

Modelado y Programación

Práctica 3

1 de septiembre de 2017

Realiza un programa en python para cada ejercicio:

1. Crea una clase llamada Videojuego, un videojuego tiene: nombre, género, precio y desarrollador. Realiza lo siguiente: Crea un constructor que inicialice los atributos anteriores, asegúrate de evitar que haya precios menores a \$0 Crea un método toString() para imprimir objetos de tipo Videojuego de la siguiente forma:

Nombre: The last of us

Género: Survival horror, aventuras

Precio: \$999.99

Desarrollador: Naughty Dog

Instancia 10 objetos de tipo Videojuego

Crea los siguientes métodos:

- elMasCaro(self)
Método llamado que regrese el objeto de tipo videojuego que sea el de mayor costo en la tienda (puede haber dos videosjuegos o más que sean los de mayor costo)
- rebajaVideojuego(self, vj)
Método que recibe un objeto de tipo videojuego y lo rebaja un 15 % de su costo original.
- getJuegosPorCompania(self)
Método que regresa todos los títulos que pertenecen a las diferentes compañías que haya
- getJuegosPorGenero(self, genero) Método que regresa todos los juegos que pertenezcan a la categoría genero que recibe el método como parámetro

2. Crea la clase llamada Fruta, una fruta debe tener tamaño (grande, mediana, pequeña) y color. Crea las clases Fresa, Mango, Sandía y Toronja que extiendan a Fruta; estas clases deben agregar las propiedades: sabor (dulce, ácido o amargo) y la propiedad conCascara que será una variable booleana que valdrá *true*, en caso de que la fruta posea una cáscara no comestible, como el caso de la sandía en nuestra lista de frutas.

Deberás crear una lista genérica por cada fruta con sabores, colores y tamaños al azar. Cada lista de frutas se puede llamar canasta.

Tendrás que implementar métodos que permitan realizar las siguientes operaciones:

- contiene(self, fruta)
Este método regresa true cuando una fruta está dentro de una canasta
- tamano(self)
Este método regresa la cantidad de frutas contenidas en una canasta
- revuelve(self)
Este método regresa una canasta de longitud mayor que contendrá frutas de dos tipos distintos
- limpia(self)
Este método recibe una canasta de frutas que pueden ser de tamaños y sabores distintos, pero sólo se queda con las frutas que son de sabor dulce y tamaño grande eliminando las que no lo son.
- cuenta(self)
Este método recibe una canasta de frutas e imprimirá en pantalla la cantidad de frutas de color distinto que haya, ejemplo: 3 rojas, 1 amarilla

3. Crea la clase *Especie*, de ella deberán heredar las clases *Herbívoro* y *Carnívoro*. Crea las clases: *Leon* y *Tiburón* que hereden de *Carnívoro* y las clases *Jirafa* y *Elefante* que hereden de *Herbívoro*, por otro lado crea la clase *SerHumano* y *Planta*. El *SerHumano* deberá heredar de *Carnívoro* y *Herbívoro*.

Crea el método *alimentarse()* el cual para los carnívoros al recibir como parámetro objetos de tipo *Carnívoro*, *Herbívoro* y *SerHumano* imprima en pantalla **A comer!!!**, en caso de recibir objetos de tipo *Planta* imprima **Nooo... gracias**.

Para los objetos de tipo *SerHumano* este método deberá recibir y aceptar bien objetos de tipo *Carnívoro* y *Herbívoro* salvo otros Seres Humanos.

Los herbívoros sólo pueden alimentarse de objetos de tipo *Planta*.

Crea un método en la clase *Especie*, el que tú decidas (dormir, comunicarse, tiempo de gestación, etc) y haz que tenga comportamientos distintos en cada clase; escribe en el Readme los detalles sobre el método que decidiste implementar.

4. Crea un archivo de excel con las siguientes columnas:

- Nombre del empleado
- Apellido paterno
- Apellido materno
- Enero
- Febrero
- Marzo
- ...
- Diciembre

Este archivo deberá tener 20 renglones, es decir, el registro de 20 trabajadores.

Cada valor en las columnas de los meses deberá tener una cantidad de tipo flotante.

La actividad consistirá en que realices la lectura de cada celda de todos los meses para cada trabajador y calcules el monto de un bono a pagarse que deberá ser equivalente al 15 % del total que haya percibido a lo largo del año, (puede haber columnas que tengan la cantidad 0.00 eso significa que ese mes el trabajador no percibió ingresos)

Tu programa deberá escribir en un archivo *extras.txt* el nombre y apellidos de cada trabajador junto con la cantidad de dinero adicional que recibirá.

Fecha de entrega: 8 de septiembre de 2017 al correo de Miguel Garrido miguel.garrido@ciencias.unam.mx, recuerda seguir los lineamientos de entrega.