|  |  |
| --- | --- |
| TEA PRIMITIVE | SEMANTICS |
| S: | |  |  | | --- | --- | | NAME | Salt | | PURPOSE | Randomly Salting or Un-salting strings | | SYNTAX  & SEMANTICS | |  | | --- | | s: | | Inject a SINGLE SPACE CHARACTER into AI at a random position. Salting the AI. | | s:STR:N:N2  s:STR:LIMIT:LLIMIT | | Return AI with the string STR injected into it at some random  Position. With N specified, injects at an index position (from 0) not greater than N within STR, injecting anywhere if N > |AI|. To precisely control where to operate, use N2, as a number to control the lower limit for the index search, so that, when N = N2, it basically means to precisely operate at index N, otherwise one can control the operating space more liberally. | | s!: | | Delete a SINGLE CHARACTER from AI at a random position. Unsalting AI. | | s!:REGEX:N:N2 | | Randomly select one of the sections in AI matching REGEX, and delete it. With N specified, only deletes an occurrence not later than the Nth-occurrence of REGEX within AI, with N2, operates on the range between N2 and N, operating on exactly the Nth occurrence if N=N2 | | s\*: vNAME:STR:N:N2 | | Same as s:STR:N:N2, but operating on the string stored in the vault with the name vNAME instead of AI. | | s\*!: vNAME:REGEX:N:N2 | | Same as s!:REGEX:N:N2, but operating on the string stored in the vault with the name vNAME instead of AI. | |  | | NOTES | First, it should be noted that salting strings is not the same as substitution. In salting, the original string contents remain, but new content is injected at random (default) or defined positions in the string. Unsalting on the other hand, leaves the original string with some sections of it randomly deleted.  Some examples follow:  i!:TEST|s:\_\_ # shall sometimes return “TE\_\_ST”, “\_\_TEST”, “T\_\_EST”, etc.  i!:TEST|s!: # might sometimes return "TET", "EST", "TES", etc. Basically, unsalting  The following example, called the “SIR Game”--- perhaps, because of the TEA primitives it employs, uses salting to create a simple child’s game where they are tasked with filling in the gap for a letter missing in a given 5-letter word, which is then marked by a “?”.  # randomly pick a word from the given list  I!:{HONEY TRICK WINDS GAMES} | a: | d:[ ].\*$  # (=”TRICK”) for example  S:-:4 | # randomly inject “-“ b4 4th pstn  #(=”TR-ICK”) for example  R:.[-]:? | R:[-].:? | q!: | # returns “TR?CK” |  |  | |