
<<Java Class>>
test
(default package)
test()
main(String[]):void
<<Java Class>>
filterCSV
(default package)
filterCSV()
FilterId(String,Network):Network
FilterByDate(Date,Date,Network):Network
FilterByRadius(double,double,double,Network):Network
<<Java Class>>
toCSV
(default package)
toCSV(String)
routerAsp(Network,String):void
<<Java Class>>
csvBase
(default package)
csvBase()
readCSV(String):Network
check(File,Network):void
add(String[]):WIFI
addGeo(String[]):GeoModDat
<<Java Class>>
toKML
(default package)
toKML(String)
<<Java Class>>
Sort
(default package)
Sort()
mergeSort(Network):Hotspots[]
mergeSort(Hotspots[],int,int):void
<<Java Class>>
kmlBase
(default package)
kmlBase()
inputCsv(File):Network
<<Java Class>>
Algo
(default package)
power: double
sigDif: double
norm: double
notInList: double
sigDifnotList: double
minDif: double
Algo()
routerPlaceAlgo1(Network):ArrayList<RouterPlace>
clientPlaceAlgo2(String,String):ArrayList<RouterPlace>
math1(ArrayList<RouterPlace>,Network):ArrayList<R...
check(Hotspots,ArrayList<RouterPlace>):void
math(ArrayList<ArrayList<RouterPlace>>):ArrayList<...
solve(ArrayList<RouterPlace>):RouterPlace
add(Network):ArrayList<RouterPlace>
sortMac(ArrayList<RouterPlace>):ArrayList<ArrayList...
sortSignal(ArrayList<ArrayList<RouterPlace>>):Array...
selectionSort(ArrayList<RouterPlace>):void
swap(ArrayList<RouterPlace>,int,int):void
getMinIndex(ArrayList<RouterPlace>,int):int
addRouter(WIFI,GeoModDat):RouterPlace


<<Java Class>>
Network
(default package)
real size: int
INIT SIZE: int
RESIZE: int
Network()
Network(int)
Network(Hotsp...
add(WIFI,Geo...
add(WIFI):void
add(Hotspots):...
resize():void
checkFS(WIFI)...
getHotspots():...
getReal size():...
<<Java Class>>
Hotspots
(default package)
real size: int
INIT SIZE: int
RESIZE: int
MAX SIZE: int
Hotspots()
Hotspots(Hotspots)
Hotspots(WIFI)
resize():void
add(WIFI):void
getDataOfdot():Geo...
setDataOfdot(Geo...
equalGeo(GeoMod...
getReal size():int
getWIFI():WIFI[]
sortRSSI():void
selectionSort(WIFI[]...
swap(WIFI[],int,int):...
getMinIndex(WIFI[],...
checkSignal(WIFI):int
-line
0..*
<<Java Class>>
WIFI
(default package)
firtseen: Date
mac: String
ssid: String
authmode: String
type: String
id: String
freq: String
channel: int
rssi: int
lat: double
lon: double
alt: double
accm: double
WIFI()
WIFI(String,String,String,int,int,double,...
WIFI(WIFI)
getid():String
setId(String):void
getMac():String
setMac(String):void
getSsid():String
setSsid(String):void
getAuthmode():String
setAuthmode(String):void
getType():String
setType(String):void
getFirtseen():Date
setFirtseen(Date):void
getChannel():int
setChannel(int):void
getRssi():int
setRssi(int):void
getLat():double
setLat(double):void
getLon():double
setLon(double):void
getAlt():double
setAlt(double):void
getAccm():double
setAccm(double):void
getFreq():String
setFreq(String):void
equals(WIFI):boolean
toString():String
-line
0..*
-dataOfdot
0..1
<<Java Class>>
GeoModDat
(default package)
firtseen: Date
Id: String
lat: double
lon: double
alt: double
GeoModDat()
GeoModDat(double,double,double)
GeoModDat(Date,double,double,do...
GeoModDat(Date,double,double,do...
getId():String
setId(String):void
getFirtseen():Date
setFirtseen(Date):void
getLat():double
setLat(double):void
getLon():double
setLon(double):void
getAlt():double
setAlt(double):void
toString():String
-position
0..1
<<Java Class>>
RouterPlace
(default package)
mac: String
SSID: String
signal: int
channel: int
RouterPlace()
RouterPlace(RouterPlace)
RouterPlace(RouterPlace,String,Date)
RouterPlace(String,int,double,double,double)
RouterPlace(String,Date,String,int,double,double,...
RouterPlace(String,String,Date,String,int,int,doub...
getSSID():String
setSSID(String):void
getChannel():int
setChannel(int):void
getMac():String
<<Java Class>>
toCSVTest
(default package)
toCSVTest()
main(String[]):void
check():void
<<Java Class>>
filterCSVTest
(default package)
filterCSVTest()
main(String[]):void
filter():void
<<Java Class>>
NetworkTest
(default package)
NetworkTest()
main(String[]):void
add():void
<<Java Class>>
kmlBaseTest
(default package)
kmlBaseTest()
main(String[]):void
add():void
<<Java Class>>
HotspotsTest
(default package)
HotspotsTest()
main(String[]):void
test():void
<<Java Class>>
WIFITest
(default package)
WIFITest()
main(String[]):void
test set get():void
test add():void
<<Java Class>>
csvBaseTest
(default package)
csvBaseTest()
main(String[]):void
add():void
<<Java Class>>
toKMLTest
(default package)
toKMLTest()
main(String[]):void
check():void