

Output

For Client:

```
conn
connected to connectfd:4 host:127.0.0.1 port:9001
connected to connectfd:5 host:127.0.0.1 port:9002
put 100000 string100000
OK数字和p-(P-)记数法(C99)
put 100001 string100001
OK
put 100002 string100002
OKdouble)
put 100003 string100003
OK(E-)记数法)
put 100004 string100004
OK
put 100005 string100005
OK与%ld相同)
put 100001 string100001
Already Added
put 100001 string100011
Replaced Value
get 100001
string100011
get 100000
string100000
del 100005
Deleted
del 100005
Not Found
get 100005
Not Found
```

After connected to the servers using “conn” command, we get the connection information.

“put 100000 string100000” will send the put instruction to server, server send “OK” back which means it store the key-value pair 100000-string100000 successfully.

Then we repeated typing “put 100001 string100001” “put 100002 string100002” “put 100003 string100003” “put 100004 string100004” “put 100005 string100005”, asked server to store other 5 key-value pairs.

When we type “put 100001 string100001” again to ask server to store the same key-value, unfortunately, server send message “Already Added” back.

When we type “put 100001 string100011” to ask server to store the same key, server will find the key

and update the value to the latest string100011, then send message “Replaced Value” back.

When we type “get 100001” to ask server return the value of key 100001, as a response, server send the latest value “string100011” as the message.

When we type “get 100000” to ask server return the value of key 100000, as a response, server send the previous value “string100000” as the message.

When we type “del 100005” to request server delete key-value pair in key 100005, after server deleted successfully, return the message “Deleted” as the message.

When we type “del 100005” to request server delete again, after server failed searching, return the message “Not found” as the message.

When we type “get 100005” to request server query the value of key 100005, after server failed searching because we deleted, return the message “Not found” as the message.

For server, I enabled HASHDEBUG at the head of “csHash.h” to allow server show the debug and response information as below.

```
connected from host:127.0.0.1 port:43363
adding 100001 -> string100001 hash: 4646
entry: (nil)
creating new entry
new entry: 0x7f65640008c0
added entry
adding 100003 -> string100003 hash: 1022
entry: (nil)
creating new entry
new entry: 0x7f6564000910
added entry
adding 100005 -> string100005 hash: 2166
entry: (nil)
creating new entry
new entry: 0x7f6564000960
added entry
adding 100001 -> string100001 hash: 4646
entry: 0x7f65640008c0
checking entry: 0x7f65640008c0
string100001
string100001
adding 100001 -> string100011 hash: 4646
entry: 0x7f65640008c0
checking entry: 0x7f65640008c0
string100011
string100001
fetching 100001 -> ?? hash: 4646
found entry key: 100001 value: string100011
deleting: 100005 hash: 2166
entry: 0x7f6564000960
checking entry: 0x7f6564000960
deleting: 100005 hash: 2166
entry: (nil)
fetching 100005 -> ?? hash: 2166
```