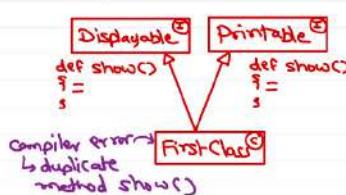
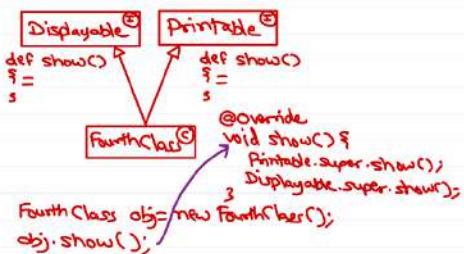
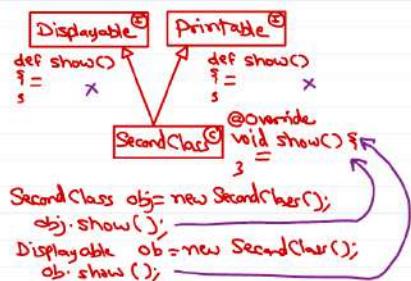
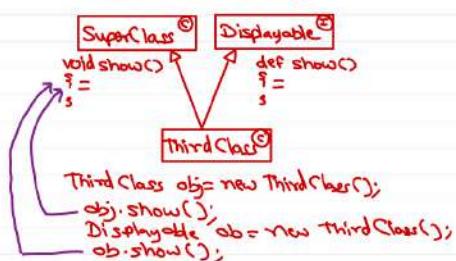


Default methods

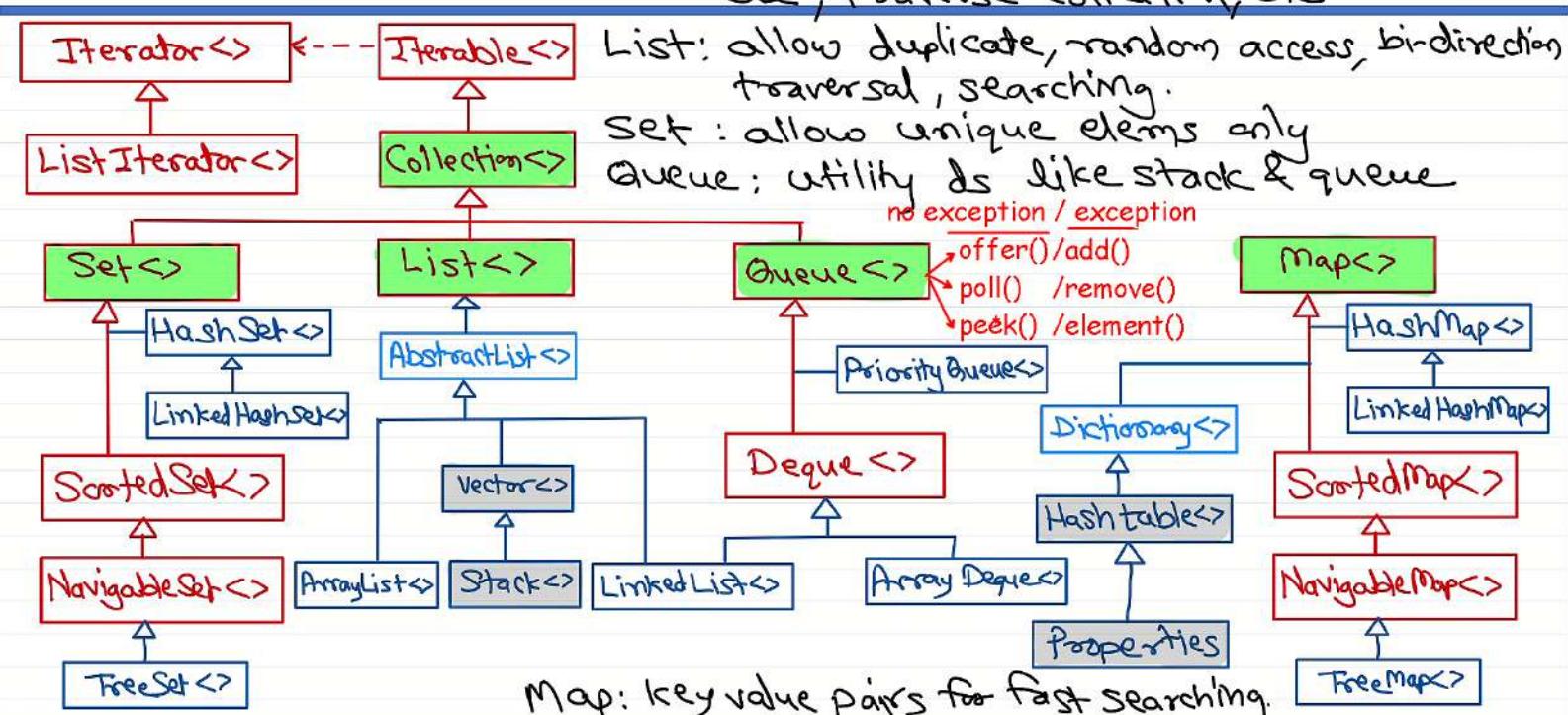
Super-interfaces clash!



Super-class wins!!



Java collection framework



The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** day14 - demo01/src/com/sunbeam/Program01.java - Spring Tool Suite 4
- Toolbar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Package Explorer:** demo01 [CI-H-02 main] > JRE System Library [JavaSE-1.8] > src > com.sunbeam > Program01.java
- Code Editor:** The code for Program01.java is displayed. It creates a LinkedList queue, adds four elements ("One", "Two", "Three", "Four"), prints the first element, and then repeatedly prints the popped element until the queue is empty. A comment explains that if peek() fails, it returns null. A red annotation points to the line `System.out.println("Popped from Empty Queue: " + q.poll()); // null` with the text "return null, because queue is empty".

```
4 import java.util.Queue;
5
6 public class Program01 {
7     public static void main(String[] args) {
8         Queue<String> q = new LinkedList<>();
9         q.offer("One");
10        q.offer("Two");
11        q.offer("Three");
12        q.offer("Four");
13        System.out.println("First Element: " + q.peek());           if operation fails, it returns null.
14        while(!q.isEmpty()) {
15            String ele = q.poll();
16            System.out.println("Popped: " + ele);
17        }
18        System.out.println("Popped from Empty Queue: " + q.poll()); // null
19    }
20 }
```

- Console:** The output window shows the execution of the program:

```
First Element: One
Popped: One
Popped: Two
Popped: Three
Popped: Four
Popped from Empty Queue: null
```
- Bottom Status Bar:** Writable, Smart Insert, 18 : 76 : 491, 9:26 AM

day14 - demo01/src/com/sunbeam/Program01.java - Spring Tool Suite 4

You are screen sharing Stop Share

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X

demo01 [CI-H-02 main]

JRE System Library [JavaSE-1.8]

src

com.sunbeam

Program01.java

```
24
25  public static void main(String[] args) {
26      //Queue<String> q = new LinkedList<>();
27      Queue<String> q = new ArrayDeque<>();
28      q.add("One");
29      q.add("Two");
30      q.add("Three");
31      q.add("Four");
32      System.out.println("First Element: " + q.element()); throws NoSuchElementException
33      while(!q.isEmpty()) {
34          String ele = q.remove();
35          System.out.println("Popped: " + ele);
36      }
37      System.out.println("Popped from Empty Queue: " + q.remove()); // null
38  }
```

If any operation fails, it throws exception.

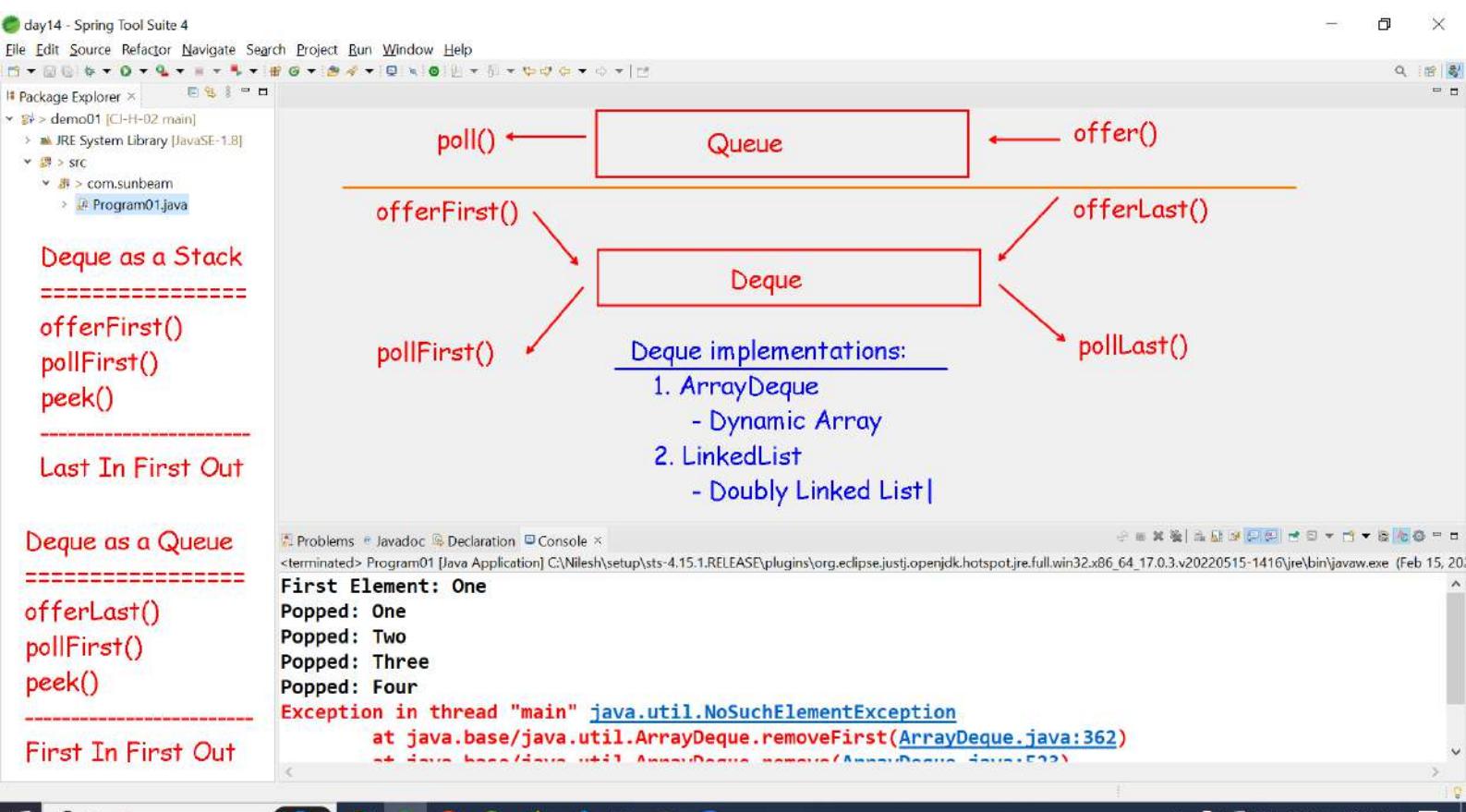
throws NoSuchElementException because queue is empty.

First Element: One
Popped: One
Popped: Two
Popped: Three
Popped: Four
Exception in thread "main" java.util.NoSuchElementException
at java.base/java.util.ArrayDeque.removeFirst(ArrayDeque.java:362)
at java.base/java.util.ArrayDeque.remove(ArrayDeque.java:523)

Syntax error on token "/", delete this token

Writable Smart Insert 37 : 9 : 940

Search



day14 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program02.java

```
5 import java.util.LinkedList;
6
7 public class Program02 {
8     public static void main(String[] args) {
9         // Deque as Stack
10        Deque<Integer> s = new ArrayDeque<>(); //new LinkedList<>();
11        s.offerFirst(11);
12        s.offerFirst(22);
13        s.offerFirst(33);
14        s.offerFirst(44);
15        while(!s.isEmpty()) {
16            Integer ele = s.pollFirst();
17            System.out.println("Popped: " + ele);
18        }
19    }
}
```

Problems Javadoc Declaration Console

<terminated> Program02 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 9:43:03 AM – 9:43:03 AM) [pid: 18556]

Popped: 44
Popped: 33
Popped: 22
Popped: 11

The word 'Deque' is not correctly spelled

Search Writable Smart Insert 9:26 [17] 9:43 AM

day14 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program02.java

```
19
20 }
21 */
22
23 public static void main(String[] args) {
24     // Deque as Queue
25     Deque<Integer> s = new ArrayDeque<>(); //new LinkedList<>();
26     s.offerLast(11);
27     s.offerLast(22);
28     s.offerLast(33);
29     s.offerLast(44);
30     while(!s.isEmpty()) {
31         Integer ele = s.pollFirst();
32         System.out.println("Popped: " + ele);
33     }
34 }
```

Problems Javadoc Declaration Console

Popped: 11
Popped: 22
Popped: 33
Popped: 44

The word 'Deque' is not correctly spelled

day14 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X Program03.java

```
5
6 public class Program03 {
7     public static void main(String[] args) {
8         // Elements are retrieved as per priority -- decided by Comparable (Natural Ordering)
9         Queue<String> q = new PriorityQueue<>();
10        q.offer("I");
11        q.offer("N");
12        q.offer("F");
13        q.offer("O");
14        q.offer("T");
15        q.offer("E");
16        q.offer("C");
17        q.offer("H");
18        while(!q.isEmpty()) {
19            String ele = q.poll();
20            System.out.print(ele + ", ");
21        }
22    }
}
```

Problems Javadoc Declaration Console X

<terminated> Program03 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64 17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2023)

C, E, F, H, I, N, O, T,

Writable Smart Insert 8 : 94 [85]

day14 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program03.java

```
26- public static void main(String[] args) {
27-     class StringDescComparator implements Comparator<String> {
28-         @Override
29-         public int compare(String x, String y) {
30-             return -x.compareTo(y);
31-         }
32-     }
33-     // Elements are retrieved as per priority - decided by given Comparator.
34-     Queue<String> q = new PriorityQueue<>(new StringDescComparator());
35-     q.offer("T");
36-     q.offer("E");
37-     q.offer("C");
38-     q.offer("H");
39-     while(!q.isEmpty()) {
40-         String ele = q.poll();
41-         System.out.print(ele + ", ");
42-     }
43- }
```

Problems Javadoc Declaration Console

<terminated> Program03 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64.17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 9:51:29 AM – 9:51:29 AM) [pid: 12816]

T, H, E, C,

Search Writable Smart Insert 34 : 73 [26] 9:51 AM

day14 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program03.java

```
26- public static void main(String[] args) {
27-     class StringDescComparator implements Comparator<String> {
28-         @Override
29-         public int compare(String x, String y) {
30-             return -x.compareTo(y);
31-         }
32-     }
33-     // Elements are retrieved as per priority -- decided by given Comparator.
34-     Queue<String> q = new PriorityQueue<>(new StringDescComparator());
35-     q.offer("T");
36-     q.offer("E");                                offer()/poll() -- time complexity = O(log n)
37-     q.offer("C");
38-     q.offer("H");                                Internally uses Heap data structure.
39-     while(!q.isEmpty()) {
40-         String ele = q.poll();
41-         System.out.print(ele + ", ");
42-     }
43-
```

Problems Javadoc Declaration Console

<terminated> Program03 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64.17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 9:51:29 AM – 9:51:29 AM) [pid: 12816]

T, H, E, C,

Search Writable Smart Insert 38 : 22 : 975

day14 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program02.java ×

```
9  public static void main(String[] args) {
10     // Deque as Stack
11     Deque<Integer> s = new ArrayDeque<>(); //new LinkedList<>();
12     s.offerFirst(11);
13     s.offerFirst(22);
14     s.offerFirst(33);
15     s.offerFirst(44);
16     while(!s.isEmpty()) {
17         Integer ele = s.pollFirst();
18         System.out.println("Popped: " + ele);
19     }
20 }
21 */
22
23 public static void main(String[] args) {
24     // Deque as Queue
25     Deque<Integer> s = new ArrayDeque<>(); //new LinkedList<>();
26     s.offerLast(11);
27     s.offerLast(22);
28     s.offerLast(33);
29     s.offerLast(44);
30     while(!s.isEmpty()) {
31         Integer ele = s.pollFirst();
32         System.out.println("Popped: " + ele);
33     }
34 }
```

Time Complexity:

- add element -- O(1)
- remove element -- O(1)

Writable Smart Insert 1:1:0

Search

9:54 AM

day14 - demo04/src/com/sunbeam/Program04.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program04.java

```
5 import java.util.LinkedHashSet;
6 import java.util.Set;
7 import java.util.TreeSet;
8
9 public class Program04 {
10    public static void main(String[] args) {
11        //Set<String> set = new HashSet<>();
12        //Set<String> set = new LinkedHashSet<>();
13        Set<String> set = new TreeSet<>(); Stores elements in
14                                         sorted order (Natural order).
15        set.add("India"); // return true
16        set.add("Japan"); // return true
17        set.add("India"); // return false - already exists
18        set.add("Germany"); // return true
19        set.add("Africa"); // return true
20        set.add("India"); // return false - already exists
21        set.add("USA"); // return true
22        set.add("Japan"); // return false - already exists
23
24        System.out.println("Size: " + set.size());
25
26        for (String str : set)
27            System.out.println(str);
28    }
29 }
30
```

Problems Javadoc Declaration Console

<terminated> Program04 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE

Size: 5
Africa
Germany
India
Japan
USA

This screenshot shows the Spring Tool Suite (STS) interface. On the left, the code editor displays 'Program04.java' with code demonstrating the use of different Set implementations (HashSet, LinkedHashSet, TreeSet). A red box highlights the line 'Set<String> set = new TreeSet<>();'. A red annotation 'Stores elements in sorted order (Natural order).' is placed next to it. On the right, the 'Console' tab shows the execution output. It prints 'Size: 5' followed by the elements 'Africa', 'Germany', 'India', 'Japan', and 'USA', each on a new line. A red arrow points from the annotation to the 'India' entry in the console output.

day14 - demo04/src/com/sunbeam/Program04.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program04.java

```
5 import java.util.LinkedHashSet;
6 import java.util.Set;
7 import java.util.TreeSet;
8
9 public class Program04 {
10    public static void main(String[] args) {
11        //Set<String> set = new HashSet<>();
12        Set<String> set = new LinkedHashSet<>(); Elements stored in order
13        //Set<String> set = new TreeSet<>();          of Insertion.
14
15        set.add("India"); // return true
16        set.add("Japan"); // return true
17        set.add("India"); // return false - already exists
18        set.add("Germany"); // return true
19        set.add("Africa"); // return true
20        set.add("India"); // return false - already exists
21        set.add("USA"); // return true
22        set.add("Japan"); // return false - already exists
23
24        System.out.println("Size: " + set.size());
25
26        for (String str : set)
27            System.out.println(str);
28    }
29 }
30
```

Problems Javadoc Declaration Console

<terminated> Program04 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE

Size: 5
India
Japan
Germany
Africa
USA

10:02 AM

day14 - demo04/src/com/sunbeam/Program04.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program04.java

```
5 import java.util.LinkedHashSet;
6 import java.util.Set;
7 import java.util.TreeSet;
8
9 public class Program04 {
10    public static void main(String[] args) {
11        Set<String> set = new HashSet<>(); Elements are stored in
12        //Set<String> set = new LinkedHashSet<>(); arbitrary order - based on
13        //Set<String> set = new TreeSet<>(); hashcode of each element.
14
15        set.add("India"); // return true
16        set.add("Japan"); // return true
17        set.add("India"); // return false - already exists
18        set.add("Germany"); // return true
19        set.add("Africa"); // return true
20        set.add("India"); // return false - already exists
21        set.add("USA"); // return true
22        set.add("Japan"); // return false - already exists
23
24        System.out.println("Size: " + set.size());
25
26        for (String str : set)
27            System.out.println(str);
28    }
29 }
30
```

Problems Javadoc Declaration Console

<terminated> Program04 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE

Size: 5

USA
Japan
Germany
Africa
India

10:03 AM

day14 - demo05/src/com/sunbeam/Program05.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program05.java Book.java

```
1 package com.sunbeam;
2
3 import java.util.HashSet;
4 import java.util.Set;
5
6 public class Program05 {
7     public static void main(String[] args) {
8         Set<Book> set = new HashSet<>();
9         set.add(new Book(4, "The Alchemist", "Novel", 493.23));
10        set.add(new Book(1, "The Archer", "Novel", 723.53));
11        set.add(new Book(5, "The Fountainhead", "Novel", 652.73));
12        set.add(new Book(2, "Atlas Shrugged", "Novel", 872.94));
13        set.add(new Book(6, "Harry Potter", "Novel", 423.68));
14        set.add(new Book(1, "The Archer", "Novel", 723.53));
15        set.add(new Book(3, "Lord of Rings", "Novel", 621.53));
16        System.out.println("Set Size: " + set.size());
17        for (Book b : set)
18            System.out.println(b);
19    }
20 }
21
```

NOTE: HashSet and LinkedHashSet considers elements equal if and only if their hashCode() is same AND calling equals() to compare them returns true.

Problems Javadoc Declaration Console

<terminated> Program05 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org

Set Size: 7

Book [id=1, name=The Archer, subject=Novel, pr
Book [id=5, name=The Fountainhead, subject=Nov
Book [id=1, name=The Archer, subject=Novel, pr
Book [id=6, name=Harry Potter, subject=Novel,
Book [id=2, name=Atlas Shrugged, subject=Novel
Book [id=4, name=The Alchemist, subject=Novel,
Book [id=3, name=Lord of Rings, subject=Novel,

Elements are duplicated in HashSet,
even if equals() is overridden in Book class.
Because HashSet doesn't compare elements
only on basis of equals().



day14 - demo05/src/com/sunbeam/Program05.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program05.java Book.java

```
1 package com.sunbeam;
2
3 import java.util.HashSet;
4 import java.util.Set;
5
6 public class Program05 {
7     public static void main(String[] args) {
8         Set<Book> set = new HashSet<>();
9         set.add(new Book(4, "The Alchemist", "Novel", 493.23));
10        set.add(new Book(1, "The Archer", "Novel", 723.53));
11        set.add(new Book(5, "The Fountainhead", "Novel", 652.73));
12        set.add(new Book(2, "Atlas Shrugged", "Novel", 872.94));
13        set.add(new Book(6, "Harry Potter", "Novel", 423.68));
14        set.add(new Book(1, "The Archer", "Novel", 723.53));
15        set.add(new Book(3, "Lord of Rings", "Novel", 621.53));
16        System.out.println("Set Size: " + set.size());
17        for (Book b : set)
18            System.out.println(b);
19    }
20 }
21
```

Problems Javadoc Declaration Console

<terminated> Program05 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org

Set Size: 6

Book [id=1, name=The Archer, subject=Novel, pr
Book [id=2, name=Atlas Shrugged, subject=Novel
Book [id=3, name=Lord of Rings, subject=Novel,
Book [id=4, name=The Alchemist, subject=Novel,
Book [id=5, name=The Fountainhead, subject=Nov
Book [id=6, name=Harry Potter, subject=Novel,

Duplicates are removed.

Book class implemented both hashCode() and equals(). -- implemented on "id" field.

day14 - demo05/src/com/sunbeam/Program05.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program05.java Book.java

```
25  */
26
27  public static void main(String[] args) {
28      Set<Book> set = new TreeSet<>();
29
30      set.add(new Book(4, "The Alchemist", "Novel", 493.23));
31      set.add(new Book(1, "The Archer", "Novel", 723.53));
32      set.add(new Book(5, "The Fountainhead", "Novel", 652.73));
33      set.add(new Book(2, "Atlas Shrugged", "Novel", 872.94));
34      set.add(new Book(6, "Harry Potter", "Novel", 423.68));
35      set.add(new Book(1, "The Archer", "Novel", 723.53));
36      set.add(new Book(3, "Lord of Rings", "Novel", 621.53));
37      System.out.println("Set Size: " + set.size());
38      for (Book b : set)
39          System.out.println(b);
40  }
41 }
42 }
```

To store elements in a TreeSet, they must be Comparable i.e. must have natural ordering.*

*NOTE: TreeSet param-less constructor is used.

Problems Javadoc Declaration Console

<terminated> Program05 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.justmyjava\hotspot\jre\full\win32\x86_64\17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 10:25:26 AM - 10:25:26 AM) [pid: 12345]

Exception in thread "main" java.lang.ClassCastException: class com.sunbeam.Book cannot be cast to class java.lang.Comparable (

at java.base/java.util.TreeMap.compare(TreeMap.java:1569)
at java.base/java.util.TreeMap.addEntryToEmptyMap(TreeMap.java:776)
at java.base/java.util.TreeMap.put(TreeMap.java:785)
at java.base/java.util.TreeMap.put(TreeMap.java:534)

day14 - demo05/src/com/sunbeam/Program05.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program05.java Book.java

```
17     set.add(new Book(2, "Atlas Shrugged", "Novel", 872.94));
18     set.add(new Book(6, "Harry Potter", "Novel", 423.68));
19     set.add(new Book(1, "The Archer", "Novel", 723.53));
20     set.add(new Book(3, "Lord of Rings", "Novel", 621.53));
21     System.out.println("Set Size: " + set.size());
22     for (Book b : set)
23         System.out.println(b);
24 }
25 */
26
27 public static void main(String[] args) {
28     Set<Book> set = new TreeSet<>();
29
30     set.add(new Book(4, "The Alchemist", "Novel", 493.23));
31     set.add(new Book(1, "The Archer", "Novel", 723.53));
32     set.add(new Book(5, "The Fountainhead", "Novel", 652.73));
33     set.add(new Book(2, "Atlas Shrugged", "Novel", 872.94));
34     set.add(new Book(6, "Harry Potter", "Novel", 423.68));
35     set.add(new Book(1, "The Archer", "Novel", 723.53));
36     set.add(new Book(3, "Lord of Rings", "Novel", 621.53));
37     System.out.println("Set Size: " + set.size());
38     for (Book b : set)
39         System.out.println(b);
40 }
41 }
42
```

Problems Javadoc Declaration Console

<terminated> Program05 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.core\src\com\sunbeam\Program05.java

Set Size: 6

Book [id=1, name=The Archer, subject=Novel, price=723.53]
Book [id=2, name=Atlas Shrugged, subject=Novel, price=872.94]
Book [id=3, name=Lord of Rings, subject=Novel, price=621.53]
Book [id=4, name=The Alchemist, subject=Novel, price=493.23]
Book [id=5, name=The Fountainhead, subject=Novel, price=652.73]
Book [id=6, name=Harry Potter, subject=Novel, price=423.68]

Duplicates removed in TreeSet as per natural ordering of the elements.

The Book class implemented Comparable and compared on basis of "id".

day14 - demo05/src/com/sunbeam/Program05.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program05.java Book.java

```
46+     class BookPriceComparator implements Comparator<Book> {
47+         @Override
48+         public int compare(Book x, Book y) {
49+             return Double.compare(x.getPrice(), y.getPrice());
50+         }
51+     }
52+     // stores Books in sorted order -- as per Comparator (on price)
53+     Set<Book> set = new TreeSet<>(new BookPriceComparator());
54+     set.add(new Book(4, "The Alchemist", "Novel", 493.23));
55+     set.add(new Book(1, "The Archer", "Novel", 723.53));
56+     set.add(new Book(5, "The Fountainhead", "Novel", 652.73));
57+     set.add(new Book(2, "Atlas Shrugged", "Novel", 872.94));
58+     set.add(new Book(6, "Harry Potter", "Novel", 621.53));
59+     set.add(new Book(1, "The Archer", "Novel", 723.53));
60+     set.add(new Book(3, "Lord of Rings", "Novel", 621.53));
61+     System.out.println("Set Size: " + set.size());
62+     for (Book b : set)
```

Problems Javadoc Declaration Console

<terminated> Program05 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 10:34:15 AM – 10:34:16 AM) [pid: 1750]

Set Size: 5

Book [id=4, name=The Alchemist, subject=Novel, price=493.23]
Book [id=6, name=Harry Potter, subject=Novel, price=621.53]
Book [id=5, name=The Fountainhead, subject=Novel, price=652.73]
~~Book [id=1, name=The Archer, subject=Novel, price=723.53]~~
Book [id=2, name=Atlas Shrugged, subject=Novel, price=872.94]

Stored in sorted order of price -- given by Comparator.
If price is same, element is removed (duplication).

You are screen sharing

Stop Share

String (Java Platform SE 8)

https://docs.oracle.com/javase/8/docs/api/java/lang/String.html#hashCode--

133%

hashCode

```
public int hashCode()
```

Returns a hash code for this string. The hash code for a String object is computed as

```
s[0]*31^(n-1) + s[1]*31^(n-2) + ... + s[n-1]
```

using int arithmetic, where $s[i]$ is the i th character of the string, n is the length of the string, and \wedge indicates exponentiation. (The hash value of the empty string is zero.)

Overrides:
hashCode in class Object

Returns:
a hash code value for this object.

See Also:
`Object.equals(java.lang.Object)`, `System.identityHashCode(java.lang.Object)`

day14 - demo08/src/com/sunbeam/Program08.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program08.java

```
10 public class Program08 {  
11     public static void main(String[] args) {  
12         Map<Integer, String> map = new HashMap<>();  
13         map.put(415110, "Karad - Satara"); // retruns -- null  
14         map.put(411052, "Hinjawadi - Pune"); // retruns -- null  
15         map.put(411046, "Katraj - Pune"); // retruns -- null  
16         map.put(400027, "Byculla - Mumbai"); // retruns -- null  
17         map.put(411002, "Bajirao Rd - Pune"); // retruns -- null  
18         map.put(411037, "Marketyard - Pune"); // retruns -- null  
19         map.put(411007, "Aundh - Pune"); // retruns -- null  
20         map.put(411052, "HINJEWADI - Pune"); // when key is duplicate, value is overwritten  
21             // returns -- old value for the key -- "Hinjawadi - Pune"  
22     }  
}
```

HashMap stores entries in arbitrary order.
Depends on hash code of key.

Problems Javadoc Declaration Console

<terminated> Program08 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 12:57:31 PM – 12:57:31 PM) [pid: 8520]

Keys (Pins): [415110, 411046, 411007, 411052, 411037, 411002, 400027]
Values (Areas): [Karad - Satara, Katraj - Pune, Aundh - Pune, HINJEWADI - Pune, Marketyard - Pune, Bajirao Rd - Pune, Byculla - Mumbai]

415110 --> Karad - Satara
411046 --> Katraj - Pune
411007 --> Aundh - Pune
411052 --> HINJEWADI - Pune
411037 --> Marketyard - Pune
411002 --> Bajirao Rd - Pune
400027 --> Byculla - Mumbai

Writable Smart Insert 46 : 1 : 1658

day14 - demo08/src/com/sunbeam/Program08.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program08.java

```
11 public class Program08 {
12     public static void main(String[] args) {
13         //Map<Integer, String> map = new HashMap<>();
14         Map<Integer, String> map = new LinkedHashMap<>();
15         map.put(415110, "Karad - Satara"); // retruns -- null
16         map.put(411052, "Hinjawadi - Pune"); // retruns -- null
17         map.put(411046, "Katraj - Pune"); // retruns -- null
18         map.put(400027, "Byculla - Mumbai"); // retruns -- null
19         map.put(411002, "Bajirao Rd - Pune"); // retruns -- null
20         map.put(411037, "Marketyard - Pune"); // retruns -- null
21         map.put(411007, "Aundh - Pune"); // retruns -- null
22         map.put(411052, "HINJEWADI - Pune"); // when key is duplicate, value is overwritten
23             // returns -- old value for the key -- "Hinjawadi - Pune"
```

LinkedHashMap stores elements in the order of insertion of keys.

Problems Javadoc Declaration Console <terminated> Program08 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 12:58:50 PM – 12:58:51 PM) [pid: 13540]

Keys (Pins): [415110, 411052, 411046, 400027, 411002, 411037, 411007]

Values (Areas): [Karad - Satara, HINJEWADI - Pune, Katraj - Pune, Byculla - Mumbai, Bajirao Rd - Pune, Marketyard - Pune, Aundh - Pune]

415110 --> Karad - Satara
411052 --> HINJEWADI - Pune
411046 --> Katraj - Pune
400027 --> Byculla - Mumbai
411002 --> Bajirao Rd - Pune
411037 --> Marketyard - Pune
411007 --> Aundh - Pune

day14 - demo08/src/com/sunbeam/Program08.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program08.java

```
12 public class Program08 {  
13     public static void main(String[] args) {  
14         //Map<Integer, String> map = new HashMap<>();  
15         //Map<Integer, String> map = new LinkedHashMap<>();  
16         Map<Integer, String> map = new TreeMap<>();  
17         map.put(415110, "Karad - Satara"); // retruns -- null  
18         map.put(411052, "Hinjawadi - Pune"); // retruns -- null  
19         map.put(411046, "Katraj - Pune"); // retruns -- null  
20         map.put(400027, "Byculla - Mumbai"); // retruns -- null  
21         map.put(411002, "Bajirao Rd - Pune"); // retruns -- null  
22         map.put(411037, "Marketyard - Pune"); // retruns -- null  
23         map.put(411007, "Aundh - Pune"); // retruns -- null  
24         map.put(411052, "HINJEWADI - Pune"); // when key is duplicate, value is overwritten
```

TreeMap stores entries in natural ordering of keys (sorted).

NOTE: TreeMap() param less ctor is used.

Problems Javadoc Declaration Console

<terminated> Program08 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Feb 15, 2024, 1:00:04 PM – 1:00:04 PM) [pid: 21328]

Keys (Pins): [400027, 411002, 411007, 411037, 411046, 411052, 415110]
Values (Areas): [Byculla - Mumbai, Bajirao Rd - Pune, Aundh - Pune, Marketyard - Pune, Katraj - Pune, HINJEWADI - Pune, Karad - Satara]

400027 --> Byculla - Mumbai
411002 --> Bajirao Rd - Pune
411007 --> Aundh - Pune
411037 --> Marketyard - Pune
411046 --> Katraj - Pune
411052 --> HINJEWADI - Pune
415110 --> Karad - Satara

day15 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer X | Problems X | Javadoc X | Declaration X | Console X

demo01 [C:\H-02 main] <terminated> Program02 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE

JRE System Library [JavaSE-1.8]

src

com.sunbeam

Circle.java

Program01.java

Rectangle.java

Shape.java

Square.java

demo02 [C:\H-02 main]

JRE System Library [JavaSE-1.8]

src

com.sunbeam

Program02.java

Program02.java

```
1 package com.sunbeam;
2
3 interface Printable {
4     void show();
5 }
6 interface Displayable {
7     void show();
8 }
9 class MyClass implements Printable, Displayable {
10    @Override
11    public void show() {
12        System.out.println("MyClass.show() called.");
13    }
14 }
15
16 public class Program02 {
17    public static void main(String[] args) {
18        MyClass obj1 = new MyClass();
19        obj1.show();
20        Printable obj2 = new MyClass();
21        obj2.show();
22        Displayable obj3 = new MyClass();
23        obj3.show();
24    }
25 }
26
```

MyClass.show() called.
MyClass.show() called.
MyClass.show() called.

Search 10:18 AM

day15 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

```
Program02.java x
56 interface Printable {
57     default void show() {
58         System.out.println("Printable.show() called.");
59     }
60 }
61 interface Displayable {
62     default void show() {
63         System.out.println("Displayable.show() called.");
64     }
65 }
66 // if two interfaces have default method with same signature and a class is inher:
67 // then it will lead to ambiguity.
68 // this problem can be resolved by overriding method in sub-class.
69 class SecondClass implements Printable, Displayable {
70     public void show() {
71         System.out.println("SecondClass.show() called.");
72     }
73 }
74 public class Program02 {
75     public static void main(String[] args) {
76         SecondClass obj1 = new SecondClass();
77         obj1.show();
78         Printable obj2 = new SecondClass();
79         obj2.show();
80         Displayable obj3 = new SecondClass();
81         obj3.show();
}
Problems Javadoc Declaration Console x
<terminated> Program02 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELE
SecondClass.show() called.
SecondClass.show() called.
SecondClass.show() called.
```

day15 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

```
File Edit Source Refactor Navigate Search Project Run Window Help
```

Program02.java ×

```
87 class Printable {
88     public void show() {
89         System.out.println("Printable.show() called.");
90     }
91 }
92 interface Displayable {
93     default void show() {
94         System.out.println("Displayable.show() called.");
95     }
96 }
97 // when same signature method is inherited from a super-class
98 // and a super-interface, the super-class method gets precedence.
99 // no compiler error for ambiguity
100 class ThirdClass extends Printable implements Displayable {
101 }
102
103 public class Program02 {
104     public static void main(String[] args) {
105         ThirdClass obj1 = new ThirdClass();
106         obj1.show();
107         Printable obj2 = new ThirdClass();
108         obj2.show();
109         Displayable obj3 = new ThirdClass();
110         obj3.show();
111     }
112 }
```

super-class wins! super-interfaces clash!!

Problems Javadoc Declaration Console

<terminated> Program02 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE
Printable.show() called.
Printable.show() called.
Printable.show() called.

Search 10:28 AM

day15 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program02.java ×

```
114
115 class Printable {
116     public void show() {
117         System.out.println("Printable.show() called.");
118     }
119 }
120 interface Displayable {
121     default void show() {
122         System.out.println("Displayable.show() called.");
123     }
124 }
125 // method overriding -- method is called depending on type of object.
126 class FourthClass extends Printable implements Displayable {
127     public void show() {
128         System.out.println("FourthClass.show() called.");
129     }
130 }
131 public class Program02 {
132     public static void main(String[] args) {
133         FourthClass obj1 = new FourthClass();
134         obj1.show();
135         Printable obj2 = new FourthClass();
136         obj2.show();
137         Displayable obj3 = new FourthClass();
138         obj3.show();
139     }
}
```

Problems Javadoc Declaration Console

<terminated> Program02 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE
FourthClass.show() called.
FourthClass.show() called.
FourthClass.show() called.

Search 10:31 AM

day15 - demo02/src/com/sunbeam/Program02.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program02.java ×

```
143 interface Printable {
144     default void show() {
145         System.out.println("Printable.show() called.");
146     }
147 }
148 interface Displayable {
149     default void show() {
150         System.out.println("Displayable.show() called.");
151     }
152 }
153 // method overriding -- method is called depending on type of object.
154 class FourthClass implements Printable, Displayable {
155     public void show() {
156         System.out.println("FourthClass.show() called.");
157         // default methods in super interface can be called from sub-class methods
158         //super.show(); // error: show() is not found in super-class i.e. Object
159         Printable.super.show();
160         Displayable.super.show();
161     }
162 }
163 public class Program02 {
164     public static void main(String[] args) {
165         FourthClass obj1 = new FourthClass();
166         obj1.show();
167     }
168 }
```

Problems Javadoc Declaration Console

<terminated> Program02 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\Program02.jar

FourthClass.show() called.
Printable.show() called.
Displayable.show() called.

Search 10:38 AM

```
day15.md x day14.md classwork.md U
day15.md > # Core Java > ## Java 8 Interfaces > ##### Functional Interface

180 ##### Functional Interface
181 * If interface contains exactly one abstract method (SAM), it is said to be functional
  interface.
182 * It may contain additional default & static methods. E.g. Comparator, Runnable, ...
183 * @FunctionalInterface annotation does compile time check, whether interface contains
  single abstract method. If not, raise compile time error.

184     ```Java
185         @FunctionalInterface // okay
186         interface Foo {
187             void foo(); // SAM
188         }
189         ```
190     ```java
191         @FunctionalInterface // okay
192         interface FooBar1 {
193             void foo(); // SAM
194             default void bar() {
195                 /*... */
196             }
197         }
198     ```



pre-defined functional interface:



1. Comparable
2. Comparator
3. Runnable
4. Closeable/AutoCloseable



java.util.function package -- since Java 8



- Predicate, Function, BinaryOperator,
- Supplier, Consumer, ...

```

day15 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Program03.java Employee.java

```
public static void main(String[] args) {
    Employee[] arr = new Employee[] {
        new Employee(4, "B", "Clerk", "Sales", 723.44),
        new Employee(8, "X", "Manager", "Accounts", 823.23),
        new Employee(2, "P", "Clerk", "Research", 234.23),
        new Employee(9, "N", "Manger", "Sales", 252.53),
        new Employee(5, "D", "Clerk", "Accounts", 923.23),
        new Employee(1, "Q", "Analyst", "Research", 826.23),
        new Employee(7, "H", "Clerk", "Research", 845.24),
        new Employee(6, "A", "Analyst", "Research", 832.23),
        new Employee(3, "G", "Analyst", "Sales", 952.44)
    };

    System.out.println("Emps sorted by id -- using EmpIdComparator -- local class");
    class EmpIdComparator implements Comparator<Employee> {
        @Override
        public int compare(Employee x, Employee y) {
            int diff = x.getId() - y.getId(); //Integer.compare(x.getId(), y.getId());
            return diff;
        }
    }
    Arrays.sort(arr, new EmpIdComparator());
    for (Employee e : arr)
        System.out.println(e);
}
```

Problems Javadoc Declaration Console

<terminated> Program03 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE

Emps sorted by id -- using EmpIdComparator

Employee [id=1, name=Q, job=Analyst, department=Sales, salary=723.44]
Employee [id=2, name=P, job=Clerk, department=Research, salary=234.23]
Employee [id=3, name=G, job=Analyst, department=Accounts, salary=952.44]
Employee [id=4, name=B, job=Clerk, department=Sales, salary=823.23]
Employee [id=5, name=D, job=Clerk, department=Research, salary=252.53]
Employee [id=6, name=A, job=Analyst, department=Research, salary=832.23]
Employee [id=7, name=H, job=Clerk, department=Manager, salary=845.24]
Employee [id=8, name=X, job=Manager, department=Accounts, salary=826.23]
Employee [id=9, name=N, job=Manger, department=Sales, salary=252.53]

day15 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4

You are screen sharing Stop Share

```
File Edit Source Refactor Navigate Search Project Run Window Help
```

```
# Program03.java X Employee.java
21-     System.out.println("Emps sorted by id -- using EmpIdComparator -- local class");
22-     class EmpIdComparator implements Comparator<Employee> {
23-         @Override
24-         public int compare(Employee x, Employee y) {
25-             int diff = x.getId() - y.getId(); //Integer.compare(x.getId(), y.getId());
26-             return diff;
27-         }
28-     }
29-     Arrays.sort(arr, new EmpIdComparator());
30-     for (Employee e : arr)
31-         System.out.println(e);

32-     System.out.println("\nEmps sorted by name -- using Anonymous Inner class");
33-     Comparator<Employee> empNameComparator = new Comparator<Employee>() {
34-         @Override
35-         public int compare(Employee x, Employee y) {
36-             int diff = x.getName().compareTo(y.getName());
37-             return diff;
38-         }
39-     };
40-     Arrays.sort(arr, empNameComparator);
41-     for (Employee e : arr)
42-         System.out.println(e);
43-     }
44- }
```

impl of \$1 (anon) cls

new Classname() {
 // ...
};

Comparator<Emp>
\$1

\$1

Writable Smart Insert 32 : 31 : 1264

Search

11:32 AM

You are screen sharing

File Edit Source Refactor Navigate Search Project Run Window Help

Program03.java Employee.java

```

31 System.out.println("\nEmps sorted by name -- using Anonymous class");
32 Comparator<Employee> empNameComparator = new Comparator<Employee>() {
33     @Override
34     public int compare(Employee x, Employee y) {
35         int diff = x.getName().compareTo(y.getName());
36         return diff;
37     }
38 };
39 Arrays.sort(arr, empNameComparator);
40 for (Employee e : arr)
41     System.out.println(e);
42
43 System.out.println("\nEmps sorted by job -- using Anonymous Inner class Anonymous object");
44 Arrays.sort(arr, new Comparator<Employee>() {
45     @Override
46     public int compare(Employee x, Employee y) {
47         int diff = x.getJob().compareTo(y.getJob());
48         return diff;
49     }
50 });
51 for (Employee e : arr)
52     System.out.println(e);
53
54 }

```

Emps sorted by job -- using Anonymous Inner class Anonymous object

Employee [id=6, name=A, job=Analyst, dept=Research, salary=100000]
Employee [id=3, name=G, job=Analyst, dept=Sales, salary=100000]
Employee [id=1, name=Q, job=Analyst, dept=Research, salary=100000]
Employee [id=4, name=B, job=Clerk, dept=Sales, salary=72000]
Employee [id=5, name=D, job=Clerk, dept=Accounts, salary=72000]
Employee [id=7, name=H, job=Clerk, dept=Research, salary=72000]
Employee [id=2, name=P, job=Clerk, dept=Research, salary=72000]
Employee [id=8, name=X, job=Manager, dept=Accounts, salary=120000]
Employee [id=9, name=N, job=Manger, dept=Sales, salary=120000]

Anonymous object of Anonymous class passed as 2nd arg in sort() method.

Comparator<Employee>

\$2

```

graph TD
    $2["$2"] --> Comparator[Comparator<Employee>]

```

You are screen sharing

Stop Share

File Edit Source Refactor Navigate Search Project Run Window Help

Program03.java Employee.java

```
43 System.out.println("\nEmps sorted by job -- using Anonymous class");
44 Arrays.sort(arr, new Comparator<Employee>() {
45     @Override
46     public int compare(Employee x, Employee y) {
47         int diff = x.getJob().compareTo(y.getJob());
48         return diff;
49     }
50 });
51 for (Employee e : arr)
52     System.out.println(e);
53
54 System.out.println("\nEmps sorted by job in desc order: ");
55 Arrays.sort(arr, (Employee x, Employee y) -> {
56     int diff = -x.getJob().compareTo(y.getJob());
57     return diff;
58 });
59 for (Employee e : arr)
60     System.out.println(e);
61
62 }
63 }
64
65
66
67
68
```

Emps sorted by job in desc order:

Employee [id=9, name=N, job=Manger, dept=Sales, sal=200000]
Employee [id=8, name=X, job=Manager, dept=Accounts, sal=150000]
Employee [id=4, name=B, job=Clerk, dept=Sales, sal=72000]
Employee [id=5, name=D, job=Clerk, dept=Accounts, sal=60000]
Employee [id=7, name=H, job=Clerk, dept=Research, sal=50000]
Employee [id=2, name=P, job=Clerk, dept=Research, sal=45000]
Employee [id=6, name=A, job=Analyst, dept=Research, sal=80000]
Employee [id=3, name=G, job=Analyst, dept=Sales, sal=90000]
Employee [id=1, name=Q, job=Analyst, dept=Research, sal=75000]

Lambda expression is short-hand implementation of the abstract method in the functional interface.

You are screen sharing Stop Share

```
day15 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4
File Edit Source Refactor Navigate Search Project Run Window Help
Program03.java Employee.java
69
70     });
71     for (Employee e : arr)
72         System.out.println(e);
73
74     System.out.println("\nEmps sorted by sal in asc order: ");
75     Arrays.sort(arr, (x, y) -> {
76         return Double.compare(x.getSal(), y.getSal());
77     });
78     for (Employee e : arr)
79         System.out.println(e);
80
81     System.out.println("\nEmps sorted by sal in desc order: ");
82     // single liner lambda expression doesn't need curly braces
83     // and the result of expression is considered to be returned.
84     Arrays.sort(arr, (x, y) -> - Double.compare(x.getSal(), y.getSal()));
85     for (Employee e : arr)
86         System.out.println(e);
87 }
88
89
90
91
92
93
94 }
```

single abstract method → arguments + one-liner implementation of the functional interface

id=3, name=G, job=Analyst, dept=Sales, sal=952.44]
id=5, name=D, job=Clerk, dept=Accounts, sal=923.23]
id=7, name=H, job=Clerk, dept=Research, sal=845.24]
id=6, name=A, job=Analyst, dept=Research, sal=832.23]
id=1, name=Q, job=Analyst, dept=Research, sal=826.23]
id=8, name=X, job=Manager, dept=Accounts, sal=823.23]
id=4, name=B, job=Clerk, dept=Sales, sal=723.44]
id=9, name=N, job=Manger, dept=Sales, sal=252.53]
id=2, name=P, job=Clerk, dept=Research, sal=234.23]

day15 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4

You are screen sharing Stop Share

File Edit Source Refactor Navigate Search Project Run Window Help

Program03.java EmployeeJava

```
79     System.out.println(e);
80
81     System.out.println("\nEmps sorted by sal in desc order: ");
82     // single liner lambda expression doesn't need curly braces
83     // and the result of expression is considered to be returned
84     Arrays.sort(arr, (x, y) -> -Double.compare(x.getSal(), y));
85     for (Employee e : arr)
86         System.out.println(e);
87
88     System.out.println("\nEmps list sorted by id in desc order:");
89     List<Employee> list = Arrays.asList(arr);
90     list.sort((x,y) -> -Integer.compare(x.getId(), y.getId()));
91     list.forEach(e -> System.out.println(e.toString()));
92 }
93 }
94
95
96
97
98
99
100
101
102
103
104
```

Problems Javadoc Declaration Console

<terminated> Program03 [Java Application] C:\Nilesh\setup\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.core\src\com\sunbeam\Program03.java

Emps list sorted by id in desc order:

| Employee ID | Name | Job | Dept | Salary |
|-------------|------|---------|----------|--------|
| 9 | N | Manger | Sales | 20000 |
| 8 | X | Manager | Accounts | 18000 |
| 7 | H | Clerk | Research | 15000 |
| 6 | A | Analyst | Research | 12000 |
| 5 | D | Clerk | Accounts | 10000 |
| 4 | B | Clerk | Sales | 7000 |
| 3 | G | Analyst | Sales | 6000 |
| 2 | P | Clerk | Research | 5000 |
| 1 | Q | Analyst | Research | 4000 |

Search 12:10 PM

Lambda expressions are
executed with a special
byte-code instruction
i.e. "invokedynamic"

```
day15 - demo03/src/com/sunbeam/Program03.java - Spring Tool Suite 4
You are screen sharing
File Edit Source Refactor Navigate Search Project Run Window Help
Program03.java EmployeeJava
70     });
71     for (Employee e : arr)
72         System.out.println(e);
73
74     System.out.println("\nEmps sorted by sal in asc order: ");
75     Arrays.sort(arr, (x, y) -> {
76         return Double.compare(x.getSal(), y.getSal());
77     });
78     for (Employee e : arr)
79         System.out.println(e);
80
81     System.out.println("\nEmps sorted by sal in desc order: ");
82     // single liner lambda expression doesn't need curly braces
83     // and the result of expression is considered to be returned.
84     Arrays.sort(arr, (x, y) -> -Double.compare(x.getSal(), y.getSal()));
85     for (Employee e : arr)
86         System.out.println(e);
87
88     System.out.println("\nEmps list sorted by id in desc order: ");
89     List<Employee> list = Arrays.asList(arr);
90     list.sort((x,y) -> -Integer.compare(x.getId(), y.getId()));
91     list.forEach(e -> System.out.println(e.toString()));
92 }
93 }
```

