

Estructuras de Datos

Práctica 4

Colección Abstracta

Alumno: Luis Enrique Altamirano Niño

4 de marzo de 2018

1. Actividades

Actividad 1 ✓
Para familiarizarte con el ambiente de trabajo, revisa la documentación de las clases `Collection<E>` e `Iterator<E>` de Java.

Actividad 2 ✓
Revisa la documentación del paquete `java.util`. ¿Qué estructuras de datos encuentras incluidas?

Respuesta:

Listas (`List<E>`), Conjuntos (`Set<E>`), Vectores (`Vector<E>`), Pilas (`Stack<E>`), Arreglos (`Arrays`), etcétera.

2. Preguntas

1. ¿Qué estructuras de datos incluye la API de Java dentro del paquete que importas, `java.util`?

Respuesta:

Listas (`List<E>`), Conjuntos (`Set<E>`), Vectores (`Vector<E>`), Pilas (`Stack<E>`), Arreglos (`Arrays`), etcétera.

2. ¿Cuál crees que es el objetivo de la interfaz `Collection`? ¿Por qué no hacer que cada estructura defina sus propios métodos?

Respuesta:

El propósito de la interfaz `Collection` es ahorrar tiempo y definir operaciones comunes para cualquier estructura de datos, sin esto se perdería tiempo programando los mismos métodos (que funcionan igual) para cada estructura.

3. ¿Qué métodos permite la interfaz `Collection` que su funcionalidad sea opcional? ¿Qué deben de hacer estos métodos opcionales si no se implementa su funcionalidad? ¿Por qué crees que son opcionales?

Respuesta:

Los métodos opcionales son `add`, `addAll`, `clear`, `remove`, `removeAll`, `retainAll`. Si no se implementa su funcionalidad se debe lanzar una excepción `UnsupportedOperationException`. Son opcionales porque, la implementación de estos depende de como lo defina cada estructura de datos.