kutils\_180911

Abstract

This packet includes utility functions of common destination. But it is need have in mind, that this packet changes behavior of the system packet “rand” by that that it have init function, which sets new value for the seed (see <https://golang.org/pkg/math/rand/#Seed> and [True random funtions](#true_rand_funcs)) For comparison with early version see [list\_of\_changes](#list_of_changes)

API

[NULL functions](#null_funcs)

[True Random functions](#true_rand_funcs)

[StrWithoutNull](#StrWithoutNull)

[ExtractCommandPar (extract number)](#ExtractCommandPar)

NULL functions

NULL functions are functions with prefix “Null”. Each of function with prefix "Null" presumes that its first parameter carries a corresponding to its name value and returns its string representation. E.g. the function “NullTimeToString” presumes that the first parameter is of type “time”. If a first parameter is nil the second one (i.e. "ifNil") is returned. If the first is not nil and the presumption is not true, then the third (i.e. "ifWhatIs") is returned. There are the next functions:

func NullTimeToString(nt interface{}, ifNil string, ifWhatIs string) string

func NullIntToString(nt interface{}, ifNil string, ifWhatIs string) string

The next function presumes float64

func NullFloatToString(nt interface{}, ifNil string, ifWhatIs string) string

True Random functions

These functions return a random value from the seed of duration from 07.11.1917 to now (see also [common notes](#abstract)). In strict formal sense these functions can return not nil error if the function init fails. In that case these anyway return a random value but from the default seed. There are the functions

func TrueRandInt() (string, error)

func TrueRandIntAsInt() (int, error)

Beside those mentioning upward there is also

func StrWithoutNull(str string, rep string) (res string). This function returns string that is produced from parameter “str” by replacing all bytes that are less than 9 with parameter “rep”. If rep==”” or its length>1 then it sets rep=”!”

//It extracts bytes from beginning up to encounting a byte outside interval of [48 - 57] or of the end of slice.

//That is, it reads digits 0 ... 9. If the such there are not it retuns "-1".

//Otherwise it renders the bytes having been read as a record of decimal number and returns a string presentation of this

//E.g., the {49,48,48,13,0,0,0} it renders as "100", the {48,48,13,0,0,0} it renders as "0"

func ExtractCommandPar(rawPar []byte) string

The name of this function is not fitted to its functionality. More truly it might be naming as ExtractNumber, but historically there is the case and let it be so.

Besides this there is also

func ExtractCommandParInt(rawPar []byte) int

It internally calls the previous function and applies to received result the function strconv.Atoi without handling an err if any.

list of changes

180911

1 This list was added

2 added [ExtractCommandPar](#ExtractCommandPar) (kot\_common\kutils\utils.go)