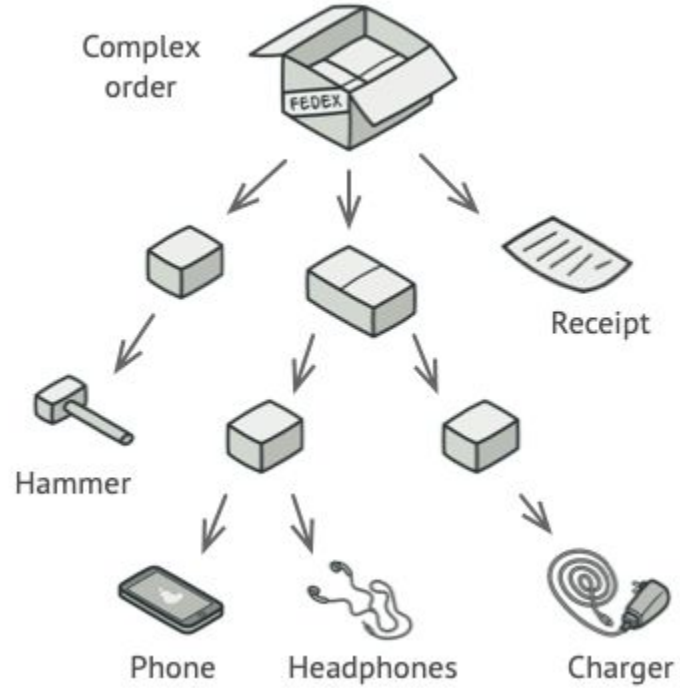




Composite

Ejemplo 1



Ejemplo 2 - Gráficos

- Supongamos que queremos modelar un editor gráfico. En particular nos interesa el manejo de las figuras.
- Podemos tener las siguientes figuras:
 - Líneas
 - Círculos
 - Texto
 - Dibujo
- Un dibujo contiene uno o varios de los objetos en la lista.
- Lo que interesa de las figuras es dibujarlas.
 - Un dibujo se dibuja dibujando las partes componentes
- ¿Cómo lo modelamos?

Ejemplo 3 - File system

- Supongamos que tenemos que diseñar un sistema que simula un **filesystem** de un sistema operativo.
- Se pueden tener archivos y carpetas.
- En las carpetas se pueden tener archivos u otras carpetas.
- Se quieren tener operaciones como:
 - Tamaño
 - En un archivo es el tamaño del mismo
 - En una carpeta es la suma del tamaño de los archivos que contiene.
- ¿Cómo lo diseñamos?

Ejemplo 4: Búsqueda

- Modele una búsqueda que puede usar operadores AND y OR.
- También operador NOT.
- Las búsquedas operan sobre una interfaz *Persona* que sabe responder a los métodos `getEdad` y `getCantidadHijos`
- Los criterios de búsqueda son: `edadMenorA`, `edadMayoA`, `cantidadDeHijosIgualA`, `ProvincialIgualA`, .
- Modele utilizando POO la clase `Búsqueda` >> `buscar(List<Persona> personas)` que representa una búsqueda específica con valores específicos. Por ejemplo

((`edadMenorA 50`) AND (`cantidadDeHijosIgualA 2`)) OR (`provincialIgualA BuenosAires`)

Cuándo uso composite?



Conclusiones

- Permite tratar a objetos individuales y a composiciones de objetos de la misma manera, lo que facilita la creación de estructuras complejas y anidadas.
- Recursión

Composite - Continuación

- Participantes:
 - Client: manipula los objetos indistintamente.
 - Leaf: define el comportamiento de los objetos primitivos.
 - Composite: define el comportamiento de objetos compuestos.
 - Component: define el protocolo.

