|  |  |
| --- | --- |
| Alumno: Cristian Krahulik, Lautaro Castillo,  Tomas Mastropietro, Alexander Filchel, Gianfranco Guzman | **Fecha límite de entrega: 20/3/25** |

El presente práctico tiene por objeto introducir al alumno en la programación orientada a objetos, observar sus particularidades y realizar un primer acercamiento a la codificación en JAVA.

A). Responda:

1. ¿Qué es un TAD?

Es un conjunto de valores y unas operaciones definidas sobre esos valores. Cada vez que deseemos emplear el TAD solo lo podemos hacer con las operaciones definidas, incluso no sabiendo cómo están implementadas.

1. ¿Qué es un Objeto o instancia? De 3 ejemplos de objetos.

Es un caso especifico que se da de una clase en concreto. Por ejemplo:

Clase Perro p1 = new Perro();

Clase Alumno a1 = new Alumno();

Clase Factura f1 = new Factura();

1. ¿Dónde se produce el encapsulamiento?

Se produce en los atributos y los métodos

1. ¿Cuáles son las semejanzas y las diferencias entre atributos y métodos?

Los atributos son características de un objeto (color, edad, tamaño)

Public int edad;

Los métodos son acciones que el objeto puede realizar.

Public int edad();

El método devuelve un valor entero.

1. ¿Qué es UML? ¿Y cómo se representa una clase en ese lenguaje?

Lenguaje unificado modelado

Es un grafico que describe el esquema de las clases y las relaciones entre ellas.

|  |
| --- |
| Auto |
| * color: String * marca: String |
| + arrancar(): void  + frenar(): void |

B). Una con flechas los calificadores de acceso con sus correspondientes:

1. public
2. private No se aplica a Clases
3. protected Sólo se aplica a atributos
4. friendly -default Sólo se aplica a Clases
5. static Se aplica a atributos, métodos y clases
6. final
7. primera letra en minúscula
8. primera letra en mayúscula

C). Realizando abstracción, escriba atributos y sus tipos, intentando que sean diferentes para cada ámbito de aplicación.

|  |  |  |
| --- | --- | --- |
| Clases | Ámbito en los que se aplica | |
| Comida | Para un comercio de venta | Para el experto que lo prepara |
| atributos | Plato | Ingredientes |
| Costo | Tiempo de cocción |
| tipoComida | Vencimiento |
| Vegano | Cantidad/proporción |
|  | Para el cliente que compra | Para la empresa que recicla los sobrantes |
| atributos | Precio | Comestible |
| Ingredientes | esOrganico |
| TipoComida | Cantidad |
|  |  |

D). Responda:

1. ¿Qué es el bytecode?
2. ¿Qué función cumple la JVM?
3. ¿Cuándo se ejecuta el recolector de basura de java?

E). Describa los calificadores de acceso de los miembros de esta clase. Desarrolle el código de los métodos de set y get para el atributo que corresponda.

**Cajero\_automático**

+ EFECTIVO: double

# acepta: boolean

-devuelve: String

+ procesarPedido(): void

+ pago(double): boolean

F). Indique verdadero (V) o falso (F):

Un constructor…

|  |  |  |  |
| --- | --- | --- | --- |
|  | V/F | Afirmación | Es falso debido a… |
| 1 | F | Es el método principal para ejecutar el programa. | Es el método para instanciar un objeto. |
| 2 | V | Crea objetos. |  |
| 3 | F | Devuelve el valor de un atributo privado. | No devuelve nada |
| 4 | F | Tiene sentencia return | No tiene sentencia return |
| 5 | V | Siempre hay uno por defecto, sin parámetros ni inicializaciones de atributos. |  |
| 6 | V | Se puede sobrecargar. |  |
| 7 | V | Su nombre se escribe con mayúscula. |  |
| 8 | F | Su calificador de acceso es static. | Su calificador es public |
| 9 | V | Su tipo de devolución no se indica y corresponde a la Clase. |  |

H). Dados las siguientes opciones determine cuál es constructor.

1. Auto(): void
2. Persona (boolean trabaja) : int
3. Paciente(int doc, int edad) X
4. calcularSuma() : void
5. determinarFeriados (): int
6. Juego(int puntaje) X

J). Defina para la clase Cajero\_automático el constructor que recibe valores para todos sus atributos

public Cajero\_Automatico(double EFECTIVO, boolean acepta, String devuelve) {  
 this.EFECTIVO = EFECTIVO;  
 this.acepta = acepta;  
 this.devuelve = devuelve;  
}