

Developer Challenge

Ejercicio Hoja de Cálculo

Implemente una hoja de cálculo (estilo Excel). Bueno, pero bien sencillo, a no asustarse!! Esta tendrá tan solo las siguientes características:

- La hoja sólo manejará datos numéricos. Solo existirán dos tipos de celdas:
 - Celdas que contienen un valor numérico fijo
 - Celdas de tipo fórmula.
- Las fórmulas son bien sencillas, solo soportan una resta o suma entre dos celdas de la hoja de cálculo, ejemplo: <celda referencia A> +/- <celda referencia B>
- NO se preocupe en parsear o interpretar el texto de la fórmula (haga de cuenta que no existen los strings), suponga que se reciben estos tres parámetros por separado (las dos celdas de referencia y la operación)
- Si se quiere obtener el valor de una celda a la que nunca se le asignó ningún valor, debe retornar 0.
- La hoja de cálculo solo tendrá una columna.
- Se valorará un pequeño esbozo de diagramas de clases.

Una posible idea es tener el elemento HojaDeCalculo, el cual contiene la colección de celdas, y operaciones para agregar y obtener el valor de las celdas. Si se le ocurre una idea que le resulte más cómoda, úsela.

No debe preocuparse por escribir sentencias de interfaz gráfica. SOLO SE PIDE QUE CODIFIQUE EL MODELO O LA LÓGICA. Suponga que la interfaz gráfica será algo que se codificará más adelante.

Suponga que hay una función "Main" o "Test", donde por código usted mostrará cómo se asignan y obtienen valores de las celdas.

Puede elegir el lenguaje de su agrado que le resulte más conveniente.

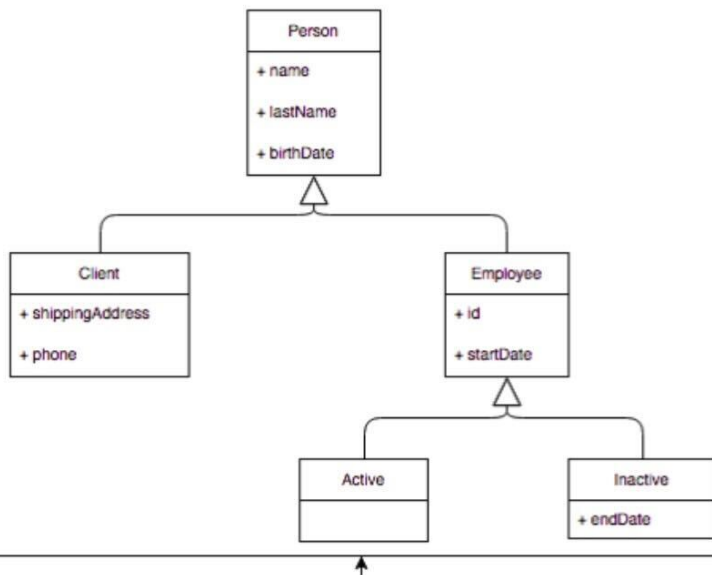
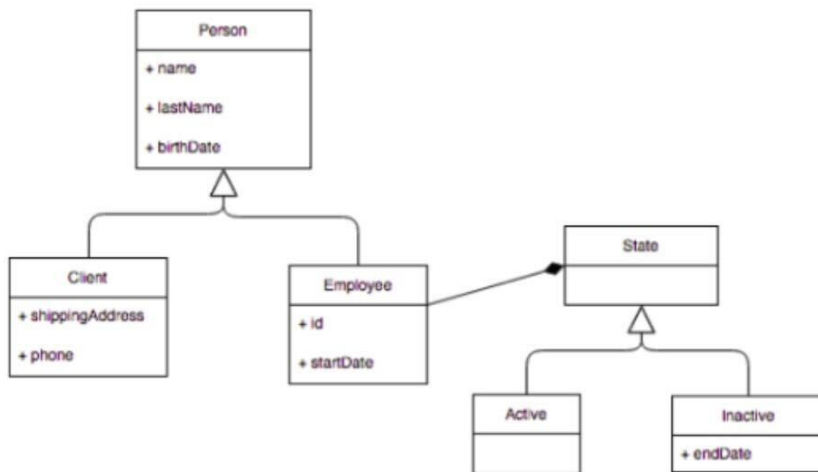
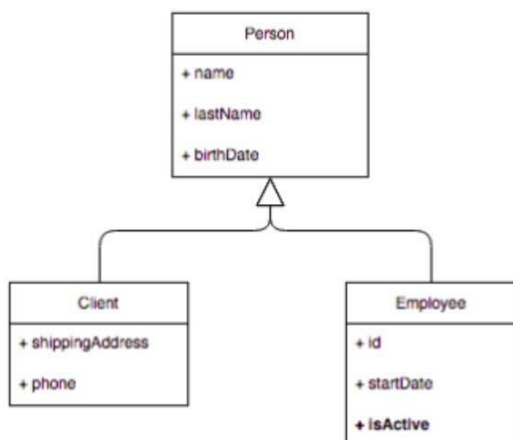
Opcionales, si te queda tiempo y ganas ...

- Hacer que la hoja de cálculo tenga múltiples columnas (grilla o matriz).
- Implementar Undo (deshacer) de N niveles sobre la asignación de valores a las celdas.

Multiple choice

Problema 1

Dado un escenario en el que necesitamos saber si un empleado está activo o inactivo porque la mayor parte de su comportamiento puede depender de ello, ¿cuál de las siguientes es la mejor solución de diseño?

OPTION 1**OPTION 2****OPTION 3**

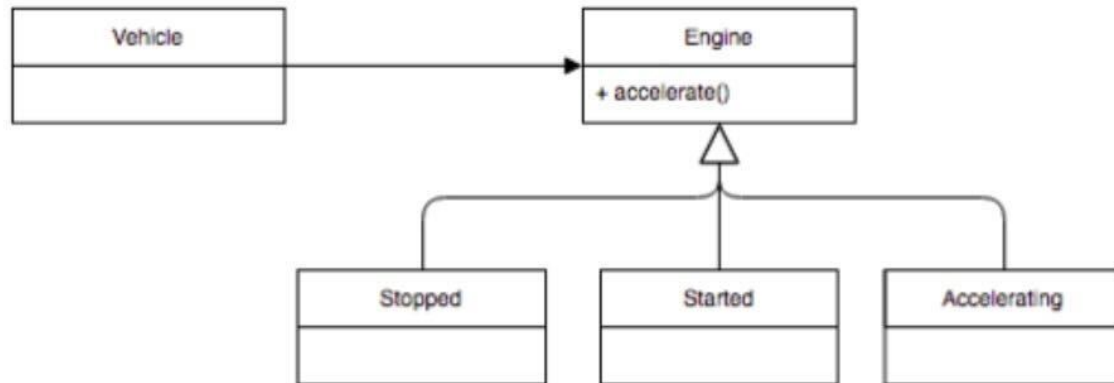
Opciones

- Opción 1, porque es la más clara
- Opción 2, porque es la más eficiente
- Opción 2, porque es la más orientada a objetos y más fácil de mantener
- Opción 3, porque es la más simple
- Ambas opción 1 y opción 2, son parecidas, Opción 3 es una pésima idea
- Opción 3 es fácil de implementar, las opciones 1 y 2 tienen demasiadas clases

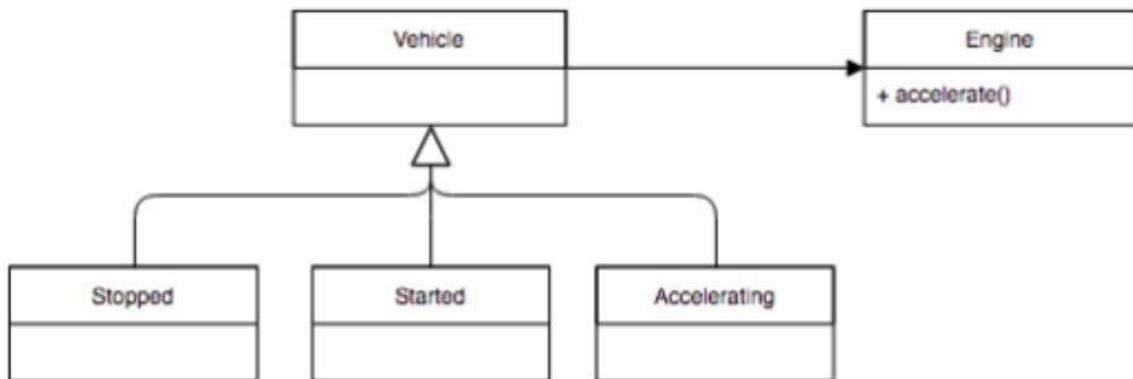
Problema 2

Necesitamos implementar el método de aceleración para algunos vehículos. Sabemos que un vehículo puede acelerar solo si se arranca el motor, y la implementación en el vehículo debe indicarle al motor que acelere (sabe cómo hacerlo). Entonces, queremos dejar de acelerar y, por supuesto, para poder hacer eso, el motor debe estar acelerando. ¿Cuál de las siguientes es la mejor solución de diseño?

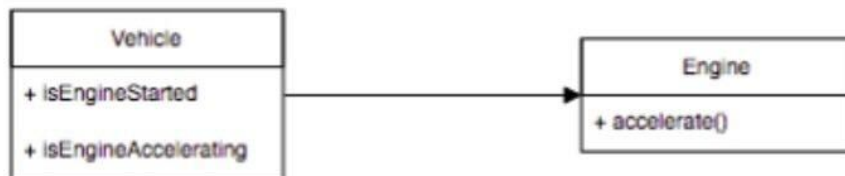
OPTION 1



OPTION 2



OPTION 3



Opciones

- Opción 2, porque me permite llamar al método de aceleración solo si el vehículo está encendido
- Opción 1, porque el estado del vehículo es en realidad el estado del motor
- Opción 3, porque es la más sencilla y eficiente

- Opción 1, porque la implementación del método de aceleración pertenece 100% al motor
- Opción 3, porque no se necesitan clases adicionales para implementar el método de aceleración correctamente
- Cualquiera de la Opción 1 o la Opción 2, porque son más o menos lo mismo
- **Agregaría una cuarta opción porque ninguna de esas es lo suficientemente buena**

