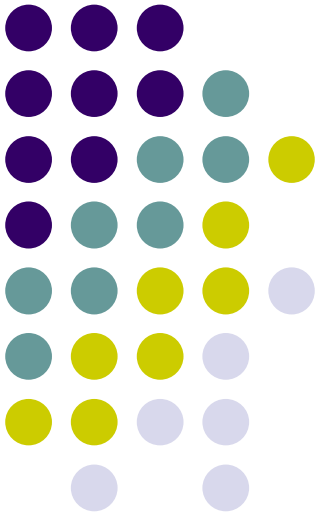
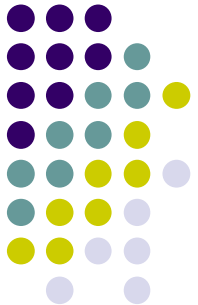
Metodologías de Diseño y Programación

Implementación

Colecciones

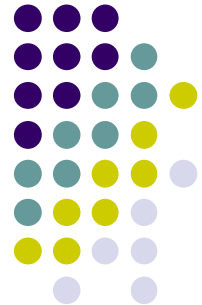


Contenido



- Introducción
- Colecciones de Objetos
- Tipos de Colecciones
- Colecciones Genéricas
- Ejemplos de Uso

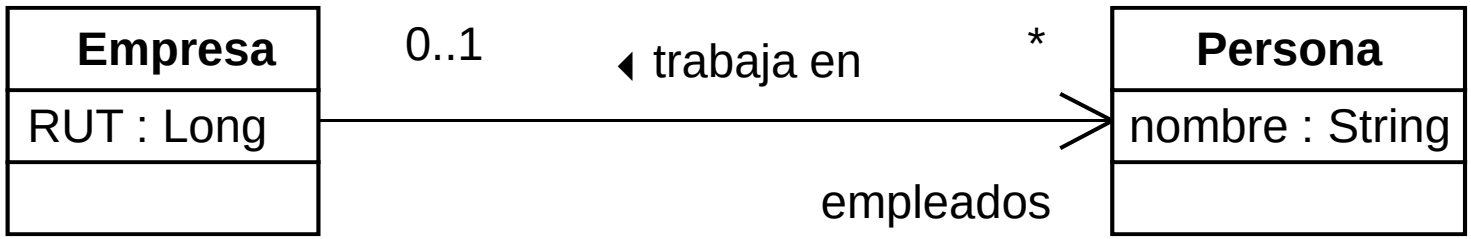
Introducción



- La implementación de asociaciones usualmente requiere del uso de colecciones para permitir links con “muchos” objetos
- Se utilizan colecciones genéricas para evitar su implementación para cada tipo de elemento a agrupar
- El tipo de las colección genérica a utilizar depende de la funcionalidad esperada sobre la colección

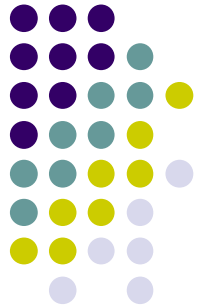
Colecciones de Objetos

- Las colecciones de objetos son una herramienta fundamental para la implementación de muchas de las asociaciones presentes en un diseño



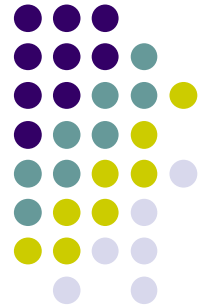
El pseudoatributo `empleados` de la clase `Empresa` introduce la necesidad de una colección de `Persona`

Colecciones de Objetos (2)



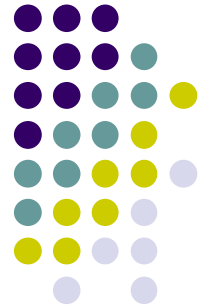
- Las colecciones deben permitir
  - Agregar y quitar elementos
  - Realizar iteraciones sobre sus elementos
  - Realizar búsquedas de elementos por clave
    - En caso de que los elementos tengan una
  - Realizar búsquedas diversas
    - Según una condición sobre los elementos

Tipos de Colecciones



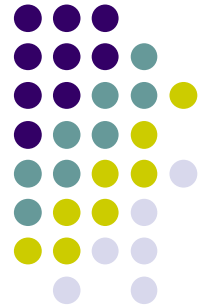
- Existen distintos tipos de colecciones que varían en las operaciones que brindan
  - Están especificadas como interfaces
- Cada tipo de colección se puede realizar sobre diferentes estructuras
  - Cada implementación diferente está definida en una clase separada
- Al usar una colección es necesario elegir su tipo y su realización

Colecciones Genéricas

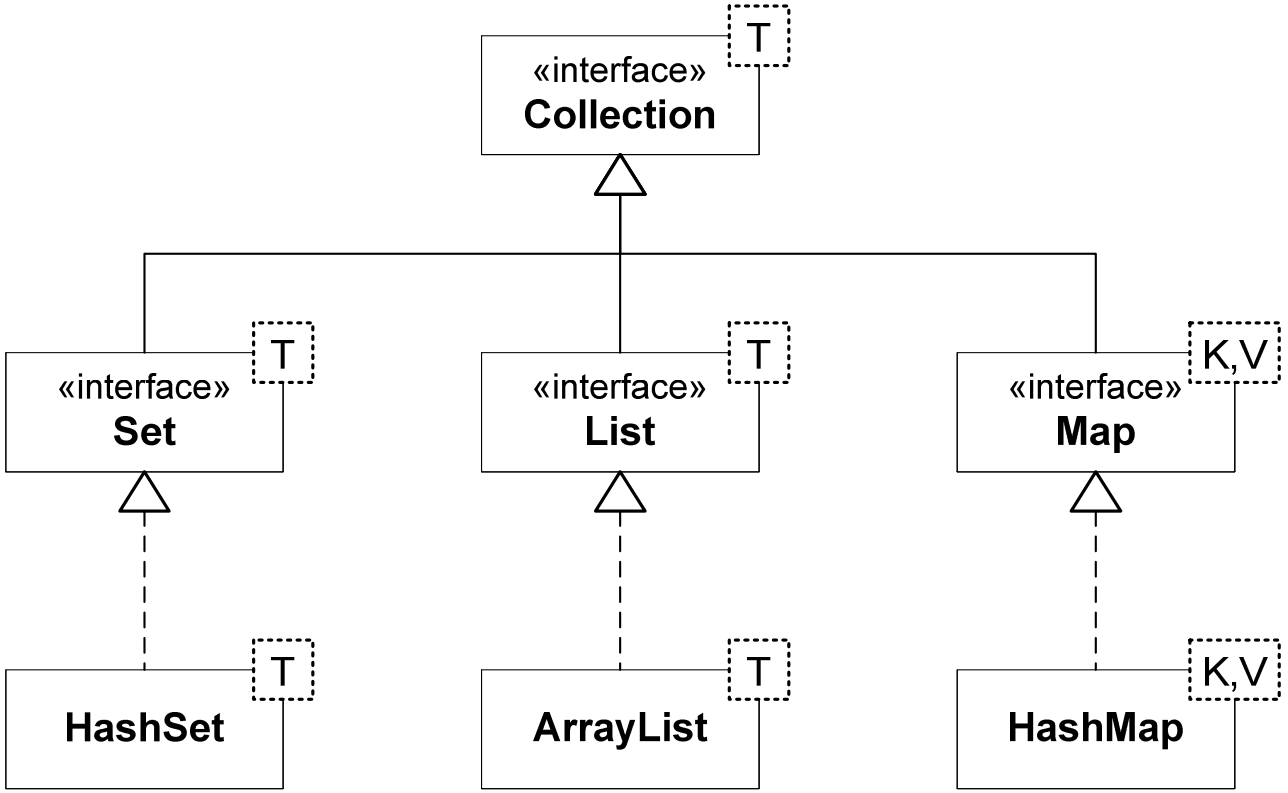


- Las colecciones y sus realizaciones están definidas en forma genérica (el tipo de elementos a contener es un parámetro)
- Una vez decidido el tipo de colección y su realización debe indicársele el tipo de elementos que contienen para poder usarla
- Por ejemplo el pseudoatributo `empleados` de la clase `Empresa`:
  - Se declara de tipo `Set<Persona>`
  - Se define de tipo `HashSet<Persona>`

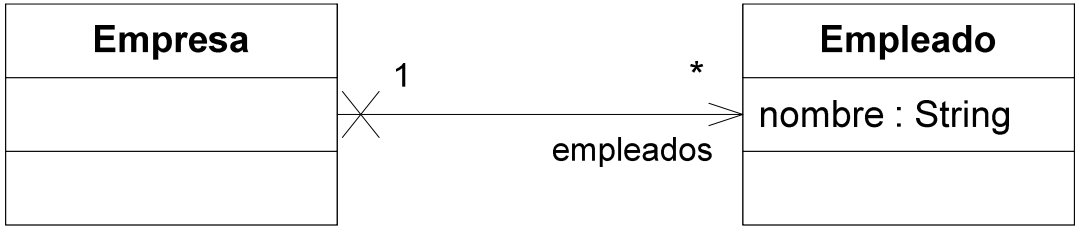
Colecciones Genéricas (2)



- Algunos tipos de colecciones, y para cada uno de ellos una posible realización



Ejemplo: Set<T>

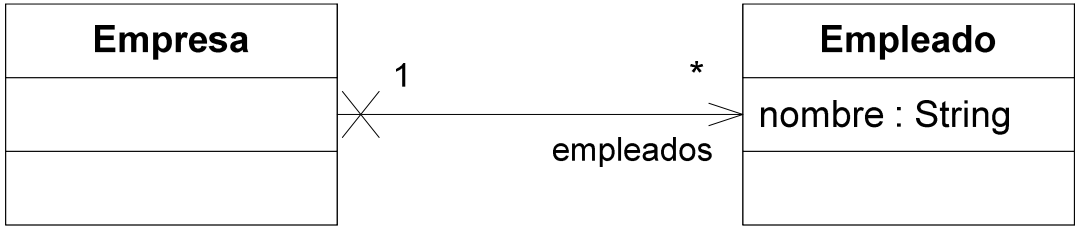



```
// declaración
private Set<Empleado> empleados;

// instanciación
this.empleados = new HashSet<Empleado>();

// tamaño
this.empleados.size()
```

Ejemplo: Set<T>



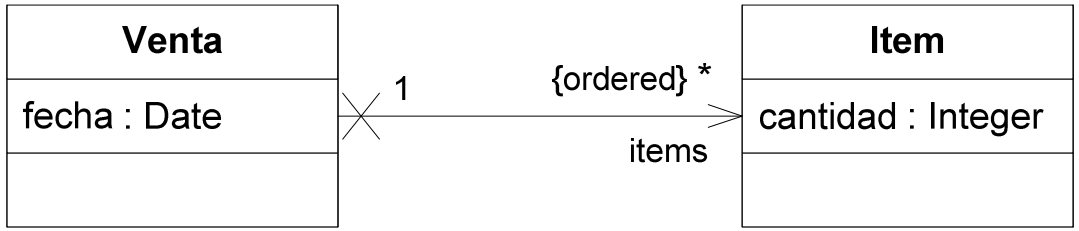

```
// agregar e:Empleado
this.empleados.add(e)

// remover
this.empleados.remove(e)

// contiene?
this.empleados.contains(e)

// iterar
for(Empleado e : this.empleados) { … }
```

Ejemplo: List<T>

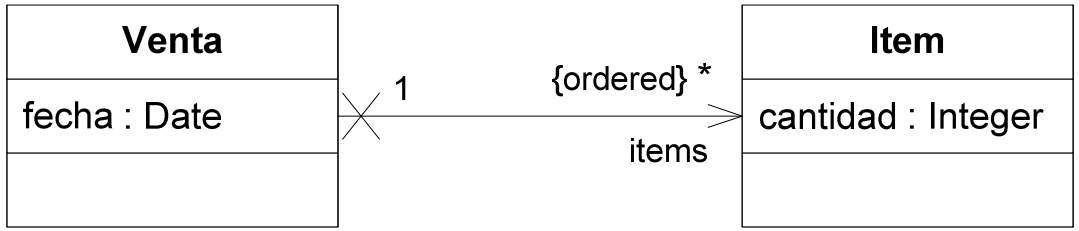



```
// declaración
private List<Item> items;

// instanciación
this.items = new ArrayList<Item>();

// tamaño
this.items.size()
```

Ejemplo: List<T>



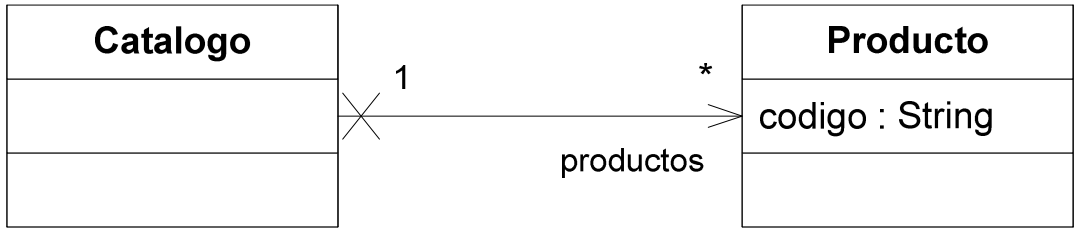

```
// agregar i:Item
this.items.add(i)
this.items.insert(indice, i)

// remover
this.items.remove(i)
this.items.remove(indice)

// obtener
i = this.items.get(indice)

// iterar
for(Item i : this.items) { … }
```

Ejemplo: Map<K,V>

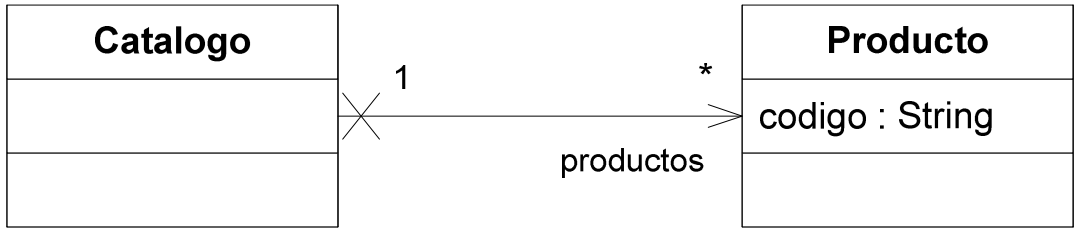



```
// declaración
private Map<String,Producto> productos;

// instanciación
this.productos = new HashMap<String,Producto>();

// tamaño
this.productos.size()
```

Ejemplo: Map<K,V>




```
// agregar p:Producto con código c:String
this.productos.put(c, p)

// remover
this.productos.remove(c)

// obtener
p = this.productos.get(c)

// iterar
for(Producto p : this.productos.values()) { … }
```