Class Set

java.lang.Object
    Set

```
public final class Set
extends java.lang.Object
```

Assumptions:

- All the elements of the Set must be Strings

Design:

- The Set can't contain more than 100 elements, this is because in Java the length of arrays is final so would I have to create new array every time I added or removed an element.
- We probably won't need more than 100 elements.
- If the Set reaches the 100th element and we try to add another element it will throw an exception that the Set is full.

Author:

Hishmat Salehi - MacID: Salehh6

Field Summary

Fields

Modifier and Type	Field and Description
`private java.lang.String[]`	`array` The array that holds the content of the set
`private int`	`index` The current index of the array; For adding new elements to the array

Constructor Summary

Constructors

Constructor and Description
`Set()` Constructor for when there is no arguments
`Set(java.lang.String[] arr, int size)` Constructor for when there is 2 arguments

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method and Description
Set	add(java.lang.String str) Method for adding an element to the Set
boolean	contains(java.lang.String str) Method for checking if a string is an element of the set.
Set	difference(Set S) Method that creates a new Set of difference of the current Set from the Set S
java.lang.String[]	getArray() Getter that returns the array of elements
int	getCount() Method that returns the size of the Set
int	getIndex() Getter that returns the current index of the array.
Set	intersection(Set S) Method that creates a new Set of intersection of the current Set and the Set S
boolean	isEqual(Set S) Method that checks if the current Set is equal to Set S
boolean	isSubset(Set S) Method that checks if the Set S is a subset of current Set
Set	product(Set S) Method that creates a new Set of Cartesian Product of the current Set with the Set S
Set	remove(java.lang.String str) Method for removing an element from the Set
java.lang.String	toString() Returns the string representation of the current Set.
Set	union(Set S) Method that creates a new Set of union of the current Set and the Set S

Methods inherited from class java.lang.Object

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

Field Detail

array

```
private final java.lang.String[] array
```

The array that holds the content of the set

index

```
private final int index
```

The current index of the array; For adding new elements to the array

Constructor Detail

Set

```
Set()
```

Constructor for when there is no arguments

Set

```
Set(java.lang.String[] arr,
    int size)
```

Constructor for when there is 2 arguments

Parameters:

`arr - Array of Strings`

`size - is the length of the array`

Method Detail

getArray

```
public java.lang.String[] getArray()
```

Getter that returns the array of elements

Returns:

`Array of elements of type String`

getIndex

```
public int getIndex()
```

Getter that returns the current index of the array.

Returns:

Current index of the array

add

```
public Set add(java.lang.String str)
        throws java.lang.Exception
```

Method for adding an element to the Set

Parameters:

str - is the element that you want to add to the Set

Returns:

A new Set with the added element

Throws:

An - Exception when the element is already in the Set

java.lang.Exception

remove

```
public Set remove(java.lang.String str)
           throws java.lang.Exception
```

Method for removing an element from the Set

Parameters:

str - is the element that you want to remove form the Set

Returns:

A new Set with the element removed

Throws:

An - Exception when the element is not in the Set

java.lang.Exception

contains

```
public boolean contains(java.lang.String str)
```

Method for checking if a string is an element of the set.

Parameters:

str - is the Element that you want to check if it's in the Set

Returns:

true if the element is in the Set, false otherwise

union

```
public Set union(Set S)
```

Method that creates a new Set of union of the current Set and the Set S

Parameters:

`S - is the Set that you want the union of with the current Set`

Returns:

`A new Set that contains all the elements of the current Set and Set S`

intersection

```
public Set intersection(Set S)
                 throws java.lang.Exception
```

Method that creates a new Set of intersection of the current Set and the Set S

Parameters:

`S - is the Set that you want the intersection of with the current Set`

Returns:

`A new Set that contains mutual elements between the current Set and Set S`

Throws:

`java.lang.Exception`

difference

```
public Set difference(Set S)
               throws java.lang.Exception
```

Method that creates a new Set of difference of the current Set from the Set S

Parameters:

`S - is the Set that you want the difference of with the current Set`

Returns:

`A new Set that contains all the elements of the current Set, except the ones that are in the intersection of Set S`

Throws:

`java.lang.Exception`

product

```
public Set product(Set S)
            throws java.lang.Exception
```

Method that creates a new Set of Cartesian Product of the current Set with the Set S

Parameters:

`S - is the Set that you want the Cartesian Product of with the current Set`

Returns:

`A new Set of all possible pairs of concatenated elements of the form rs, where r is in Set R and s is in Set S.`

Throws:

`An - Exception when the element is already in the Set`

`java.lang.Exception`

isEqual

```
public boolean isEqual(Set S)
```

Method that checks if the current Set is equal to Set S

Parameters:

`S - is the Set that you are comparing the current Set with`

Returns:

`true if the current S is equal to Set S, false otherwise`

isSubset

```
public boolean isSubset(Set S)
```

Method that checks if the Set S is a subset of current Set

Parameters:

`S - is the Set that you are comparing the current Set with`

Returns:

`true if the Set S is a subset of current Set, false otherwise`

getCount

```
public int getCount()
```

Method that returns the size of the Set

Returns:

`The size of the Set`

toString

```
public java.lang.String toString()
```

Returns the string representation of the current Set.

Overrides:

```
toString in class java.lang.Object
```

Returns:

The string representation with the format "{e_1, e_2, e_3, ... , e_n}"