

## ***Entornos de Desarrollo***

### **Boletín 3.1. Clases**

Una gallina es un ave doméstica que se cría en granjas. Para que cada gallina se sienta más feliz, nos gusta ponerle nombre, así podemos distinguirlas una de otras.

Cuando nos traen una nueva gallina a la granja lo primero que hacemos es asignarle un nombre y pesarla.

A las gallinas les encanta comer trigo y maíz, y hemos descubierto que en cada comida, engordan el equivalente a la mitad de la cantidad ingerida. Es decir, si le damos de comer a “Caponata” 100 gramos de trigo, engordará 50 gramos, si “Caponata” pesaba 1000 gramos, ahora pasará a pesar  $1000 + 50 = 1050$  gramos.

También hemos descubierto que una gallina comienza a poner huevos cuando su peso supera el kilo, y que jamás ponen ningún huevo cuando su peso es inferior. Por cada huevo puesto, las gallinas pierden 100 gramos de peso. Es decir, si “Caponata” pesa 1250 gramos y pone un huevo, entonces pasará a pesar  $1250 - 100 = 1150$  gramos.

A partir de nuestros conocimientos de las gallinas, hemos diseñado una clase que no sirva como representación de una gallina. La clase queda:

CLASE Gallina

nombre: CADENA

peso: ENTERO

Gallina (nombre:CADENA, peso:ENTERO)

    this.nombre = nombre

    this.peso = peso

FIN

MÉTODO come(cantidad: ENTERO)

    VARIABLES

        engorda: ENTERO

    COMIENZO

        engorda = cantidad / 2

        ESCRIBIR (this.nombre, " ha engordado ", engorda, " gramos")

        this.peso = this.peso + engorda

        ESCRIBIR ("ahora pesa ", this.peso)

FIN

MÉTODO ponHuevo():BOOLEANO

    VARIABLES

        result: BOOLEANO

    COMIENZO

        SI esPosiblePonerHuevo()

            result = CIERTO

            ESCRIBIR (this.nombre, " huevo puesto")

            this.peso = this.peso - 100 //un huevo son 100 gramos

        SINO

            result = FALSO

            ESCRIBIR (this.nombre, " no puede poner huevos")

        FIN SI

    DEVOLVER result

FIN

```

MÉTODO esPosiblePonerHuevo(): BOOLEANO
    VARIABLES
        result: BOOLEANO
    COMIENZO
        SI this.peso < 1000
            result = FALSO
        SINO
            result = CIERTO
        FIN SI

        DEVOLVER result
    FIN
FIN Gallina

```

Desde que disponemos de la clase gallina podemos utilizarla para gestionar a nuestras gallinas. Un ejemplo:

```

PROGRAMA Ejemplo
    VARIABLES
        a, b: Gallina
    COMIENZO
        a = new Gallina ("Caponata", 1200)
        b = new Gallina ("Pepa", 800)

        a.ponHuevo()
        a.come(300)
        a.ponHuevo()

        b.ponHuevo()
        b.come(200)
    FIN PROGRAMA

```

Que genera una salida por pantalla:

```

Caponata huevo puesto
Caponata ha engordado 150 gramos
ahora peso 1250 gramos
Caponata huevo puesto

Pepa no puede poner huevos
Pepa ha engordado 100 gramos
ahora pesa 900 gramos

```

Utilizando como base la clase Gallina, se pide:

1. No queremos que los programadores entrometidos que utilicen la clase Gallina, de forma malintencionada (o de forma fortuita), puedan alterar el estado de una gallina. Por ese motivo se pide aplicar a cada atributo y cada método la visibilidad correspondiente: público o privado.
2. Estamos recibiendo en la granja montones de gallinas. Incluso llegan por camiones. Debido a este incremento del negocio no tenemos la capacidad de ponerle nombre a cada una de nuestras gallinas. Se pide escribir un constructor que permita crear un objeto de tipo Gallina sin tener que asignarle forzosamente un nombre. Tan solo nos dedicamos a pesarlas.

3. Cuando una gallina está todo el día comiendo y poniendo huevos, llega un momento que perdemos la cuenta de su estado (nombre y peso). El hecho que la granja esté creciendo de esta manera nos obliga a rediseñar la clase *Gallina*. Ahora necesitamos “algo” que nos muestre el estado de cada gallina en un momento dado.
4. El mejor científico de gallinas acaba de descubrir, tras muchos años de estudio, que las gallinas por mucho que se alimenten nunca superan los 3 kilos (3000 gramos de peso). Esto es muy interesante y creemos que nuestra clase *Gallina* debe incorporar ese comportamiento en sus métodos.
5. El negocio de los huevos es una auténtica mina de oro. El problema que Hacienda ha venido a inspeccionar la granja. Y no se les ha ocurrido otra cosa que saber cuantos huevos ha puesto cada gallina. Nos hemos excusado diciendo que los informáticos que han diseñado la clase *Gallina* no habían tenido eso en cuenta, parece que los técnicos de Hacienda no están muy convencidos, pero nos han dado unos días de plazo, para que los informáticos modifiquen la clase *Gallina* y sean capaces de controlar el número exacto de huevos que ha puesto cada una de las gallinas.
6. El mejor experto en informática de gallinas ha visitado hoy la granja, y se ha llevado las manos a la cabeza cuando se ha dado cuenta que tenemos métodos públicos que no siguen las convenciones de la programación orientada a objeto. Nos han recomendado ponernos inmediatamente en contacto con nuestro grupo de informáticos para poner nuestra clase *Gallina* al más alto nivel de acceso a métodos públicos.
7. Ahora que disponemos de la clase *Gallina*, hemos recibido un encargo mucho más ambicioso: modelar el comportamiento de una granja completa. Nuestro cliente (*Farm & Hen Corporation*) construye granjas pensadas para un determinado número de gallinas, esto quiere decir que en la granja jamás vivirán más gallinas, pero puede ocurrir que por motivos varios exista, en un momento dado, un número menor de gallinas en el corral de la granja.  
  
Diseñar la clase *Granja* con un tamaño inicial (pensando en el número de gallinas), y donde es necesario tener en cada momento controlado el número actual de gallinas (que nunca podrá exceder el número máximo de gallinas con el que se construyó la granja).  
  
Por otro lado, en la granja se alimenta, con la misma cantidad de comida, a todas las gallinas por igual. Y a la hora de recolectar los huevos, se hace, igualmente, para todas las gallinas por igual. Teniendo en cuenta que debido al peso algunas gallinas pondrán huevos y otras no.  
  
La granja debe disponer de un mecanismo para la llegada de nuevas gallinas. En caso de estar la granja completa, rechazaremos la entrada de la nueva gallina.  
  
Y por último como Hacienda siempre quiere tener un control absoluto sobre la granja, es necesario disponer de un mecanismo que nos permita saber el número de huevos que han puesto todas las gallinas de la granja.
8. Tras un periodo de funcionamiento de las distintas granjas construidas, nuestro cliente, se ha percatado que durante el día las gallinas hacen su vida normal, pero durante la noche las gallinas duermen y por lo tanto no comen, ni ponen peso y tampoco ponen huevos. Las gallinas se duermen en cuanto se pone el Sol y se despiertan en cuanto sale.  
  
Es necesario adaptar el código desarrollado a este nuevo comportamiento. Sabiendo que todas las granjas (y por lo tanto todas las gallinas) están ubicadas en la misma zona horaria, es decir, se hace de día y de noche a la vez en todas las granjas.
9. Tras una intensa reunión con *Farm & Hen Corporation*, nos comunican que quieren diversificar el negocio. El mundo de las gallinas está muy bien, pero el mercado necesita otros productos

avícolas. En principio quieren introducir en las granjas patos, pero quizás en breve se decidan a poner otros tipos de aves.

Los patos son una aves muy parecidas a las gallinas: engordan cuando comen y ponen huevos. Pero los patos no engordan la mitad de lo que comen como las gallinas. Engordan la tercera parte de lo que comen. Es decir, si un pato come 60 gramos, engordará 20 gramos.

Otra peculiaridad de los patos es que dicen “*cuac, cuac*” y tenemos que disponer de un mecanismo para que los patos se expresen libremente.

Ademas la empresa ha decidido que no pondrá nombre a cada uno de los patos como hace con las gallinas (es muy estresante tener que buscarles un nombre a cada una). Los patos se numeraran de forma consecutiva: pato 1, pato 2, ...

Se pide reestructurar las clases existentes y crear nuevas clases para que se adapten al nuevo negocio de patos y gallinas. Hemos de ser conscientes que el negocio es probable que se modifique en breve introduciendo nuevas aves.

10. Como era de esperar, ahora en las granjas también viven codornices. Las codornices, viendo que los patos no paran de decir “*cuac, cuac*”, han reivindicado su derecho a decir “*cuchichí, cuchichí*”. Mientras, las gallinas, que pasan de eso de hacer tanto ruido, prefieren seguir calladitas.