Package JavaContainerPackage

package JavaContainerPackage

All Classes and Interfaces Interfaces Classes

Class	Description
Driver	
Iterator<T>	
JavaContainer<T>	
JavaSet<T>	A generic set implementation.
JavaVector<T>	A generic vector implementation.

Package JavaContainerPackage

Class Driver

java.lang.Object
    JavaContainerPackage.Driver

```
public class Driver
extends Object
```

Constructor Summary

Constructors

Constructor	Description
`Driver()`	

Method Summary

All Methods Static Methods Concrete Methods

Modifier and Type	Method	Description
static void	`main(String [] args)`	

Methods inherited from class java.lang.Object

clone , equals , finalize , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait

Constructor Details

Driver

```
public Driver()
```

Method Details

main

```
public static void main(String [] args)
```

Package JavaContainerPackage

Interface JavaContainer<T>

All Known Implementing Classes:

JavaSet, JavaVector

---

public interface **JavaContainer<T>**

Method Summary

All Methods Instance Methods Abstract Methods

Modifier and Type	Method	Description
void	**add**(T element)	Adds an element to the set.
Iterator<T>	**getIterator**()	Returns an iterator over the elements in the Container.
void	**print**()	Prints the string representation of the set.
void	**remove**(T element)	Removes an element from the Container.
int	**size**()	Returns the number of elements in the Container.

Method Details

add

void add(T element)

Adds an element to the set.

Parameters:

element - the element to be added

Throws:

IllegalArgumentException - if the element is already in the set

remove

```
void remove(T element)
```

Removes an element from the Container.

Parameters:

`element` - the element to be removed

Throws:

`IllegalArgumentException` - if the element is not in the Container

size

```
int size()
```

Returns the number of elements in the Container.

Returns:

the number of elements in the Container.

getIterator

```
Iterator<T> getIterator()
```

Returns an iterator over the elements in the Container.

Returns:

an iterator over the elements in the Container

print

```
void print()
```

Prints the string representation of the set.

Package JavaContainerPackage

Class JavaSet<T>

java.lang.Object
    JavaContainerPackage.JavaSet<T>

Type Parameters:

T - the type of elements in the set

All Implemented Interfaces:

JavaContainer<T>

```
public class JavaSet<T>
extends Object
implements JavaContainer<T>
```

A generic set implementation.

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
void	**add**(T element)	Adds an element to the set.
boolean	**equals**(Object other)	Checks if this set is equal to another object.
Iterator<T>	**getIterator**()	Returns an iterator over the elements in the set.
void	**print**()	Prints the string representation of the set.
void	**remove**(T element)	Removes an element from the set.
int	**size**()	Returns the number of elements in the set.
String	**toString**()	Returns a string representation of the set.

Methods inherited from class java.lang.Object

clone , finalize , getClass , hashCode , notify , notifyAll , wait , wait , wait

Method Details

size

```
public int size()
```

Returns the number of elements in the set.

Specified by:

`size` in interface `JavaContainer<T>`

Returns:

the number of elements in the set

print

```
public void print()
```

Prints the string representation of the set.

Specified by:

`print` in interface `JavaContainer<T>`

toString

```
public String  toString()
```

Returns a string representation of the set.

Overrides:

`toString` in class `Object`

Returns:

a string representation of the set

equals

```
public boolean equals(Object  other)
```

Checks if this set is equal to another object.

Overrides:

`equals` in class `Object`

Parameters:

`other` - the object to compare with

Returns:

true if the sets are equal, false otherwise

add

```
public void add(T element)
```

Adds an element to the set.

Specified by:

`add` in interface `JavaContainer<T>`

Parameters:

`element` - the element to be added

Throws:

`IllegalArgumentException` - if the element is already in the set

remove

```
public void remove(T element)
```

Removes an element from the set.

Specified by:

`remove` in interface `JavaContainer<T>`

Parameters:

`element` - the element to be removed

Throws:

`IllegalArgumentException` - if the element is not in the set

getIterator

```
public Iterator<T> getIterator()
```

Returns an iterator over the elements in the set.

Specified by:

`getIterator` in interface `JavaContainer<T>`

Returns:

an iterator over the elements in the set

Package JavaContainerPackage

Class JavaVector<T>

java.lang.Object
    JavaContainerPackage.JavaVector<T>

Type Parameters:

T - the type of elements in the Vector

All Implemented Interfaces:

JavaContainer<T>

```
public class JavaVector<T>
extends Object
implements JavaContainer<T>
```

A generic vector implementation.

Constructor Summary

Constructors

Constructor	Description
JavaVector()	Constructs an empty vector with an initial capacity of 1.

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
void	add(T element)	Adds an element to the set.
boolean	equals(Object other)	Checks if this vector is equal to another object.
Iterator<T>	getIterator()	Returns an iterator over the elements in the set.
void	print()	Prints the string representation of the vector.
void	remove(T element)	Removes an element from the set.
int	size()	Returns the number of elements in the vector.
String	toString()	Returns a string representation of the set.

Methods inherited from class java.lang.Object

`clone` , `finalize` , `getClass` , `hashCode` , `notify` , `notifyAll` , `wait` , `wait` , `wait`

Constructor Details

JavaVector

```
public JavaVector()
```

Constructs an empty vector with an initial capacity of 1.

Method Details

size

```
public int size()
```

Returns the number of elements in the vector.

Specified by:

`size` in interface `JavaContainer<T>`

Returns:

the number of elements in the vector

print

```
public void print()
```

Prints the string representation of the vector.

Specified by:

`print` in interface `JavaContainer<T>`

toString

```
public String  toString()
```

Returns a string representation of the set.

Overrides:

`toString` in class `Object`

Returns:

a string representation of the set

equals

```
public boolean equals(Object  other)
```

Checks if this vector is equal to another object.

Overrides:

`equals` in class `Object`

Parameters:

`other` - the object to compare with

Returns:

true if the vectors are equal, false otherwise

add

```
public void add(T element)
```

Adds an element to the set.

Specified by:

`add` in interface `JavaContainer<T>`

Parameters:

`element` - the element to be added

Throws:

`IllegalArgumentException` - if the element is already in the set

remove

```
public void remove(T element)
```

Removes an element from the set.

Specified by:

`remove` in interface `JavaContainer<T>`

Parameters:

`element` - the element to be removed

Throws:

`IllegalArgumentException` - if the element is not in the set

getIterator

```
public Iterator<T> getIterator()
```

Returns an iterator over the elements in the set.

Specified by:

`getIterator` in interface `JavaContainer<T>`

Returns:

an iterator over the elements in the set