**Cuestiones U2**

1. Los objetos se crean a partir de clases. La clase describe el tipo de objeto. Los objetos representan instancias individuales de la CLASE.
2. Los objetos se comunican unos con otros a través de Mensajes. Un objeto hace algo si invocamos a un METODO de ese objeto.
3. La cabecera de un método se denomina LA SIGNATURA.
4. Los atributos de un objeto definen su estado y los PARAMETROS definen su comportamiento.
5. Los métodos pueden tener PARAMETROS que proporcionan información adicional para que el método realice su tarea.
6. Los parámetros tienen un TIPO DE DATOS. El TIPO DE DATOS indica la clase de valores que el parámetro puede tomar.
7. Los métodos pueden devolver información vía un valor de retorno.
8. Objeto es sinónimo de INSTANCIA.
9. ¿Qué es un programa orientado a objetos? (POO, u OOP según sus siglas en inglés) Un programa orientado a objetos es un conjunto de objetos que interactúan entre sí a través de mensajes. ¿Qué son los mensajes? Es la información enviada al objeto cuando se invoca un metodo sobre el objeto.
10. Diferencia entre clase y objeto. Pon un ejemplo. Una ***clase*** describe un **tipo de objeto y u**n ***objeto*** representa instancias individuales de una clase. mi coche rojo es una instancia (Objeto) de la clase coche.
11. Tengo un ordenador de marca XXXXX con 512Mb de Ram y 160 Gb de disco duro. Mi ordenador es capaz de ejecutar programas, informarme acerca de la cantidad de memoria que posee y acerca de la capacidad del disco duro. De qué estoy hablando, ¿de un objeto o de una clase? Un objeto porque se identifica como un objeto en concreto (mi ordenador) de la clase ordenador (genérico). Identifica los atributos y su tipo y los métodos (servicios).

String marca;

Int ram;

Int capacidad;

Método

Void ejecutarPrograma();

Int getMemoria();

Int getCapacidad();

1. El ordenador anterior ofrece también la posibilidad de aumentar la memoria en una determinada cantidad. Escribe la signatura del método *anadirMemoria*() indicando en caso de que los haya, parámetros y valor de retorno. public void anadirMemoria(int nuevaMemoria);
2. Pon un ejemplo de un atributo de tipo primitivo y otro de tipo referencia. Private int tipoPrimitivo, private String nombreEstudiante1;
3. Imaginemos una clase Urna que modela una urna capaz de almacenar bolas de diferentes colores. Las bolas son objetos de la clase Bola. Sobre una urna se pueden añadir y extraer bolas. Define la signatura de los métodos *anadirBola*() y *extraerBola*() que añaden una nueva bola a la urna y devuelven la bola extraída de la urna respectivamente.

**public void** agregar(Bola b) {}

**public void** extraer(Bola b) {}

1. Para los siguientes ejemplos de clases indica cuáles podrían ser sus atributos y el tipo de éstos:
   * una clase TelefonoMovil cuyas instancias guardan el nº de teléfono móvil y la String numeroTelefono; cantidad de dinero que todavía queda de la recarga double saldoDisponible;
   * una clase Punto que modela a los puntos del espacio bidimensional doble x; double y;
   * una clase Hora cuyos objetos guardan la hora actual int hora; int minuto; int segundo;
   * una clase Bombilla cuyas instancias guardan la potencia y el estado (si está encendida o apagada) int potencia; boolean estado;
   * la clase Producto que modela el comportamiento de los productos de un almacén registrando para cada uno su identificador (una serie de caracteres) y la cantidad en stock String identificador, int cantidad
2. Escribe ahora las signaturas de alguno de los métodos que describen el comportamiento de las clases anteriores:

* podemos recargar un móvil añadiendo una determinada cantidad public void recargarMovil(double cantidad)
* un objeto Punto puede desplazarse a la derecha una distancia determinada public void moverDerecha (int distancia)
* una bombilla se puede encender y apagar public void setEstado (boolean estado) encendida =true/ apagada = false
* se puede sacar de un almacén una determinada cantidad de un producto public void int extraerProducto (string identificador, int cantidad);
* la clase Hora nos devuelve la hora actual en formato “hh:mm:ss” public string getHora ()

public String toString(){

//return String.valueOf(h)+”:”+String.valueOf(m)+”:”+String.valueOf(s);

return String.format(“%02d:%02d:%02d”, h,m,s);

}

* una bombilla es capaz de indicarnos cuál es su potencia public int getPotencia()