)
- Syntax to Run Compiled Class:
 - **java LstRedisDbStrVals4DbId 15**

NetRexx Code Examples using the Jedis Client - IX

- Console Shell Output of Run Start



```
Terminal - tonyd@tizthelaw: ~/NetRexx/source
File Edit View Terminal Tabs Help

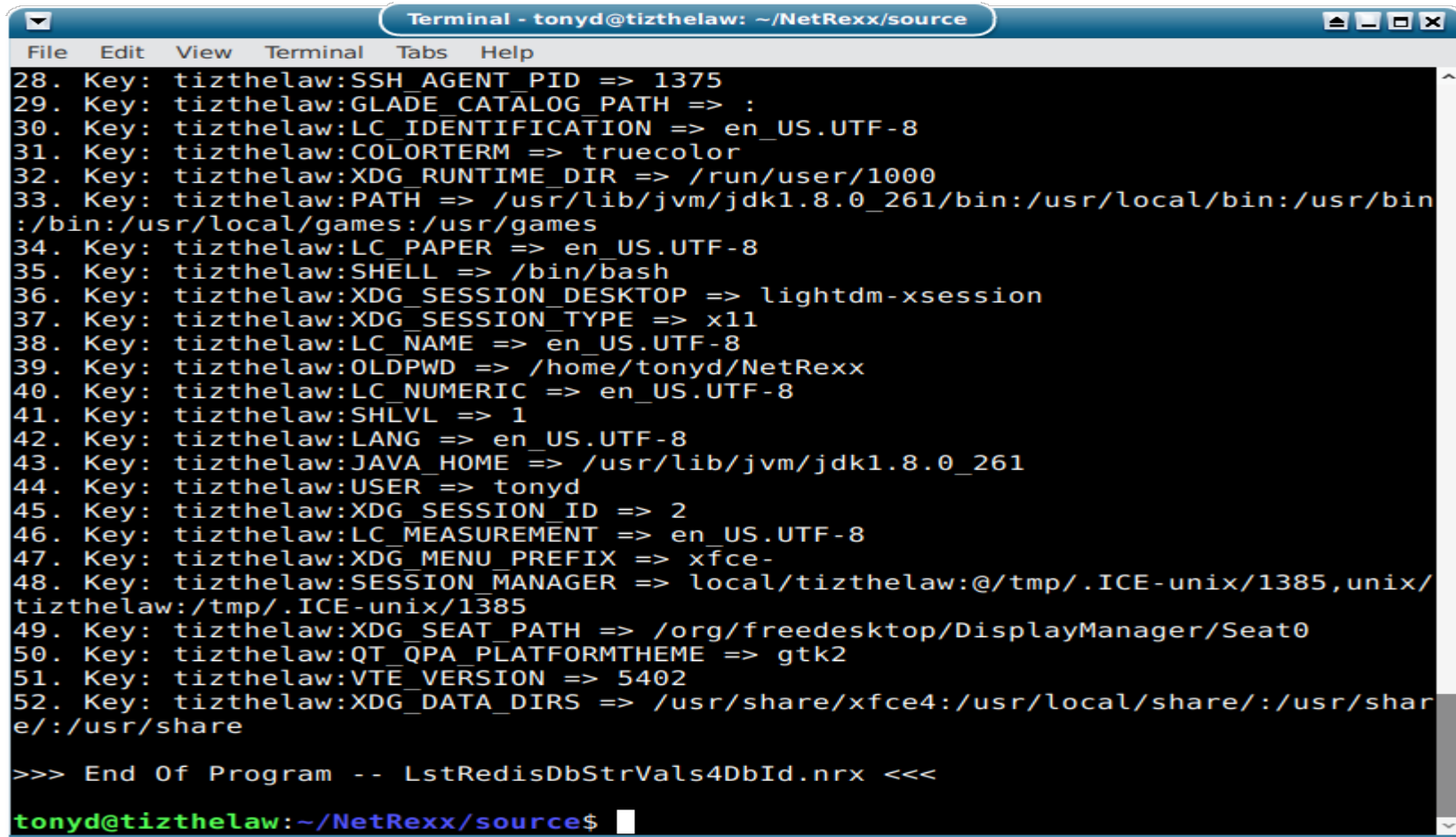
LstRedisDbStrVals4DbId.nrx
List All Keys and Values for Tokyo Cabinet Hash File Entered as a Command Line Argument
Written By: Tony Dycks
Revised By: Tony Dycks
Date Written: September 13, 2020
Date Revised: September 13, 2020

Redis Db Id #: 15
Opening Redis Db ...
Listing Entries ...

1. Key: tizthelaw:_ => /usr/lib/jvm/jdk1.8.0_261/bin/java
2. Key: tizthelaw:LC_MONETARY => en_US.UTF-8
3. Key: tizthelaw:XDG_SEAT => seat0
4. Key: tizthelaw:SSH_AUTH_SOCK => /tmp/ssh-hbAajZ1DGoSo/agent.1342
5. Key: tizthelaw:LS_COLORS => rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.wim=01;31:*.swm=01;31:*.dwm=01;31:*.esd=01;31:*.jpg=01;35:*.jpeg=01;35:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.
--More--
```

NetRexx Code Examples using the Jedis Client - X

- Console Shell Output of Run End

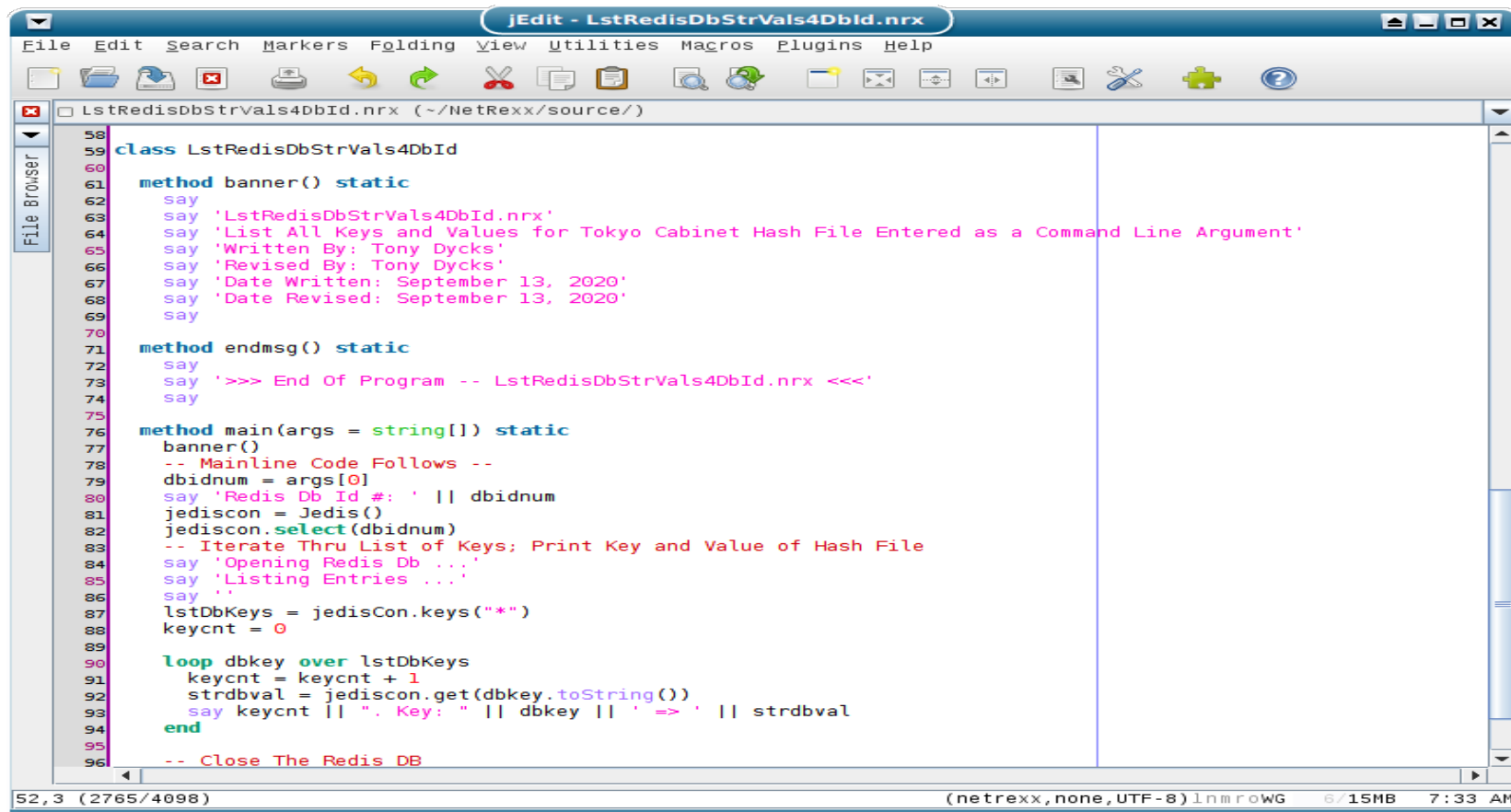


```
Terminal - tonyd@tizthelaw: ~/NetRexx/source
File Edit View Terminal Tabs Help
28. Key: tizthelaw:SSH_AGENT_PID => 1375
29. Key: tizthelaw:GLADE_CATALOG_PATH => :
30. Key: tizthelaw:LC_IDENTIFICATION => en_US.UTF-8
31. Key: tizthelaw:COLORTERM => truecolor
32. Key: tizthelaw:XDG_RUNTIME_DIR => /run/user/1000
33. Key: tizthelaw:PATH => /usr/lib/jvm/jdk1.8.0_261/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
34. Key: tizthelaw:LC_PAPER => en_US.UTF-8
35. Key: tizthelaw:SHELL => /bin/bash
36. Key: tizthelaw:XDG_SESSION_DESKTOP => lightdm-xsession
37. Key: tizthelaw:XDG_SESSION_TYPE => x11
38. Key: tizthelaw:LC_NAME => en_US.UTF-8
39. Key: tizthelaw:OLDPWD => /home/tonyd/NetRexx
40. Key: tizthelaw:LC_NUMERIC => en_US.UTF-8
41. Key: tizthelaw:SHLVL => 1
42. Key: tizthelaw:LANG => en_US.UTF-8
43. Key: tizthelaw:JAVA_HOME => /usr/lib/jvm/jdk1.8.0_261
44. Key: tizthelaw:USER => tonyd
45. Key: tizthelaw:XDG_SESSION_ID => 2
46. Key: tizthelaw:LC_MEASUREMENT => en_US.UTF-8
47. Key: tizthelaw:XDG_MENU_PREFIX => xfce-
48. Key: tizthelaw:SESSION_MANAGER => local/tizthelaw:@/tmp/.ICE-unix/1385,unix/tizthelaw:/tmp/.ICE-unix/1385
49. Key: tizthelaw:XDG_SEAT_PATH => /org/freedesktop/DisplayManager/Seat0
50. Key: tizthelaw:QT_QPA_PLATFORMTHEME => gtk2
51. Key: tizthelaw:VTE_VERSION => 5402
52. Key: tizthelaw:XDG_DATA_DIRS => /usr/share/xfce4:/usr/local/share:/usr/share:/usr/share

>>> End Of Program -- LstRedisDbStrVals4DbId.nrx <<<
tonyd@tizthelaw:~/NetRexx/source$
```

NetRexx Code Examples using the Jedis Client - XI

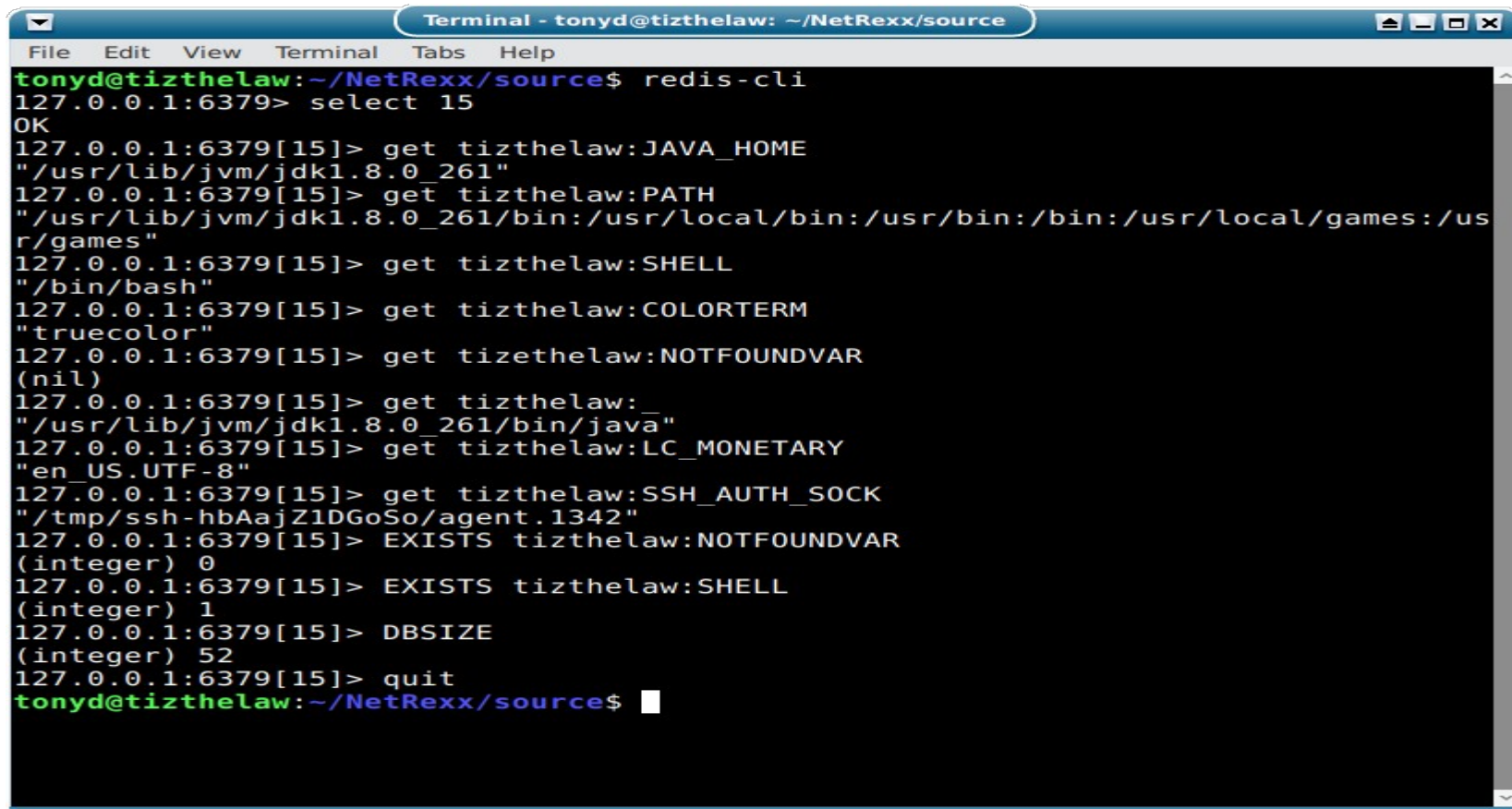
- NetRexx Code Snippet



```
58 class LstRedisDbStrVals4DbId
59
60 method banner() static
61 say
62 say 'LstRedisDbStrVals4DbId.nrx'
63 say 'List All Keys and Values for Tokyo Cabinet Hash File Entered as a Command Line Argument'
64 say 'Written By: Tony Dycks'
65 say 'Revised By: Tony Dycks'
66 say 'Date Written: September 13, 2020'
67 say 'Date Revised: September 13, 2020'
68 say
69
70 method endmsg() static
71 say
72 say '>>> End Of Program -- LstRedisDbStrVals4DbId.nrx <<<'
73 say
74
75 method main(args = string[]) static
76 banner()
77 -- Mainline Code Follows --
78 dbidnum = args[0]
79 say 'Redis Db Id #: ' || dbidnum
80 jediscon = Jedis()
81 jediscon.select(dbidnum)
82 -- Iterate Thru List of Keys; Print Key and Value of Hash File
83 say 'Opening Redis Db ...'
84 say 'Listing Entries ...'
85 say ''
86 lstDbKeys = jediscon.keys("*")
87 keycnt = 0
88
89 loop dbkey over lstDbKeys
90 keycnt = keycnt + 1
91 strdbval = jediscon.get(dbkey.toString())
92 say keycnt || ". Key: " || dbkey || ' => ' || strdbval
93 end
94
95 -- Close The Redis DB
96
```

NetRexx Code Examples using the Jedis Client - XII

- Redis CLI Validation of Db Id: 15



```
Terminal - tonyd@tizthelaw: ~/NetRexx/source
File Edit View Terminal Tabs Help
tonyd@tizthelaw:~/NetRexx/source$ redis-cli
127.0.0.1:6379> select 15
OK
127.0.0.1:6379[15]> get tizthelaw:JAVA_HOME
"/usr/lib/jvm/jdk1.8.0_261"
127.0.0.1:6379[15]> get tizthelaw:PATH
"/usr/lib/jvm/jdk1.8.0_261/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games"
127.0.0.1:6379[15]> get tizthelaw:SHELL
"/bin/bash"
127.0.0.1:6379[15]> get tizthelaw:COLORTERM
"truecolor"
127.0.0.1:6379[15]> get tizthelaw:NOTFOUNDVAR
(nil)
127.0.0.1:6379[15]> get tizthelaw:_
"/usr/lib/jvm/jdk1.8.0_261/bin/java"
127.0.0.1:6379[15]> get tizthelaw:LC_MONETARY
"en_US.UTF-8"
127.0.0.1:6379[15]> get tizthelaw:SSH_AUTH_SOCK
"/tmp/ssh-hbAajZ1DGoSo/agent.1342"
127.0.0.1:6379[15]> EXISTS tizthelaw:NOTFOUNDVAR
(integer) 0
127.0.0.1:6379[15]> EXISTS tizthelaw:SHELL
(integer) 1
127.0.0.1:6379[15]> DBSIZE
(integer) 52
127.0.0.1:6379[15]> quit
tonyd@tizthelaw:~/NetRexx/source$
```

BSF4ooRexx Code Examples using the Jedis Client - I

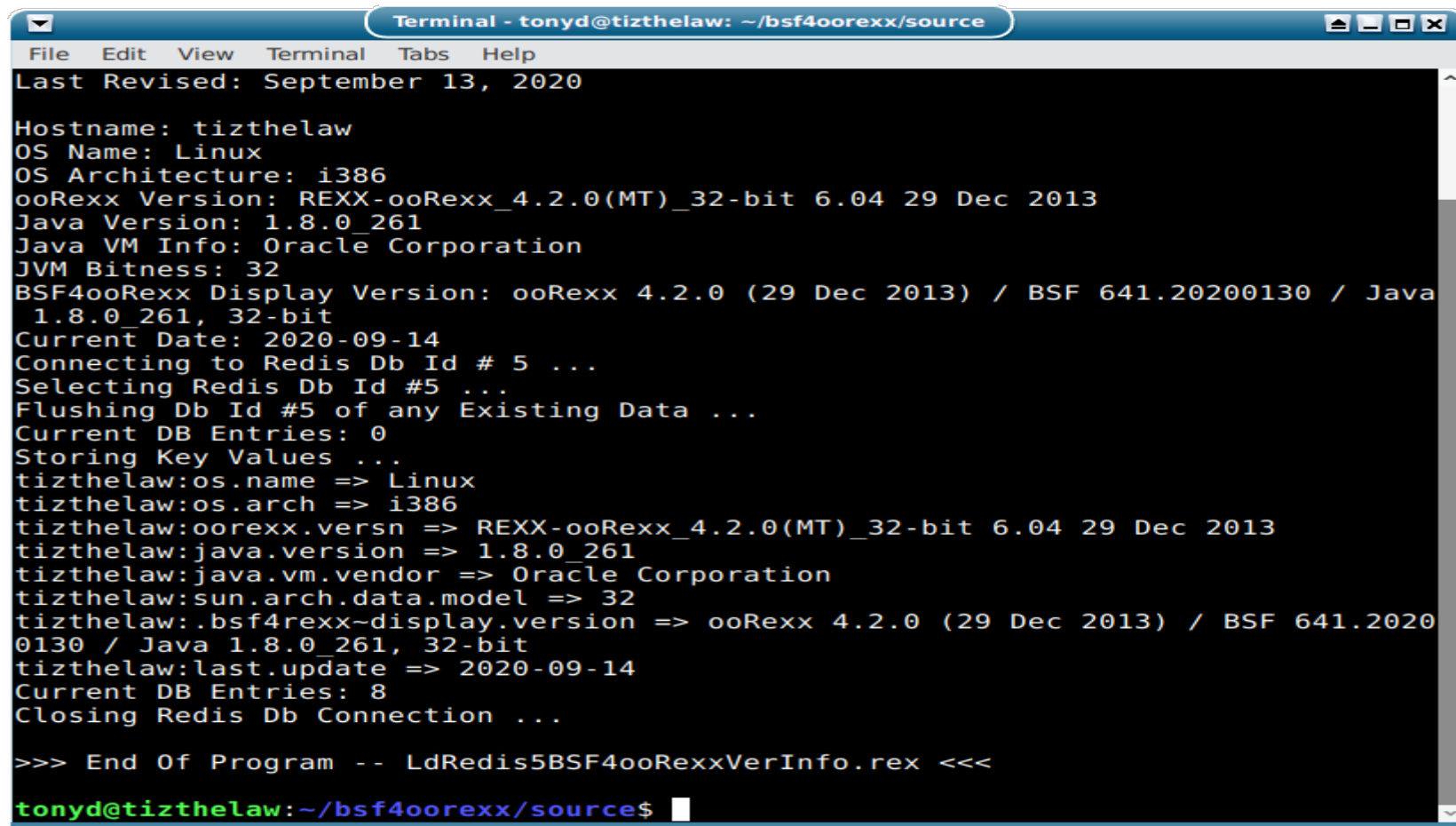
- **Program:** LdRedis5BSF4ooRexxVerInfo.rex
- Load Redis Db Id #5 with Java, ooRexx and BSF4ooRexx Version Information Properties and Methods
- **Input:** Java and Rexx Version Info from Properties and Functions
- **Output:** Redis Db Id #5 Will Contain The String Key / Value Records
- **Key Convention:** <hostname>:<property>
- **Validation:** Use redis-cli Client to Verify Data

BSF4ooRexx Code Examples using the Jedis Client - II

- **Linux Comand Shell Run Syntax:**
- `sh ./rexxj2.sh LdRedis5BSF4ooRexxVerInfo.rex`
- **Shell Environment:** Bash
- **Shell Output:** Program Status Message
Displays and Record Counts
- Sample Run Example Follows ...

BSF4ooRexx Code Examples using the Jedis Client - III

- Comand Shell Run Output:



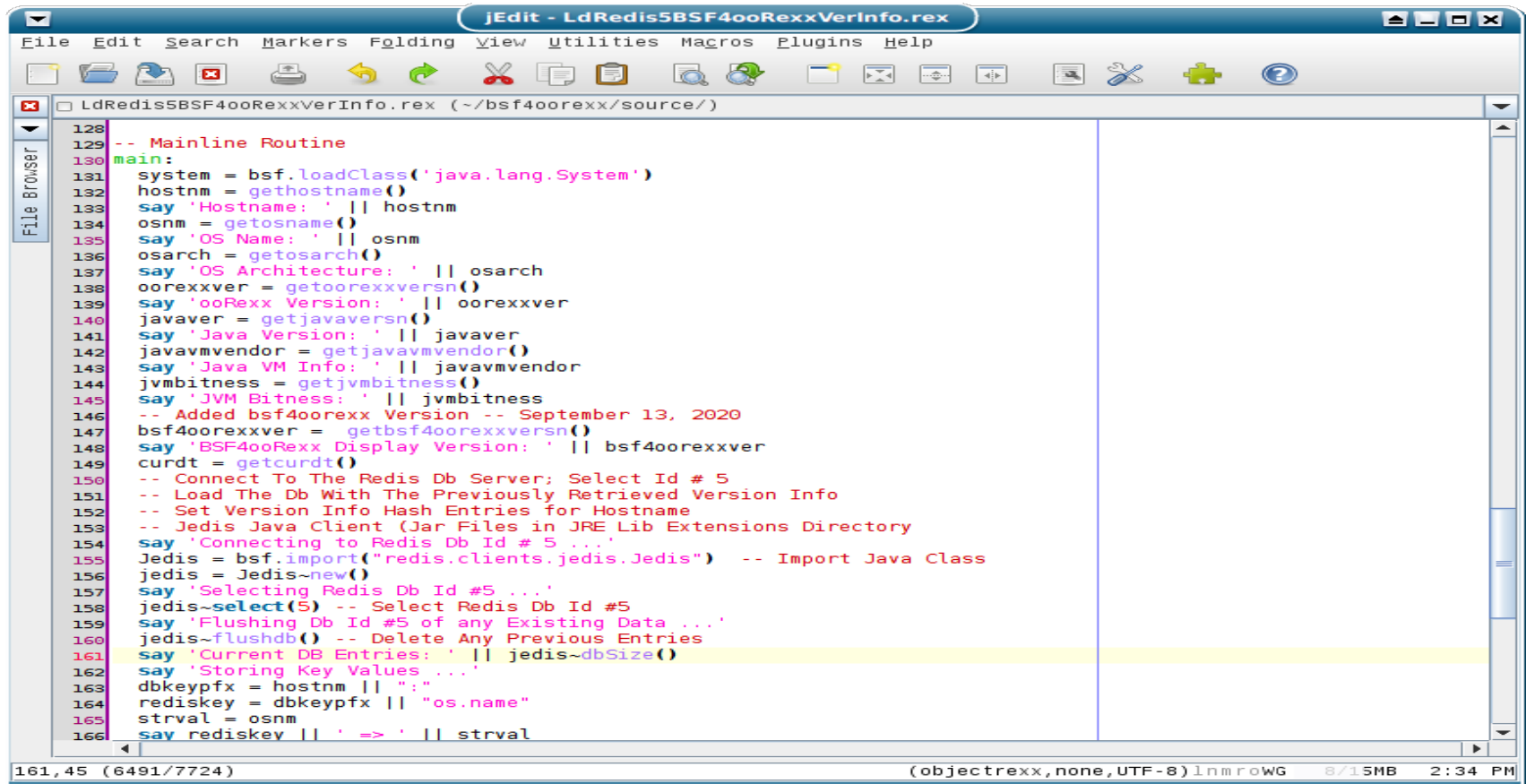
```
Terminal - tonyd@tizthelaw: ~/bsf4oorexx/source
File Edit View Terminal Tabs Help
Last Revised: September 13, 2020
Hostname: tizthelaw
OS Name: Linux
OS Architecture: i386
ooRexx Version: REXX-ooRexx_4.2.0(MT)_32-bit 6.04 29 Dec 2013
Java Version: 1.8.0_261
Java VM Info: Oracle Corporation
JVM Bitness: 32
BSF4ooRexx Display Version: ooRexx 4.2.0 (29 Dec 2013) / BSF 641.20200130 / Java
1.8.0_261, 32-bit
Current Date: 2020-09-14
Connecting to Redis Db Id # 5 ...
Selecting Redis Db Id #5 ...
Flushing Db Id #5 of any Existing Data ...
Current DB Entries: 0
Storing Key Values ...
tizthelaw:os.name => Linux
tizthelaw:os.arch => i386
tizthelaw:ooorexx.verasn => REXX-ooRexx_4.2.0(MT)_32-bit 6.04 29 Dec 2013
tizthelaw:java.version => 1.8.0_261
tizthelaw:java.vm.vendor => Oracle Corporation
tizthelaw:sun.arch.data.model => 32
tizthelaw:.bsf4rexx~display.version => ooRexx 4.2.0 (29 Dec 2013) / BSF 641.2020
0130 / Java 1.8.0_261, 32-bit
tizthelaw:last.update => 2020-09-14
Current DB Entries: 8
Closing Redis Db Connection ...

>>> End Of Program -- LdRedis5BSF4ooRexxVerInfo.rex <<<

tonyd@tizthelaw:~/bsf4oorexx/source$
```

BSF4ooRexx Code Examples using the Jedis Client - IV

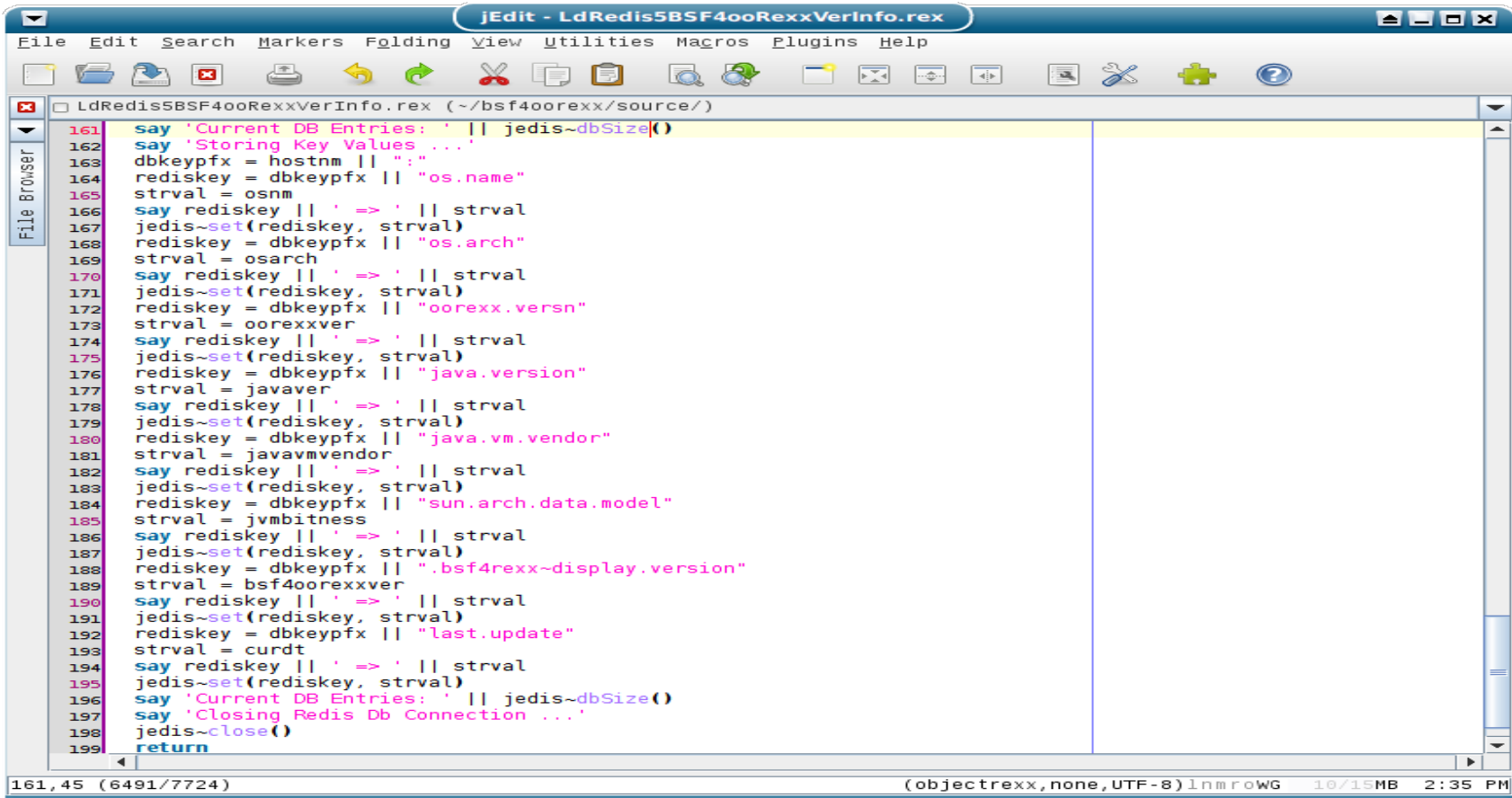
- BSF4ooRexx Code Snippet - Part 1



```
jEdit - LdRedis5BSF4ooRexxVerInfo.rex
File Edit Search Markers Folding View Utilities Macros Plugins Help
LdRedis5BSF4ooRexxVerInfo.rex (~/.bsf4ooorex/source/)
128 -- Mainline Routine
129
130 main:
131 system = bsf.loadClass('java.lang.System')
132 hostnm = gethostname()
133 say 'Hostname: ' || hostnm
134 osnm = getosname()
135 say 'OS Name: ' || osnm
136 osarch = getosarch()
137 say 'OS Architecture: ' || osarch
138 oorexxver = getoorexxversn()
139 say 'ooRexx Version: ' || oorexxver
140 javaver = getjavaversn()
141 say 'Java Version: ' || javaver
142 javavmvendor = getjavavmvendor()
143 say 'Java VM Info: ' || javavmvendor
144 jvmbitness = getjvmbitness()
145 say 'JVM Bitness: ' || jvmbitness
146 -- Added bsf4ooorex Version -- September 13, 2020
147 bsf4ooorexver = getbsf4ooorexversn()
148 say 'BSF4ooRexx Display Version: ' || bsf4ooorexver
149 curdt = getcurdt()
150 -- Connect To The Redis Db Server; Select Id # 5
151 -- Load The Db With The Previously Retrieved Version Info
152 -- Set Version Info Hash Entries for Hostname
153 -- Jedis Java Client (Jar Files in JRE Lib Extensions Directory
154 say 'Connecting to Redis Db Id # 5 ...'
155 Jedis = bsf.import("redis.clients.jedis.Jedis") -- Import Java Class
156 jedis = Jedis~new()
157 say 'Selecting Redis Db Id #5 ...'
158 jedis~select(5) -- Select Redis Db Id #5
159 say 'Flushing Db Id #5 of any Existing Data ...'
160 jedis~flushdb() -- Delete Any Previous Entries
161 say 'Current DB Entries: ' || jedis~dbSize()
162 say 'Storing Key Values ...'
163 dbkeypfx = hostnm || ":"
164 rediskey = dbkeypfx || "os.name"
165 strval = osnm
166 say rediskey || ' => ' || strval
161,45 (6491/7724) (objectrexx,none,UTF-8)lnmrowG 8/15MB 2:34 PM
```


BSF4ooRexx Code Examples using the Jedis Client - V

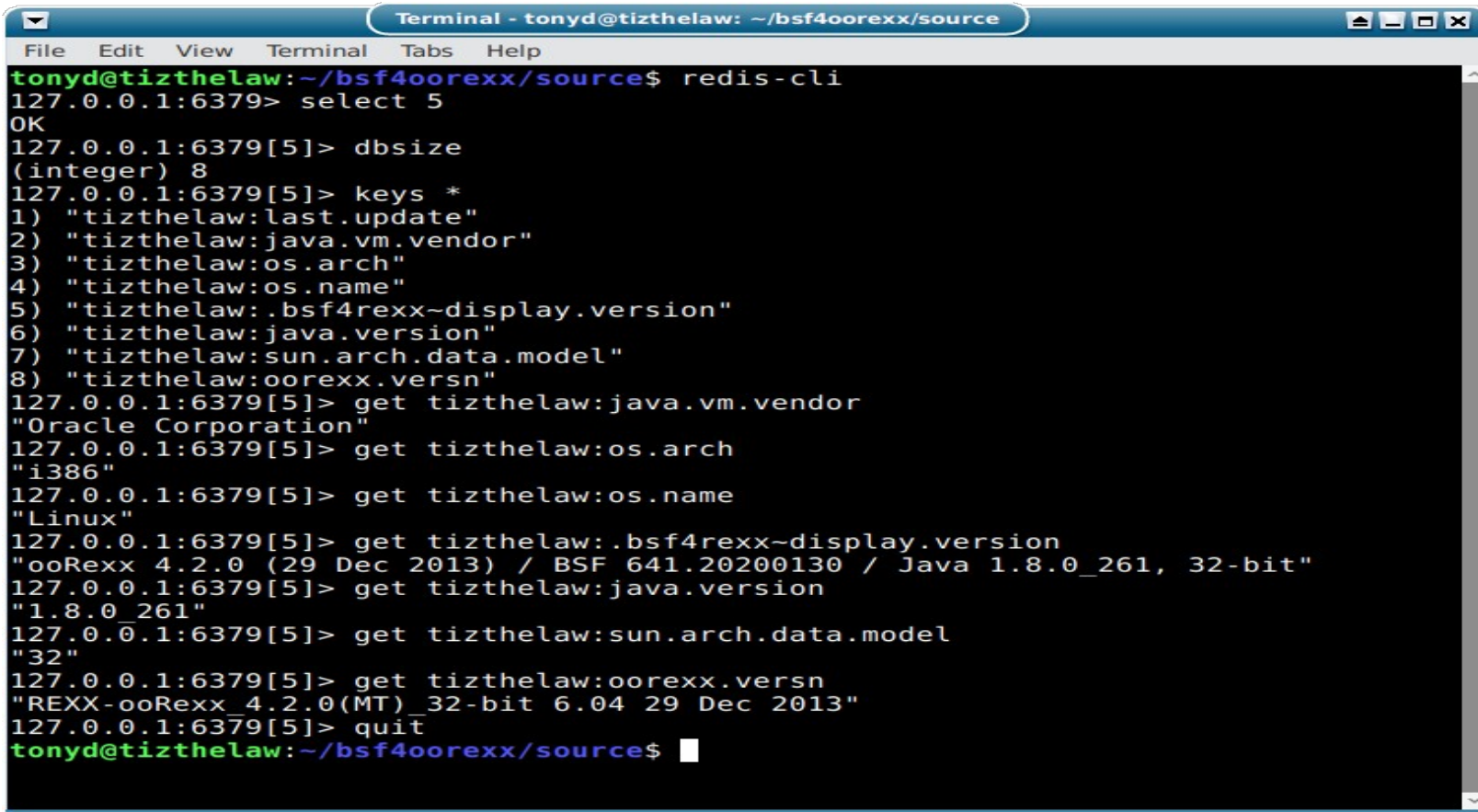
- BSF4ooRexx Code Snippet - Part 2



```
jEdit - LdRedis5BSF4ooRexxVerInfo.rex
File Edit Search Markers Folding View Utilities Macros Plugins Help
LdRedis5BSF4ooRexxVerInfo.rex (~/bsf4oorex/source/)
161 say 'Current DB Entries: ' || jedis~dbSize()
162 say 'Storing Key Values ...'
163 dbkeypfx = hostnm || ":"
164 rediskey = dbkeypfx || "os.name"
165 strval = osnm
166 say rediskey || ' => ' || strval
167 jedis~set(rediskey, strval)
168 rediskey = dbkeypfx || "os.arch"
169 strval = osarch
170 say rediskey || ' => ' || strval
171 jedis~set(rediskey, strval)
172 rediskey = dbkeypfx || "oorexx.versn"
173 strval = oorexxver
174 say rediskey || ' => ' || strval
175 jedis~set(rediskey, strval)
176 rediskey = dbkeypfx || "java.version"
177 strval = javaver
178 say rediskey || ' => ' || strval
179 jedis~set(rediskey, strval)
180 rediskey = dbkeypfx || "java.vm.vendor"
181 strval = javavmvendor
182 say rediskey || ' => ' || strval
183 jedis~set(rediskey, strval)
184 rediskey = dbkeypfx || "sun.arch.data.model"
185 strval = jvmbitness
186 say rediskey || ' => ' || strval
187 jedis~set(rediskey, strval)
188 rediskey = dbkeypfx || ".bsf4rexx-display.version"
189 strval = bsf4oorexver
190 say rediskey || ' => ' || strval
191 jedis~set(rediskey, strval)
192 rediskey = dbkeypfx || "last.update"
193 strval = curdt
194 say rediskey || ' => ' || strval
195 jedis~set(rediskey, strval)
196 say 'Current DB Entries: ' || jedis~dbSize()
197 say 'Closing Redis Db Connection ...'
198 jedis~close()
199 return
161, 45 (6491/7724) (objectrexx, none, UTF-8)lnmrowG 10/15MB 2:35 PM
```

BSF4ooRexx Code Examples using the Jedis Client - VI

- Validating the Data with the Redis Client



```
Terminal - tonyd@tizthelaw: ~/bsf4ooorexx/source
File Edit View Terminal Tabs Help
tonyd@tizthelaw:~/bsf4ooorexx/source$ redis-cli
127.0.0.1:6379> select 5
OK
127.0.0.1:6379[5]> dbsize
(integer) 8
127.0.0.1:6379[5]> keys *
1) "tizthelaw:last.update"
2) "tizthelaw:java.vm.vendor"
3) "tizthelaw:os.arch"
4) "tizthelaw:os.name"
5) "tizthelaw:.bsf4orexx~display.version"
6) "tizthelaw:java.version"
7) "tizthelaw:sun.arch.data.model"
8) "tizthelaw:ooorexx.vernsn"
127.0.0.1:6379[5]> get tizthelaw:java.vm.vendor
"Oracle Corporation"
127.0.0.1:6379[5]> get tizthelaw:os.arch
"i386"
127.0.0.1:6379[5]> get tizthelaw:os.name
"Linux"
127.0.0.1:6379[5]> get tizthelaw:.bsf4orexx~display.version
"ooRexx 4.2.0 (29 Dec 2013) / BSF 641.20200130 / Java 1.8.0_261, 32-bit"
127.0.0.1:6379[5]> get tizthelaw:java.version
"1.8.0_261"
127.0.0.1:6379[5]> get tizthelaw:sun.arch.data.model
"32"
127.0.0.1:6379[5]> get tizthelaw:ooorexx.vernsn
"REXX-ooRexx_4.2.0(MT)_32-bit 6.04 29 Dec 2013"
127.0.0.1:6379[5]> quit
tonyd@tizthelaw:~/bsf4ooorexx/source$
```

BSF4ooRexx Code Examples using the Jedis Client - VII

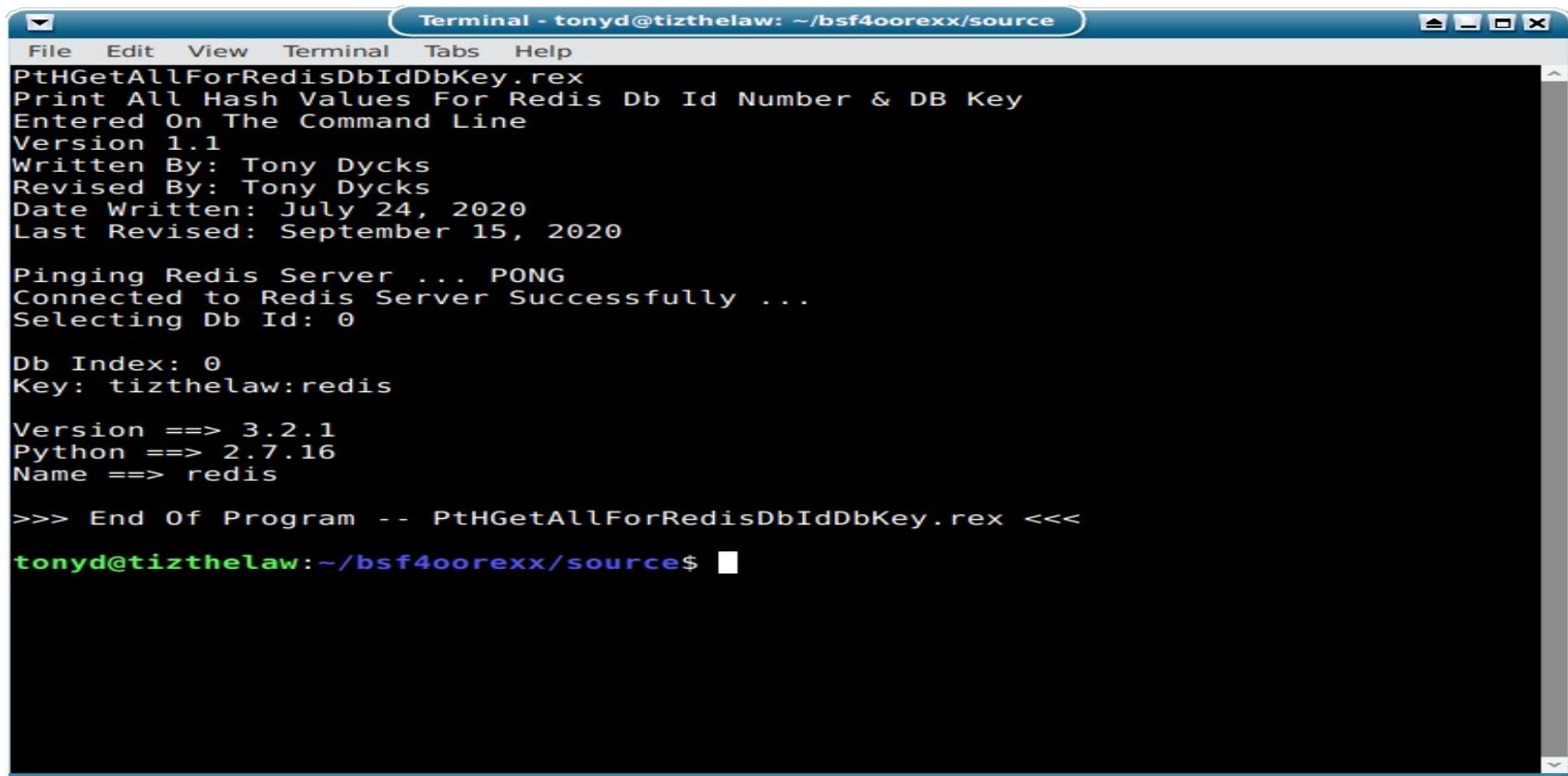
- **Program:** PthGetAllForRedisDbIdDbKey.rex
- Print Key & All Hash Structure Key / Data Values for Redis Db Id & Db Search Key Entered As Command Line Arguments
- **Input:** Redis Db of Previously Loaded Python PIP Library Modules Read from Python PIP Library
- **Output:** Console Display of Keys & Hash Values
- **Validation:** Use Redis CLI Client to Check Selected Values
- **Key Convention:** <hostname>:<pip-package-name>
- **Jedis Methods:** hgetall(dbkey) & hget(dbkey, hashkey)

BSF4ooRexx Code Examples using the Jedis Client - VIII

- **Command Shell Run Syntax:**
- `sh rexxj2.sh PtHGetAllForRedisDbIdDbKey.rex <DbId> <Dbkey>`
- **Examples:**
- `sh rexxj2.sh PtHGetAllForRedisDbIdDbKey.rex 0 tizthelaw:pyodbc`
- `sh rexxj2.sh PtHGetAllForRedisDbIdDbKey.rex 0 tizthelaw:pyqdbm`
- `sh rexxj2.sh PtHGetAllForRedisDbIdDbKey.rex 0 tizthelaw:redis`

BSF4ooRexx Code Examples using the Jedis Client - IX

- Command Shell Run Output:



```
Terminal - tonyd@tizthelaw: ~/bsf4ooorex/source
File Edit View Terminal Tabs Help
PtHGetAllForRedisDbIdDbKey.rex
Print All Hash Values For Redis Db Id Number & DB Key
Entered On The Command Line
Version 1.1
Written By: Tony Dycks
Revised By: Tony Dycks
Date Written: July 24, 2020
Last Revised: September 15, 2020

Pinging Redis Server ... PONG
Connected to Redis Server Successfully ...
Selecting Db Id: 0

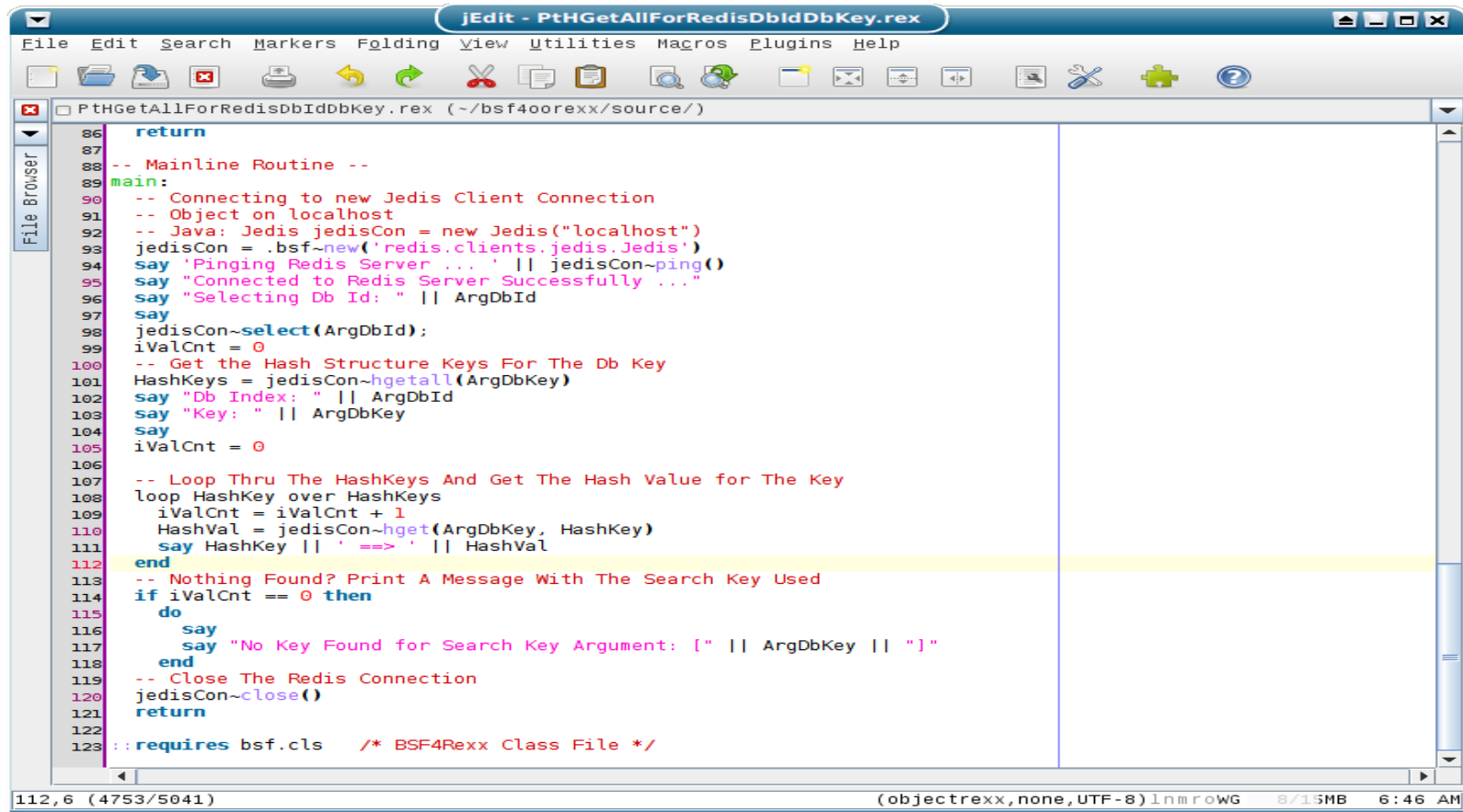
Db Index: 0
Key: tizthelaw:redis

Version ==> 3.2.1
Python ==> 2.7.16
Name ==> redis

>>> End Of Program -- PtHGetAllForRedisDbIdDbKey.rex <<<
tonyd@tizthelaw:~/bsf4ooorex/source$
```

BSF4ooRexx Code Examples using the Jedis Client - X

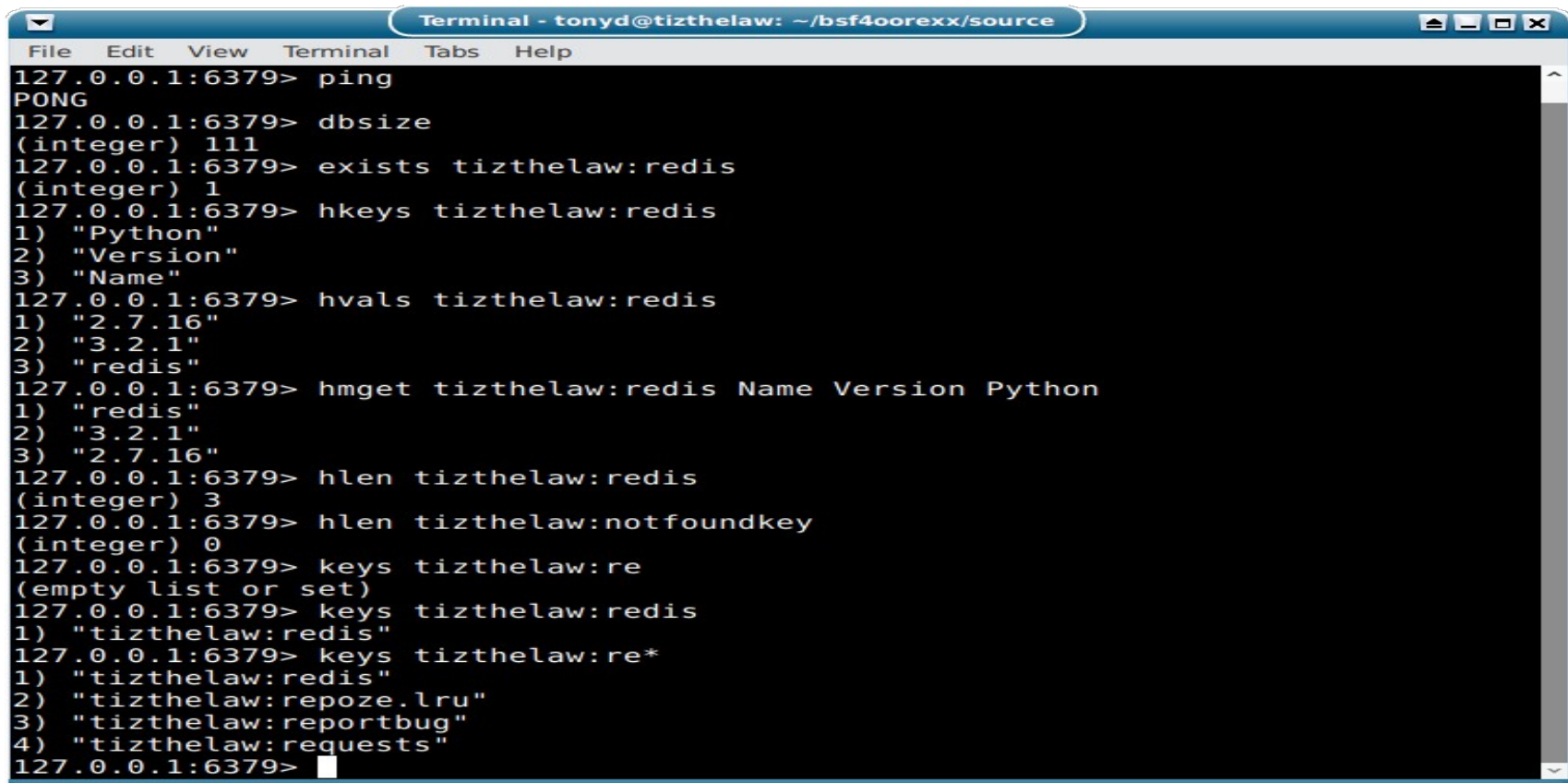
- BSF4ooRexx Code Snippet



```
86 return
87
88 -- Mainline Routine --
89 main:
90 -- Connecting to new Jedis Client Connection
91 -- Object on localhost
92 -- Java: Jedis jedisCon = new Jedis("localhost")
93 jedisCon = .bsf~new('redis.clients.jedis.Jedis')
94 say 'Pinging Redis Server ... ' || jedisCon~ping()
95 say "Connected to Redis Server Successfully ..."
96 say "Selecting Db Id: " || ArgDbId
97 say
98 jedisCon~select(ArgDbId);
99 iValCnt = 0
100 -- Get the Hash Structure Keys For The Db Key
101 HashKeys = jedisCon~hgetall(ArgDbKey)
102 say "Db Index: " || ArgDbId
103 say "Key: " || ArgDbKey
104 say
105 iValCnt = 0
106
107 -- Loop Thru The HashKeys And Get The Hash Value for The Key
108 loop HashKey over HashKeys
109   iValCnt = iValCnt + 1
110   HashVal = jedisCon~hget(ArgDbKey, HashKey)
111   say HashKey || ' ==> ' || HashVal
112 end
113 -- Nothing Found? Print A Message With The Search Key Used
114 if iValCnt == 0 then
115   do
116     say
117     say "No Key Found for Search Key Argument: [" || ArgDbKey || "]"
118   end
119 -- Close The Redis Connection
120 jedisCon~close()
121 return
122
123 ::requires bsf.cls /* BSF4Rexx Class File */
```

BSF4ooRexx Code Examples using the Jedis Client - XI

- Validating the Program Run via redis-cli



```
Terminal - tonyd@tizthelaw: ~/bsf4ooorex/source
File Edit View Terminal Tabs Help
127.0.0.1:6379> ping
PONG
127.0.0.1:6379> dbsize
(integer) 111
127.0.0.1:6379> exists tizthelaw:redis
(integer) 1
127.0.0.1:6379> hkeys tizthelaw:redis
1) "Python"
2) "Version"
3) "Name"
127.0.0.1:6379> hvals tizthelaw:redis
1) "2.7.16"
2) "3.2.1"
3) "redis"
127.0.0.1:6379> hmget tizthelaw:redis Name Version Python
1) "redis"
2) "3.2.1"
3) "2.7.16"
127.0.0.1:6379> hlen tizthelaw:redis
(integer) 3
127.0.0.1:6379> hlen tizthelaw:notfoundkey
(integer) 0
127.0.0.1:6379> keys tizthelaw:re
(empty list or set)
127.0.0.1:6379> keys tizthelaw:redis
1) "tizthelaw:redis"
127.0.0.1:6379> keys tizthelaw:re*
1) "tizthelaw:redis"
2) "tizthelaw:repoze.lru"
3) "tizthelaw:reportbug"
4) "tizthelaw:requests"
127.0.0.1:6379>
```


Findings & Recommendations

- Redis Db is an Easy to Install on Linux & Up to Date for most Distros for a Development environment
- Windows Binary is also easy to install but a number of versions out of date (v3.2 vs. 6.0 Stable)
- Performance is quite fast; Scaling a bit of a challenge vs. MongoDB and Cassandra
- Offers a wider range of data structures over File Based DBMs such as Gnu DBM & QDBM
- Very Good Command Reference on Redis Website
- Jedis Java Client is Quite Easy to Implement for both NetRexx and BSF4ooRexx

End of Presentation redis

- Thanks for Your Time and Attention
- Questions?
- Zip File of NetRexx and BSF4ooRexx Program Samples will be included on the Symposium 2020 Web Page
- Will try to address any unanswered Questions from this session on the Rexx LA forum, if Okay in terms of Forum Etiquette?