Sets

- **Set is an unordered list of items**
  - Items are unique! Only one copy of each item in set!
- **In Java we will use TreeSet to manipulate a set**
  - A TreeSet is a particular implementation of a Set
- **Operations:**
  - Create a set
  - Add an item to a set
  - Check if item is in a set
  - Is set empty?
  - Remove item from set
Example – Create and add to Set

TreeSet<String> firstnames = new TreeSet<String>();

firstnames.add("John");
firstnames.add("Emily");
firstnames.add("Alex");
firstnames.add("Mike");
firstnames.add("John");
firstnames.add("Mike");
Example – Is object in set?

```java
if (firstnames.contains("Zed"))
    System.out.println("Zed is in the set.");
else
    System.out.println("Zed is not in the set.");
if (firstnames.contains("Mike"))
    System.out.println("Mike is in the set.");
else
    System.out.println("Mike is not in the set.");
```
Iterator – Look at each element in a Set

- Can create an iterator to look at each element in the set (as covered recently)
- Don’t know the order of the elements
- Guaranteed to give you all the elements in the set – one at a time
Iterate over elements in Set `firstnames`

- With collections loop, iterator is automatically created for you!

```java
// Print elements in set
for (String name: firstnames)
{
    System.out.println(name);
}
```
Alternative way to use Iterator

// you must create iterator for set
Iterator<String> iter2 = firstnames.iterator();
// use iterator to print elements in set
while (iter2.hasNext())
{
    System.out.println(iter2.next());
}
Example – Other Operations on Sets

- **size()** – returns size of set
  
  ```java
  System.out.println("Size of set is "+ firstnames.size());
  ```

- **remove(object)** – remove object from set if there

- **isEmpty()** – return true if set is empty

- See “Sets” and “Iterator” on Java API page
Output for Code shown
(Set only printed once)

Zed is not in the set.
Mike is in the set.
Alex
Emily
John
Mike
Size of set is 4
Set Operations

- **Union of two sets**
  - all the elements from both sets

- **Intersection of two sets**
  - the elements that are in both sets

- **Difference of two sets (A – B)**
  - the elements in A that are not in B
Classwork Today

- Implement set operations for two sets
  - Union, intersection, difference
- Implement set operations for array of sets
  - Union, intersection