Opciones de refactorización con NetBeans

Índice de contenidos:

Introducción.

En este tutorial vamos a prácticar distintas formas de refactorización que trae NetBeans
Instrucciones: Importa en Netbeans el proyecto Refactoring.zip que viene con este tutorial.

1.- Refactorización: Introducir Variable (ALT+SHIFT+V)
Paso 1: En source packages despliega packageB y abre la ClassC.
Paso 2: Selecciona el valor de la variable d (3.3). Haz clic en el icono de la bombilla que te aparece en el número de línea. Se desplegará un conjunto de opciones, entre ellas Introduce Variable. Haz clic aquí.
Nota: Otra forma de llegar a esta opción es haciendo clic en la opción de Menú Refactor>Introduce>Variable
Paso 3: En esta ventana de dialogo nos ofrecen cambiar el nombre, hacer a la variable final e incluso reemplazar el nombre en todas las ocurrencias de esa variable. Ponle el nombre a la variable de nuevo_nombre y desmarca la opción de Replace All Ocurrences
¿Qué ha ocurrido? ¿Para qué puede ser útil esto?

2.- Refactorización: Introducir Constante (ALT+SHIFT+C)
Paso 1: Ahora dentro de packageB y abre la ClassB
Paso 2: Selecciona el valor de la variable s (“text”).
Paso 3: Haz clic en Menú Refactor>Introduce>Constant
Paso 4: En esta ventana de dialogo cambia el nombre y escribe MI_TEXTO, asegurate de tener marcada la opción de Replace All Ocurrences
¿Qué ha ocurrido?

3.- Refactorización: Introducir Field (ALT+SHIFT+F)
Paso 1: Ahora dentro de packageB y abre la ClassA
Paso 2: Selecciona el valor de la variable x (3+3).
Paso 3: Haz clic en Menú Refactor>Introduce>Field
Paso 4: En esta ventana de dialogo cambia el nombre y escribe nuevo_campo, selecciona acceso private asegurate de tener marcada las opciones Declare Final y Replace All Ocurrences. En Initialize marca field
¿Qué ha ocurrido? ¿cómo mejora esta refactorización el código?

4.- Refactorización: Introducir Parámetro (ALT+SHIFT+P)
Paso 1: Ahora dentro de packageA y abre la ClassA
Paso 2: Ve a la línea 71 del código y selecciona la variable a.
Paso 3: Haz clic en Menú Refactor>Introduce>Parameter
Paso 4: En esta ventana de dialogo solo haz clic en el botón Refactor
¿Qué ha ocurrido? ¿cómo mejora esta refactorización el código?

5.- Refactorización: Introducir Método (ALT+SHIFT+P)
Paso 1: Ahora dentro de packageA y abre la ClassA
Paso 2: Ve a la línea 28 del código y selecciona las líneas desde la 28 a la 31.

Paso 3: Haz clic en Menú Source>Fix code y haz clic en Introduce Method
Paso 4: En esta ventana de dialogo cambia el nombre y escribe MostrarSuma, selecciona acceso private
¿Qué ha ocurrido? ¿cómo mejora esta refactorización el código?

6.- Refactorización: Introducir Local Extension (ALT+SHIFT+X))
Paso 1: Ahora dentro de packageA y abre la ClassA
Paso 2: Ve a la línea 18 del código y seleccionala (private void MostrarSuma()).
Paso 3: Haz clic en Menú Refactor>Introduce>Local Extension
Paso 4: En esta ventana de dialogo haz clic en Preview
Paso 5: En esta ventana de dialogo haz clic en Do Refactoring

¿Qué ha ocurrido? ¿cómo mejora esta refactorización el código?

7.- Refactorización: Encapsular campos
Esta refactorización consiste en coger los campos (atributos) públicos y convertirlos en privados. Además nos creará los correspondientes métodos getters y setters
Paso 1: Ahora dentro de packageC y abre la Person
Paso 2: Haz clic en Menú Refactor>Encapsulate Fields..
Paso 3: En esta ventana de dialogo haz clic en Select All. Asegúrate de seleccionar:
Sort: Getters/Setters Pairs
JavaDoc: Create default comments
Fields´s visibility: private
Accesors visibility: public
¿Qué ha ocurrido? ¿cómo mejora esta refactorización el código?

8.- Refactorización: Extraer interfaz.
Crea una nueva interfaz con los elementos seleccionados y los métodos no estáticos de la o la interfaz
9.- Refactorización: Extraer Superclase
Crea una nueva clase abstracta, cambia la clase actual de extender la nueva clase, y mueve los métodos y campos seleccionados para la nueva clase.
10.- Refactorización: Mover una clase interior a un nivel exterior

Ejercicios
1.- Realiza una tabla comparativa de los distintos métodos de refactorización de los IDE´s Netbeans y eclipse
2.- Realiza el anterior tutorial con el IDE eclipse.
3.- Crea una nueva clase en packageC llamada MiClase.
Añade los siguientes atributos / campos: (el nombre lo decides tu)
- 3 campos tipo String
- 2 campos tipo int
- 1 campo tipo double
- 2 campos tipo boolean
- 1 campo tipo date

Añade los siguientes métodos: (el nombre lo decides tu)
- 2 constructores
- 3 métodos estáticos: 1 privado 2 publicos
- 2 métodos públicos
- Los métodos deben contar con parámetros

a.-) Usa la refactorización **Encapsular campos** para crear los getter y setters
b.-) Usa la **Refactorización: Extraer interfaz**
c.-) Usa la **Refactorización: Extraer Superclase**
d.-) Usa la **Refactorización: Mover una clase interior a un nivel exterior**