

Plan Formativo:	Nivel de Dificultad:
Full Stack Java Trainee	Básico/medio.
Módulo:	Tema:
Programación básica en Java.	Herencia y poliformismos
Título del Problema	Guía de metodológica Práctica de ejercitación, desarrollando conceptos de herencia y poliformismos, basados en programación orientada a objetos.
Intención del aprendizaje o aprendizaje esperado	
<ul style="list-style-type: none"> • Comprender y ejercitar conceptos algorítmicos, basado en lógica programática, ejecutando principios de herencia y poliformismos. • Desarrollar capacidades y mentalidad algorítmica para resolver problemas, para luego llevarlos a código de programación. 	
Planteamiento del Problema	
<p>1. Tenemos una microempresa, la cual está buscando desarrollar nuevos mercados. Para ello emplea una nueva función dentro de sus empleados, la figura de repartidor. Tenemos dos tipos empleados, la de figura de vendedor y otro repartidor, ambos son considerados empleados de la empresa.</p> <ol style="list-style-type: none"> a. El empleado tendrá los siguientes atributos, nombre, apellido, edad, salario. b. Además si este empleado tiene más de 40 años se le dará un bono adicional de 100mil pesos. c. Su Programa debe imprimir por pantalla, el nombre de dos empleados, aplicando el valor del bono mencionado según corresponda. <p>(Tiempo aproximado 1:20hrs)</p> <p>2. Realiza un programa en JAVA en el realices la gestión de una pequeña tienda de bebidas. Para ello manejarás objetos que tendrán 3 atributos:</p> <p>Nombre de un producto de supermercado Pecio Stock.</p> <p>Los productos de la tienda (objetos) estarán dentro de una lista y serán los que desee el usuario, al que le iremos preguntando si desea introducir otro.</p> <ul style="list-style-type: none"> • Nota 1: Modifica el constructor por defecto de tal forma que llame a 3 métodos: uno le pedirá al usuario el nombre de cada producto, otro método su precio y otro su stock. 	

• Nota 2: Realiza un método en el que le muestre al usuario un menú para que elija que producto comprar y luego le pregunte cuantas unidades desea de él. Luego se le preguntará si desea comprar otro producto o salir. Por último se le mostrará el importe total de la compra.

• Nota 3: Realiza otro método para actualizar el valor del stock de un producto cuando el usuario lo compre. En caso de que el usuario pida más unidades de las que quedan se le avisará por pantalla del error, se le comunicarán las unidades restantes y le preguntará si desea comprar otro producto.

(Tiempo aproximado 1:30 hrs)

Datos de apoyo al planteamiento

Pregunta Guía:

Recursos Bibliográficos.

<https://asociacionaepi.es/programacion-orientada-a-objetos-en-java/>

Nombre del Autor y Fecha:

Mario Canedo 12/12/2019