WORKOUT Command List

This command list is a work in progress. A WORKOUT file has one command per line, and the format of each line is `command arg1 arg2 … argn`, designed to be simple to both parse and for other programs to generate. This file tries to document the commands, which have always been defined by "whatever the code does." These commands were originally designed for a scripting environment, and will have to be changed to fit in with the web.

Commands:

```
# comment line
```

Lines which begin with a # are comments.

```
QUIT
```

```
EXIT
```

Either of these stops the current script, which is useful to test part of a script and exit without running the rest of the script.

This is a problematic command in Java, because we can't simply call `System.exit()`. After all, we could be in a J2EE container and crash it. For now, I haven't implemented this in the Java version.

```
ECHO msg
```

Prints the message, which extends to the end of the line. (This command does not translate well to the web where there is no terminal and may not be supported.)

```
HELP
```

Supposed to display a help message with command syntax and usage, but I don't think I ever actually implemented it, and besides it does not translate well into a web environment.

```
STATS
```

Displays the number of actions taken so far (inserts, updates, deletes, fetches, time spent waiting). This will not translate well to the web and will probably need to be augmented.

```
I1 k v1 v2 v3
```

```
I2 k v1 v2 v3
```

```
I3 k v1 v2 v3
```

These three commands insert rows into the WORKOUT tables. Usually, the `PUSH` command will be used instead by end users, although these commands are available for special cases.

```
D1 k
```

```
D2 k
```

```
D3 k
```

Deletes keys. In practice, the `NUKE` command is much faster and easier. Rarely do I ever want to delete specific keys. I normally want to blow away an entire test case and start over.

```
U1 k v1 v2 v3
```

```
U2 k v1 v2 v3
```

```
U3 k v1 v2 v3
```

Updates rows. Keys do not change, but the values do. Note that this could upset referential integrity. Only values on the leaf tables ought to change. The ability to change the main table is available, because someone may want to update it with the insert command and then update it with the update command, and never use the full RI set of tables.

```
PUSH lo hi
```

The main command to push a lot of data into the tables, by using `lo` and `hi` as a range of keys, and sequentially pushing all three rows of the RI set with correct keys. This command does all the work of keeping track of the RI set, so it's the easiest to use.

```
SLOW
```

Causes a pause between each operation. This does not translate well to the web environment, but is useful in scripts.

```
NUKE
```

Deletes all the rows from all three tables, and in the correct order so the RI is not violated.

```
DUMP
```

Displays rows in all three tables. Will not translate well to the web.