
ipaddressstools v1.2.12

Release 1.2.12

May 04, 2021

Contents:

1	ipaddresstools	1
1.1	ipaddresstools package	1
2	Indices and tables	7
	Python Module Index	9
	Index	11

1.1 ipaddressstools package

1.1.1 Submodules

1.1.2 ipaddressstools.ipaddressstools module

Script Name: ipaddressstools.py Script Type: Python Updated By: Benjamin P. Trachtenberg Date Written 1/11/2015

Description: Collection of tools for IP Address's

`ipaddressstools.ipaddressstools.all_ip_address_in_subnet(ip_net, cidr)`

Function to return every ip in a subnet

Parameters

- **ip_net** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 0 to 32

Return type List

Returns A List of IPv4 Address's

`ipaddressstools.ipaddressstools.all_subnets_longer_prefix(ip_net, cidr)`

Function to return every subnet a ip can belong to with a longer prefix

Parameters

- **ip_net** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 0 to 32

Return type List

Returns A List of subnets

`ipaddressstools.ipaddressstools.all_subnets_possible(ip_net, cidr)`

Function to return every subnet a ip can belong to with a longer prefix

Parameters

- **ip_net** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 0 to 32

Return type List

Returns A List of subnets

`ipaddrstools.ipaddrstools.all_subnets_shorter_prefix(ip_net, cidr, include_default=False)`

Function to return every subnet a ip can belong to with a shorter prefix

Parameters

- **ip_net** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 0 to 32
- **include_default** (*Boolean*) – True or False

Return type List

Returns A List of subnets

`ipaddrstools.ipaddrstools.cidr_check(cidr, return_cidr=True)`

Function to verify a good CIDR value

Parameters

- **cidr** (*Integer*) – 30
- **return_cidr** (*Boolean*) – True or False

Return type String

Returns String

`ipaddrstools.ipaddrstools.get_neighbor_ip(ip_addr, cidr='30')`

Function to figure out the IP's between neighbors address

Parameters

- **ip_addr** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 30, or 31

Return type Tuple

Returns returns Our IP and the Neighbor IP in a tuple

`ipaddrstools.ipaddrstools.ip(ip_addr, return_tuple=True)`

Function to check if a address is good

Parameters

- **ip_addr** (*String*) – String format 192.168.1.1
- **return_tuple** (*Boolean*) – True or False

Return type Tuple, or Boolean

Returns Tuple, or Boolean

`ipaddrstools.ipaddrstools.ip_mask(ip_addr_and_mask, return_tuple=True)`

Function to check if a address and CIDR mask is good

Parameters

- **ip_addr_and_mask** (*String*) – String format 192.168.1.1/24
- **return_tuple** (*Boolean*) – True or False

Return type Tuple, or Boolean

Returns Tuple, or Boolean

`ipaddressstools.ipaddressstools.mcast_ip(ip_addr, return_tuple=True)`

Function to check if a address is multicast

Parameters

- **ip_addr** (*String*) – String format 239.1.1.1
- **return_tuple** (*Boolean*) – True or False

Return type Tuple, or Boolean

Returns Tuple, or Boolean

`ipaddressstools.ipaddressstools.mcast_ip_mask(ip_addr_and_mask, return_tuple=True)`

Function to check if a address is multicast and that the CIDR mask is good

Parameters

- **ip_addr_and_mask** (*String*) – String format 239.0.0.0/24
- **return_tuple** (*Boolean*) – True or False

Return type Tuple, or Boolean

Returns Tuple, or Boolean

`ipaddressstools.ipaddressstools.number_check(check, return_number=True)`

Function to verify item entered is a number

Parameters

- **check** (*Anything*) – What to check
- **return_number** (*Boolean*) – True or False

Return type Boolean

Returns Check return_number for return options

`ipaddressstools.ipaddressstools.random_cidr_mask(lowest_mask=16)`

Function to generate a random CIDR value

Parameters **lowest_mask** (*Integer*) – An integer value for the lowest mask you want it to generate

Return type String

Returns A string of a random CIDR mask

`ipaddressstools.ipaddressstools.random_mcast_ip()`

Function to generate a random multicast ip address

Return type String

Returns A multicast IP Address

`ipaddressstools.ipaddressstools.random_mcast_ip_mask(lowest_mask=16)`

Function to generate a random multicast ip address and cidr mask pair in the following format X.X.X.X/X

Parameters **lowest_mask** (*Integer*) – An integer value for the lowest mask you want it to generate

Return type String

Returns A multicast IP Address and CIDR pair in the following format X.X.X.X/X

`ipaddressstools.ipaddressstools.random_ucast_ip()`

Function to generate a random unicast ip address

Return type String

Returns A unicast IP Address

`ipaddressstools.ipaddressstools.random_ucast_ip_mask(lowest_mask=16)`

Function to generate a random unicast ip address and cidr mask pair in the following format X.X.X.X/X

Parameters **lowest_mask** (*Integer*) – An integer value for the lowest mask you want it to generate

Return type String

Returns A unicast IP Address and CIDR pair in the following format X.X.X.X/X

`ipaddressstools.ipaddressstools.subnet_range(ip_net, cidr)`

Function to return a subnet range value from a IP address and CIDR pair

Parameters

- **ip_net** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 1 to 32

Return type Dict

Returns returns a dictionary of info

`ipaddressstools.ipaddressstools.unicast_ip(ip_addr, return_tuple=True)`

Function to check if a address is unicast

Parameters

- **ip_addr** (*String*) – String format 192.168.1.1
- **return_tuple** (*Boolean*) – True or False

Return type Tuple, or Boolean

Returns Tuple, or Boolean

`ipaddressstools.ipaddressstools.unicast_ip_mask(ip_addr_and_mask, return_tuple=True)`

Function to check if a address is unicast and that the CIDR mask is good

Parameters

- **ip_addr_and_mask** (*String*) – String format 192.168.1.1/24
- **return_tuple** (*Boolean*) – True or False

Return type Tuple, or Boolean

Returns Tuple, or Boolean

`ipaddressstools.ipaddressstools.whole_subnet_maker(ip_addr, cidr)`

Function to return a whole subnet value from a IP address and CIDR pair

Parameters

- **ip_addr** (*String*) – String format 192.168.1.1
- **cidr** (*String*) – CIDR value of 0 to 32

Return type String

Returns returns the corrected whole subnet

1.1.3 Module contents

Init for ipaddressstools

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

i

`ipaddresstools`, 5

`ipaddresstools.ipaddresstools`, 1

A

`all_ip_address_in_subnet()` (in module *ipaddressstools.ipaddressstools*), 1
`all_subnets_longer_prefix()` (in module *ipaddressstools.ipaddressstools*), 1
`all_subnets_possible()` (in module *ipaddressstools.ipaddressstools*), 1
`all_subnets_shorter_prefix()` (in module *ipaddressstools.ipaddressstools*), 2

C

`cidr_check()` (in module *ipaddressstools.ipaddressstools*), 2

G

`get_neighbor_ip()` (in module *ipaddressstools.ipaddressstools*), 2

I

`ip()` (in module *ipaddressstools.ipaddressstools*), 2
`ip_mask()` (in module *ipaddressstools.ipaddressstools*), 2
ipaddressstools (module), 5
ipaddressstools.ipaddressstools (module), 1

M

`mcast_ip()` (in module *ipaddressstools.ipaddressstools*), 3
`mcast_ip_mask()` (in module *ipaddressstools.ipaddressstools*), 3

N

`number_check()` (in module *ipaddressstools.ipaddressstools*), 3

R

`random_cidr_mask()` (in module *ipaddressstools.ipaddressstools*), 3

`random_mcast_ip()` (in module *ipaddressstools.ipaddressstools*), 3
`random_mcast_ip_mask()` (in module *ipaddressstools.ipaddressstools*), 3
`random_ucast_ip()` (in module *ipaddressstools.ipaddressstools*), 4
`random_ucast_ip_mask()` (in module *ipaddressstools.ipaddressstools*), 4

S

`subnet_range()` (in module *ipaddressstools.ipaddressstools*), 4

U

`ucast_ip()` (in module *ipaddressstools.ipaddressstools*), 4
`ucast_ip_mask()` (in module *ipaddressstools.ipaddressstools*), 4

W

`whole_subnet_maker()` (in module *ipaddressstools.ipaddressstools*), 4