

1.) Si se desea modelar un conjunto de objetos del tipo "Vendedor de sopaipillas" aplicando los conceptos de la POO.

a. ¿Qué atributos debe tener? ¿tipo de dato?

El vendedor de sopaipillas debería tener los siguientes atributos:

"ventaDeldia" que refleje la cantidad de dinero ganado, de tipo double.

"tipoDesalsas" para la cantidad de salsas que piden los clientes, un array tipo String.

"cantidadDesopaipillasVendidas" que entregue la cantidad de sopaipillas vendidas por día, del tipo int.

"precioSopaipilla", que almacene el precio de la sopaipilla, tipo int.

b. ¿Qué métodos debe tener? ¿tipo de retorno? ¿parámetros de entrada?

"menu()", saludaría al cliente y preguntaría la cantidad de sopaipillas y las salsas de cada una, podría finalizar la operación, de tipo void sin parámetros de entrada.

"venderSopaipillas(int sopaipillas)", tomaría la variable de la cantidad de sopaipillas, aumentaría el contador de sopaipillas vendidas y el valor de ventaDeldia, también void.

"salir()", terminaría la ejecución del programa, tipo void, sin parámetro de entrada.

2.) Ahora considere además, la opción que los objetos de tipo vendedor de sopaipillas puedan "dar el vuelto para cada venta de sopaipillas" que realizan.

a. ¿Qué nuevos atributos debe tener? ¿tipo de dato?

Debería añadir los siguientes atributos:

"dineroRecibido", de tipo double, representa la cantidad de dinero que recibimos por la transacción.

"dineroCajachica", de tipo double, la cantidad de dinero que disponemos para dar vuelto.

b. ¿Qué nuevos métodos debe tener? ¿tipo de retorno? ¿parámetros de entrada?

“menu()” debe modificarse, para preguntar cuanto dinero va a entregar por la sopaipilla.

“entregarVuelto(double dineroEntregado, double valorPedido)” se va a ejecutar dentro de “venderSopapillas()”, consiste en revisar el dinero de la caja chica y modificar la cantidad de dinero luego de cada transacción, solo si es necesario dar vuelto.