Universidad del Valle de Guatemala

Laboratorio Programación orientada a objetos

Sección 40

**Ejercicio #1**

Sebastian Estrada – 21405

**Análisis**

* ¿Qué propiedades y métodos tendrá cada clase?

Para el perro tenemos las siguientes propiedades:

* + Propiedades del perro
    - Scan
    - Rand
    - Suerte
    - Nombre
    - Sexualidad (hembra o macho)
    - Un color especifico
    - Una raza
    - Alguna otra característica especial.
  + Métodos del perro
    - Constructor perro
    - getSuerte
    - getNombre
    - getHembraMacho
    - getColor
    - getRaza
    - getOtracaract

Para el amigo se tienen las siguientes propiedades:

* + Propiedades del amigo
    - rand
    - scan
    - Nombre
    - Número de teléfono
    - Una dirección
    - Capacidad económica para buscar a su perro
    - La opción que desea ingresar al driver
    - Una recompensa (Es independiente de la capacidad económica)
  + Métodos del amigo
    - Constructor amigo
    - Llamar a la perrera
    - Imprimir anuncio
    - Publicar anuncio
    - Anuncio en el periódico
    - Mostrar el dinero restante
    - getOpcion
* ¿Qué tipo deben tener las propiedades y métodos de cada clase?

Para el perro tenemos las siguientes propiedades:

* + Propiedades del perro
    - Scan es tipo Scanner
    - Rand es tipo Random
    - Suerte es tipo int
    - Nombre es tipo String
    - Sexualidad es tipo String
    - Color es tipo String
    - Raza es tipo String
    - Otra característica especial es tipo String
  + Métodos del perro
    - Constructor perro no tiene tipo
    - getSuerte es tipo int
    - getNombre es tipo String
    - getHembraMacho es tipo String
    - getColor es tipo String
    - getRaza es tipo String
    - getOtracaract es tipo String

Para el amigo se tienen las siguientes propiedades:

* + Propiedades del amigo
    - Rand tipo Random
    - Scan tipo Scanner
    - Nombre es tipo String
    - Número de teléfono es tipo String
    - Una dirección es tipo String
    - Capacidad económica para buscar a su perro es tipo int
    - La opción que desea ingresar al driver es tipo int
    - Una recompensa (Es independiente de la capacidad económica) es tipo String
  + Métodos del amigo
    - Constructor amigo no tiene tipo
    - Llamar a la perrera es tipo void
    - Imprimir anuncio es tipo void
    - Publicar anuncio es tipo void
    - Anuncio en el periódico es tipo void
    - Mostrar el dinero restante es tipo void
    - getOpcion es tipo int
* ¿Cuáles deben ser los modificadores de visibilidad de los miembros en cada clase?

Para el perro tenemos las siguientes propiedades:

* + Propiedades del perro
    - Suerte es private
    - Nombre es private
    - Sexualidad es private
    - Color es private
    - Raza es private
    - Otra característica especial es private
  + Métodos del perro
    - getSuerte es public
    - getNombre es public
    - getHembraMacho es public
    - getColor es public
    - getRaza es public
    - getOtracaract public

Para el amigo se tienen las siguientes propiedades:

* + Propiedades del amigo
    - Nombre es private
    - Número de teléfono es private
    - Una dirección es private
    - Capacidad económica para buscar a su perro es private
    - La opción que desea ingresar al driver es private
    - Una recompensa (Es independiente de la capacidad económica) es private
  + Métodos del amigo
    - Llamar a la perrera es public
    - Imprimir anuncio es public
    - Publicar anuncio es public
    - Anuncio en el periódico es public
    - Mostrar el dinero restante es public
    - getOpcion es public
* ¿Qué parámetros serán requeridos por los métodos en sus clases? ¿Qué métodos diferenciarán su firma solo por parámetros, no nombre, y cómo lo harán?
  + Métodos del perro
    - Public Perro() No tiene parametros
    - Public int GetSuerte() No tiene parámetros
    - Public String getNombre() No tiene parámetros
    - Public String getHembraMacho () No tiene parámetros
    - Public String getColor () No tiene parámetros
    - Public String getRaza () No tiene parámetros
    - Public String getOtracaract () No tiene parámetros
  + Métodos del amigo
    - Public Amigo() (No tiene parámetro)
    - Public void LlamarPerrera (Perro p) como parámetro
    - Public void ImprimirAnuncio (Perro p) como parámetro
    - Public void publicarAnuncio (Perro p) como parámetro
    - Public void anuncioPeriodico (Perro p) como parámetro
    - Public void dineroRestante () No tiene parámetros
    - Public int getOpcion () No tiene parámetros
* ¿Cómo proveerá de valores iniciales a sus objetos? ¿Qué valores iniciales les asignará?

Para la clase amigo se utilizará un Scanner para asignar los Strings al nombre, numTel, dirección y la recompensa estos valores empiezan con un valor null. Para el valor del dinero se utilizará random para asignar un entero entre 0 y 1000. Por último la variable opción empieza con un valor de 0 despues mediante un scanner se cambia el valor y al encontrar al perro se settea el valor de opción como un 7.

Para la clase perro se utiliza un Scanner para asignar los Strings a las siguientes variables: Nombre, hembramacho, color, raza y otraCaracteristica, estas variables empiezan con un valor null. Mediante el paquete Random se le asigna un valor entero entre 0 y 20 a la variable suerte, en donde si es menor a 5 se encontrará al llamar a la perrera, si es menor a 10 se encontrara al publicar un anuncio en la ciudad y si es menor a 15 se encontrara tras publicar un anuncio en la perrera.

En el driver existe la variable amigo que comienza como un objeto amigo, la variable perro que empieza como un objeto perro, y la opción para el menú que guarda el valor escogido por el usuario o amigo para el menú.