Test

+<<constructor>> Test()
+main(String[] args)(): void

Control

-a: Admin
-index1: int
-index2: int
-static ArrayList<> detalle: String

+<<constructor>> Control(a:Admin)
+callCheckin(): void
+reporte(): void
-Screentocharge(): void
+ multiDetalleAtencion(): ArrayList<String>
+detalleAtencion(): String

Admin

-admittedinterns(numerointernos): int
-docName: String = ""
-doctors: int = 5
- static LinkedList<> doctor: Doctor
-static LinkedList<> revisado: Animal

+<<constructor>> Admin()
-addinterns(numero:int): int
+star(): void
-checkin(): void
-DoctorManage(): String
-indoctor(name:String): Doctor
+void inreturnAnimal(tipo:String,nombre:String ,
            raza:String, edad:int,
            causaatencion:String): void
-returnAnimal(): void
-Screentocharg(): void

Animal

-tipo: String
-nombre: String
-raza: String
-causaatencion: String
-edad: int
-revisado: booleano
-revision: String
-caused: String
-tratamiento: String

+<<constructor>> Animal(tipo:String,nombre:String,
            raza:String,edad:int,
            in causa:String): String
+revisionresultados(resultado:String): String
+causadeenfermedad(causa:String): String const
+tratamiento(medicinas:String): String

<<hija Animal>>
Perro

+super

+<<constructor>> Perro(tipo:String,nombre:String,
            raza:String,edad:int,
            causa:String): String
+ToString(): String

<<hija Animal>>
Gato

+super

+<<constructor>> Perro(tipo:String,nombre:String,
            raza:String,edad:int,
            causa:String): String
+ToString(): String

<<hija Animal>>
Conejo

+super

+<<constructor>> Perro(tipo:String,nombre:String,
            raza:String,edad:int,
            causa:String): String
+ToString(): String

Doctor

-Numeroanimales: int
-nombre: String
+LinkedList<> Animal: Animal

+<<constructor generico>> Doctor()
+<<constructor>> Doctor(nombre:String)
+<<constructor>> Doctor(nombre:String,numerodeanimales:int)
+accetAnimal(numero:int): void
+addAnimal(Animal:int): void
+toString(): String

0...1   0...*   1   0..5   0...*