

Mas Conceptos de Orientada a Objetos

Juan David Balcázar Bedoya, código: 2879963

John Jairo Montoya Gómez, código: 2879540

Taller

1. Responda las siguientes preguntas de Colecciones:

a. ¿Cuál es la principal diferencia entre Set y Map?

List es una lista que tiene un orden arbitrario y permite acceder a un elemento por orden, en un list se puede introducir elementos repetidos.

Set es una lista pero se le conoce como conjunto. La diferencia que tiene con un list es que no se le puede añadir duplicados.

b. ¿Qué pasa cuando se agrega un valor de tipo primitivo a una colección?

Los valores de tipo primitivos se pueden almacenar en arrays estáticos.

Para agregar un valor de tipo primitivo primero tiene que ser puesto en un objeto contenedor.

c. En clase se estudió, ArrayList, HashMap, TreeMap; busque otra colección y realice una tabla comparativa entre las 4 colecciones, en donde se muestre las características especiales de cada una, cuando usar una con respecto a la otra, etc.

ArrayList	HashMap	TreeMap	LinkedList
<i>Contiene tantos objetos como necesitamos, almacenando los elementos en un array de objetos. Esta clase tiene varios constructores</i> <i>ArrayList implementa la interfaz List y extiende de la clase abstracta AbstractList.</i>	<i>Esta clase mapea claves con valores, pero no permite claves duplicadas porque se solaparía el valor.</i> <i>Esta clase no garantiza el orden de los elementos ni que no puedan cambiar de orden. HashMap implementa la interfaz Map y extiende de la clase abstracta AbstractMap.</i>	<i>Esta clase implementa SortedMap y se basa en un árbol binario.</i>	<i>Almacena los elementos en una lista vinculada y permiten un acceso a ella de manera secuencial, pero su uso no es tan eficiente como los arrays.</i>

2. Responda las siguientes preguntas (Tema de parcial)

a. ¿Por qué se puede decir que un método abstracto es un contrato?

Un método abstracto consiste en la especificación de la firma en la declaración del método en cuestión. Este deberá ser implementado por los tipos de datos heredados. Implícitamente un método abstracto es virtual ([virtual](#)) y deberá ser sobrescrito ([override](#)).

b. ¿Cuál es el propósito principal de la clase Abstracta?

Permite desarrollar una jerarquía de clases en el que algún comportamiento está presente en todas ellas pero se comporta de forma distinta para cada una. Es una clase que declara la existencia de métodos pero no la implementación de dichos métodos.

Una clase abstracta puede contener métodos no-abstractos pero al menos uno de los métodos debe ser declarado abstracto.

c. ¿Cómo se puede tener ventaja del polimorfismo teniendo un método abstracto?

Las subclases deben redefinir o sobre-escribir el método abstracto. En cada subclase, se mantiene la misma firma del método, pero el cuerpo puede ser distinto. Cuando en el código no incluye ningún tipo de especificación sobre el tipo de datos sobre el que se trabaja, esto permite el polimorfismo dinámico o polimorfismo paramétrico. Así, puede ser utilizado a todo tipo de datos compatible.

d. Falso o verdadero, Si es falso Justifique.

i. Todos los métodos en una clase abstracta deben ser declarados como métodos abstractos.

Falso- No todos los métodos de una clase abstracta deben ser abstractos, solo si una clase tiene un método abstracto, esta clase ya se considera abstracta.

ii. Si una superclase declara un método abstracto, una subclase debe implementar ese método.

Verdadero

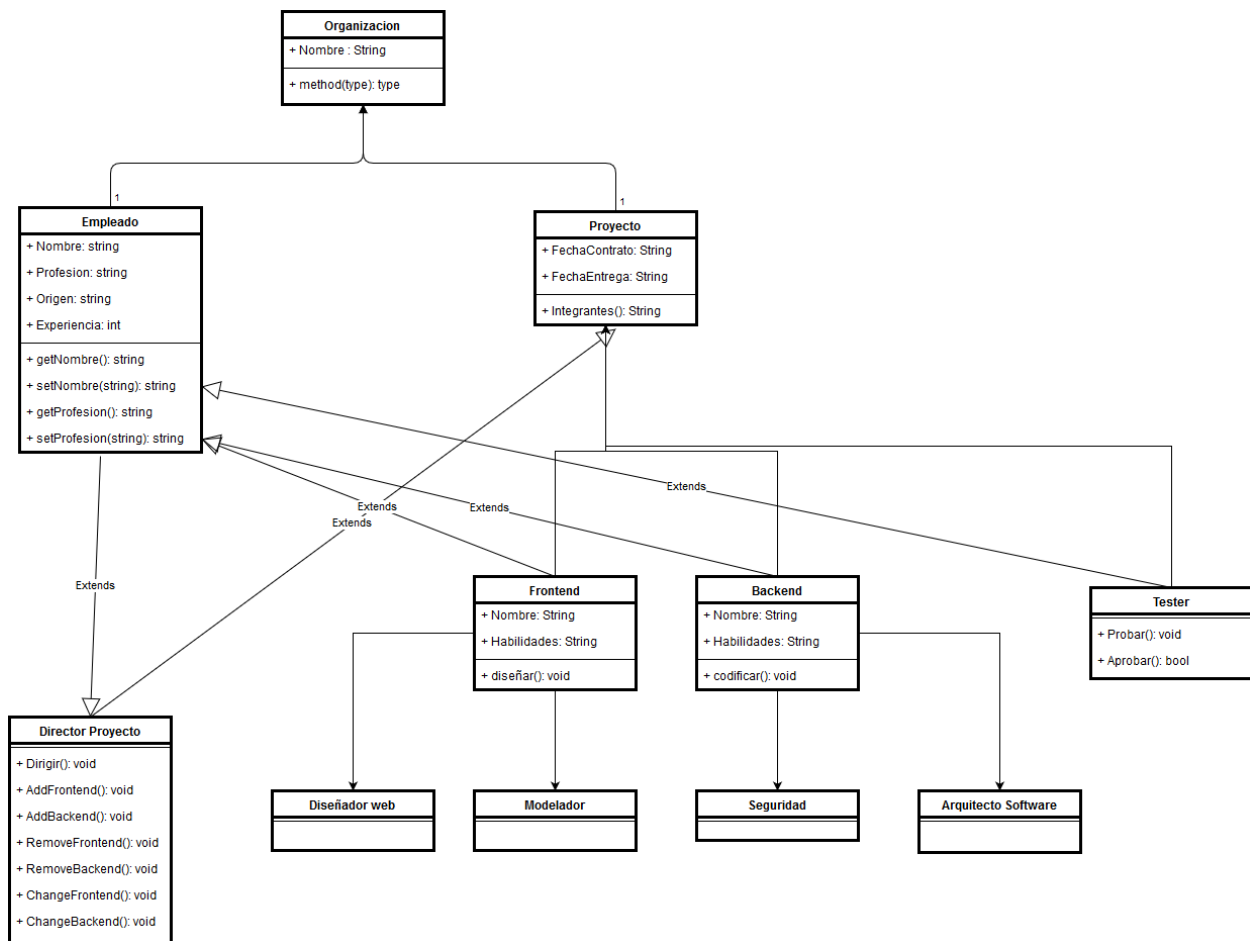
iii. Un objeto de una clase que implementa una interfaz puede ser pensado como un objeto de ese tipo de interfaz.

e. ¿Cuándo se usa una clase abstracta y cuando una interfaz?

La clase abstracta cuando se desarrolla una jerarquía de clases en que algún comportamiento está presente en todas ellas pero se materializa de forma distinta para cada una.

La interfaz cuando se necesita herencia, y como en java una clase sólo puede heredar de otra clase pero si puede tener multiple interfaces.

6.



Bibliografía

<http://miveticom.blogspot.com.co/2008/05/diferencia-entre-listset-y-map.html>

<http://vayajava.blogspot.com.co/2008/05/diferencias-entre-las-colecciones-list.html>

http://profesores.fi-b.unam.mx/carlos/java/java_basico4_8.html