## Class Subnetting

* java.lang.Object
  + Subnetting

public class Subnetting

extends java.lang.Object

### Constructor Summary

|  |
| --- |
| Constructors |
| **Constructor and Description** |
| [Subnetting](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#Subnetting-java.lang.String-java.lang.String-int-)(java.lang.String origNA, java.lang.String subnetMask, int numOfSubnets)  This is the constructor of the Subnetting class. |

### Method Summary

|  |  |
| --- | --- |
| All Methods | |
| Modifier and Type | **Method and Description** |
| java.lang.String | [binaryFormNoPeriod](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#binaryFormNoPeriod-int-int-int-int-)(int n1, int n2, int n3, int n4)  This combines four octets into a single String value |
| java.lang.String | [binaryFormWithPeriod](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#binaryFormWithPeriod-int-int-int-int-)(int n1, int n2, int n3, int n4)  This combines four octets and places a dot after each octet |
| int | [binaryToDecimal](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#binaryToDecimal-int-)(int binaryN)  This method converts a binary value to a decimal number |
| int | [bitsToAdd](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#bitsToAdd-int-)(int numOfSubnets)  This method computes how many bits are needed to be borrowed |
| boolean | [checkNetworkAddress](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#checkNetworkAddress-java.lang.String-)(java.lang.String networkAddress)  This method checks whether the network address is valid |
| boolean | [checkSubnetMask](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#checkSubnetMask-java.lang.String-)(java.lang.String subnetMask)  This method checks whether the subnet mask is valid |
| java.lang.String | [decimalToBinary](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#decimalToBinary-int-)(int decimalN)  This method converts a decimal number to a binary value |
| java.lang.String | [getNewHostsPerNet](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#getNewHostsPerNet--)()  This returns the value of the Hosts per subnet |
| java.lang.String | [getNewNetMask](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#getNewNetMask--)()  This returns the value of the new Netmask |
| java.util.ArrayList<java.lang.String> | [getSubnets](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#getSubnets--)()  This method computes the subnet and returns an arraylist of strings containing the Newtwork Address, First Usable Address, Last Usable Address, and the Broadcast Address of each subnet |
| int | [hostsPerNet](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#hostsPerNet-int-)(int numOfSubnets)  This computes how many hosts per net are available per subnet |
| java.lang.String | [IPadd](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#IPadd-java.lang.String-)(java.lang.String n)  This method converts an binary String into a decimal into an IP address format |
| int | [netmask](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#netmask-int-)(int n)  This method computes the netmask |
| int | [numOf1](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#numOf1-java.lang.String-)(java.lang.String n)  This method counts how many times the number 1 appears in a String |
| java.lang.String | [numOf1ToBinary](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#numOf1ToBinary-int-)(int n)  This method converts an integer value into a binary containing several 1's with periods |
| java.lang.String | [numOf1ToDecimal](file:///C:\Users\carl\Desktop\Files\DASTAPP\Subnetting%20Final\doc\Subnetting.html#numOf1ToDecimal-int-)(int n)  This method converts an integer value into a binary containing several 1's and converting it back to a decimal value with dots separting each octet |

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructor Detail

#### Subnetting

* + - public Subnetting(java.lang.String origNA,
    - java.lang.String subnetMask,

int numOfSubnets)

This is the constructor of the Subnetting class. It is required to build the object.

Parameters:

origNA - network address derived from the input of the user

subnetMask - subnet mask from the input of the user

numOfSubnets - number of subnets required by the user

### Method Detail

#### checkNetworkAddress

public boolean checkNetworkAddress(java.lang.String networkAddress)

This method checks whether the network address is valid

Parameters:

networkAddress - String containing the network address

Returns:

returns true if network address is valid, otherwise returns false

#### checkSubnetMask

public boolean checkSubnetMask(java.lang.String subnetMask)

This method checks whether the subnet mask is valid

Parameters:

subnetMask - String containing the subnet mask

Returns:

returns true if subnet mask is valid, otherwise returns false

#### numOf1

public int numOf1(java.lang.String n)

This method counts how many times the number 1 appears in a String

Parameters:

n - String value

Returns:

Integer value containing how many times the number 1 occurred

#### getNewNetMask

public java.lang.String getNewNetMask()

This returns the value of the new Netmask

Returns:

String of the new Netmask

#### getNewHostsPerNet

public java.lang.String getNewHostsPerNet()

This returns the value of the Hosts per subnet

Returns:

String representing the number of hosts per subnet

#### getSubnets

public java.util.ArrayList<java.lang.String> getSubnets()

This method computes the subnet and returns an arraylist of strings containing the Newtwork Address, First Usable Address, Last Usable Address, and the Broadcast Address of each subnet

Returns:

an Arraylist of String containing all the Subnets

#### binaryToDecimal

public int binaryToDecimal(int binaryN)

This method converts a binary value to a decimal number

Parameters:

binaryN - the binary value of number being converted

Returns:

an Integer containing the decimal value

#### decimalToBinary

public java.lang.String decimalToBinary(int decimalN)

This method converts a decimal number to a binary value

Parameters:

decimalN - the decimal number of the binary being converted

Returns:

a String containing the binary value

#### binaryFormWithPeriod

* + - public java.lang.String binaryFormWithPeriod(int n1,
    - int n2,
    - int n3,

int n4)

This combines four octets and places a dot after each octet

Parameters:

n1 - the first octet

n2 - the second octet

n3 - the third octet

n4 - the fourth octet

Returns:

String containing the merged octets with dot as separator

#### binaryFormNoPeriod

* + - public java.lang.String binaryFormNoPeriod(int n1,
    - int n2,
    - int n3,

int n4)

This combines four octets into a single String value

Parameters:

n1 - the first octet

n2 - the second octet

n3 - the third octet

n4 - the fourth octet

Returns:

String containing the merged octets without dots in between each octet

#### netmask

public int netmask(int n)

This method computes the netmask

Parameters:

n - a binary value

Returns:

Integer value containing the new Netmask

#### bitsToAdd

public int bitsToAdd(int numOfSubnets)

This method computes how many bits are needed to be borrowed

Parameters:

numOfSubnets - the number of subnets specified by the user

Returns:

Integer value containing how many bits are borrowed

#### hostsPerNet

public int hostsPerNet(int numOfSubnets)

This computes how many hosts per net are available per subnet

Parameters:

numOfSubnets - the number of subnets specified by the user

Returns:

Integer value containing the number of hosts per subnet

#### numOf1ToDecimal

public java.lang.String numOf1ToDecimal(int n)

This method converts an integer value into a binary containing several 1's and converting it back to a decimal value with dots separting each octet

Parameters:

n - the Integer to be converted

Returns:

String value containing the decimal value

#### numOf1ToBinary

public java.lang.String numOf1ToBinary(int n)

This method converts an integer value into a binary containing several 1's with periods

Parameters:

n - the Integer to be converted

Returns:

String value containing the binary value

#### IPadd

public java.lang.String IPadd(java.lang.String n)

This method converts an binary String into a decimal into an IP address format

Parameters:

n - the binary String to be converted

Returns:

the converted binary String into an IP address format