



Ejercicio Nivel 10

FábricaDeCarros

Descripción global

Se desea crear una aplicación que permita realizar, de manera gráfica, el diseño de autos. La aplicación permite la creación de nuevos diseños uniendo diferentes partes que se encuentran en un panel de partes. Se debe permitir la visualización de todas las partes así como la de diseños completos.

Objetivos

El objetivo de este ejercicio es que el estudiante comprenda y adquiera práctica en:

- El desarrollo de aplicaciones siguiendo un proceso incremental.
- La utilización de interfaces de Java (Interface) para disminuir el acoplamiento de clases en una aplicación.
- La utilización del concepto de herencia como mecanismo para la reutilización de código.
- La utilización de Interfaces ya existentes en Java como son Collection, List y otras.
- El recorrido de colecciones mediante Iteradores.
- El dibujo simple de 2D.
- El uso de eventos del ratón.
- La escritura de clases autopersistentes.

Los siguientes pasos conforman el plan sugerido para desarrollar el ejercicio. La idea es ir desarrollando y probando incrementalmente los métodos de las clases. **No se preocupe si las clases de la interfaz o de las pruebas (test) tienen errores. Estos desaparecerán cuando termine (correctamente) los cambios en el modelo del mundo.**

Este ejercicio debe ser realizado de manera **INDIVIDUAL**.

Preparación

1. Descargue del sitio web del curso el archivo demo de la aplicación (del enlace llamado **n10_fabricaDeCarros_demo**) y ejecútelo para conocer el funcionamiento esperado del programa.

2. Descargue del sitio web del curso el esqueleto del ejercicio (del enlace llamado **n10_fabricaDeCarros_esqueleto**) del ejercicio del sitio del curso. Descomprima este archivo e importe el proyecto llamado **n10_fabricaDeCarros** en Eclipse.
3. Lea el enunciado del problema disponible en:
n10_fabricaDeCarros/docs/specs/Descripcion.pdf.
4. Estudie el documento de requerimientos funcionales disponible en:
n10_fabricaDeCarros/docs/specs/RequerimientosFuncionales.pdf.
5. Estudie el documento de requerimientos no funcionales disponible en:
n10_fabricaDeCarros/docs/specs/RequerimientosNoFuncionales.pdf.
6. Estudie el documento de consideraciones adicionales de diseño disponible en:
n10_fabricaDeCarros/docs/specs/ConsideracionesAdicionalesDeDiseno.pdf.
7. Estudie el modelo del mundo diseñado para este ejercicio. Este modelo se encuentra en:
n10_fabricaDeCarros/docs/specs/ModeloConceptual.jpg. Identifique las clases, relaciones entre clases, constantes, atributos y métodos
8. Asegúrese de tener activado el uso de aserciones para la ejecución del programa. Ver el tutorial en:
http://cupi2.uniandes.edu.co/sitio/images/cursosCupi2/apo2/tutoriales/n7_assert.pdf

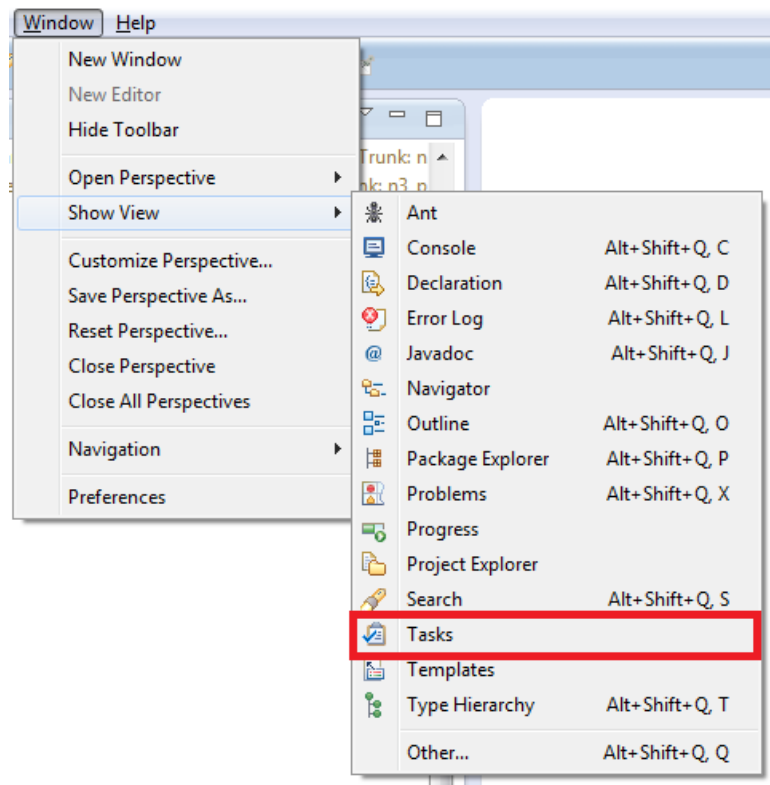
Desarrollo

Dentro del código del esqueleto se encuentran indicados los puntos donde usted debe realizar alguna modificación (añadir atributos, completar métodos, construir nuevos métodos, etc.), por medio de comentarios de la siguiente forma:

// TODO ParteX PuntoY: Breve explicación de la modificación que debe realizar.

Donde X hace referencia a una parte de la guía de trabajo, y Y hace referencia a un punto de la parte X. Para visualizar los **TODOs** vaya al menú Window -> Show View -> Tasks como se muestra en la siguiente figura:





En el siguiente vínculo encuentra un video que le explica cómo consultar los TODOs de un ejercicio:

<https://www.youtube.com/watch?v=pV54O42D2ow>

Parte 1: Completar la clase IParte

Estudie y familiarícese con los métodos definidos de la clase IParte.

Completar los TODOs que comienzan por: **//TODO Parte1 PuntoY**

Utilice (invoque) el método `verificarInvariante` para verificar el invariante en todos aquellos métodos de la clase que modifican el estado.

Parte 2: Completar la clase Parte

Estudie y familiarícese con los métodos definidos de la clase Parte.

Completar los TODOs que comienzan por: **//TODO Parte2 PuntoY**

Utilice (invoque) el método `verificarInvariante` para verificar el invariante en todos aquellos métodos de la clase que modifican el estado.



Parte 3: Completar la clase FabricaDeCarros

Estudie y familiarícese con los métodos definidos de la clase FabricaDeCarros.

Completar los TODOs que comienzan por: **//TODO Parte3 PuntoY**

Utilice (invoque) el método `verificarInvariante` para verificar el invariante en todos aquellos métodos de la clase que modifican el estado.

Parte 4: Crear la clase Llanta

Basado en el modelo conceptual, la descripción del problema y el documento de consideraciones adicionales de diseño, desarrolle los siguientes puntos:

Estudie y familiarícese con la implementación de las clases `Estencil` y `Chasis`. Estas le pueden servir de guía para la implementación de las siguientes clases.

1. Cree y complete la clase `Llanta`, ésta es abstracta y extiende de la clase `Parte`.

Nota: El método `pintarRines` es abstracto y debe ser llamado al final del método `pintar`.

Parte 5: Completar clase RinesGamaMedia

Estudie y familiarícese con los métodos definidos de la clase `RinesGamaMedia`.

Completar los TODOs que comienzan por: **//TODO Parte5 PuntoY**

Utilice (invoque) el método `verificarInvariante` para verificar el invariante en todos aquellos métodos de la clase que modifican el estado.

Parte 6: Crear clases tipo Llanta

Basado en el modelo conceptual, la descripción del problema y el documento de consideraciones adicionales de diseño, desarrolle los siguientes puntos:

Estudie y familiarícese con la implementación de las clases `RinesGamaMedia` y `RinesEconomicos`. Estas le pueden servir de guía para la implementación de las siguientes clases.

1. Cree y complete la clase `RinesDeLujo`, ésta extiende de la clase `Llanta`.



Parte 7: Crear clases tipo Chasis

Basado en el modelo conceptual, la descripción del problema y el documento de consideraciones adicionales de diseño, desarrolle los siguientes puntos:

Estudie y familiarícese con la implementación de las clases `DelanteraCamioneta`, `DelanteraSedan`, `TraseraCamioneta` y `TraseraSedan`. Estas le pueden servir de guía para la implementación de las siguientes clases.

1. Cree y complete la clase `DelanteraCompacto`, ésta extiende de la clase `Chasis`.
2. Cree y complete la clase `TraseraCompacto`, ésta extiende de la clase `Chasis`.

Parte 8: Crear clases tipo Estencil

Basado en el modelo conceptual, la descripción del problema y el documento de consideraciones adicionales de diseño, desarrolle los siguientes puntos:

Estudie y familiarícese con la implementación de la clase `Calavera` y `Teselacion`. Estas le pueden servir de guía para la implementación de las siguientes clases.

1. Cree y complete la clase `Rayo`, ésta extiende de la clase `Estencil`.

Parte 9: Completar la clase `FabricaDeCarros` con las nuevas partes.

Completar los TODOs que comienzan por: `//TODO Parte9 PuntoY`

Parte 10: Completar la clase `BarraMenu`

Completar los TODOs que comienzan por: `//TODO Parte10 PuntoY`

Parte 11: Completar la clase `PanelFabrica`

Completar los TODOs que comienzan por: `//TODO Parte11 PuntoY`

Validación

Para comprobar el funcionamiento de su ejercicio usted puede:

1. Ejecutar las pruebas automáticas disponibles en el ejercicio (validación funcional del mundo) que le permitirán verificar la correcta implementación de sus métodos. Para ejecutar las pruebas en el entorno de Eclipse presione click derecho sobre el paquete `"uniandes.cupi2.observatorio.test"` (o alguna de sus clases internas), seleccione la



opción *Run as* y posteriormente la opción *JUnitTest*. En el siguiente vínculo encuentra un video que le explica cómo ejecutar e interpretar las pruebas automáticas:

<https://www.youtube.com/watch?v=h3r7wSFaIOo>

2. Ejecutar el programa e interactuar con todas las opciones disponibles en la interfaz.

Entrega

Este ejercicio debe ser realizado de manera **INDIVIDUAL**.

1. Construya el archivo entregable con el ejercicio completo. No olvide revisar que su entrega cumple con lo especificado en las normas del curso referentes a entregas de ejercicios. Consultar:

<http://cupi2.uniandes.edu.co/sitio/index.php/cursos/apo1/normas-administrativas>

2. Entregue el archivo del ejercicio vía SicuaPlus, de acuerdo con las normas, fecha y hora de entrega.

NOTA. No olvide:

1. Renombrar el archivo de entrega con su login de estudiante. El nombre del archivo que contiene el ejercicio debe cumplir el siguiente formato:

n<nivel del ejercicio>_<login estudiante>.zip
(por ejemplo: n5_tsuarez.zip)

2. En el siguiente vínculo encuentra un video que le explica cómo indentar el código de su proyecto:

<https://www.youtube.com/watch?v=BH9H0e-Z56E>

