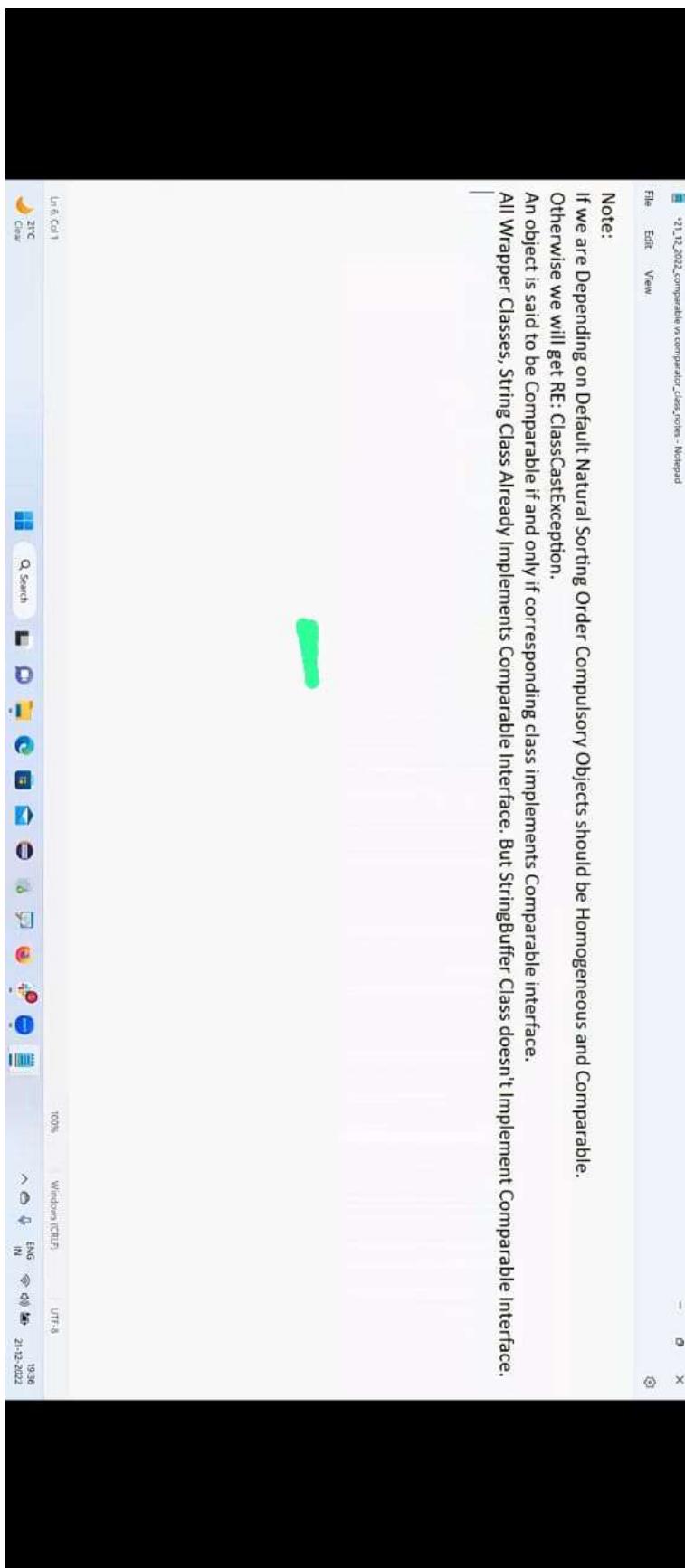


43Java_StreamApi





```
Editor - D:\Whisper classes\TempJava1
File Edit View Search Document Project Tools Browser Emmet Window Help
D:\ New Volume D:\ Whisper classes ItemTest.java TestJava
1 2 3 4 5 6 7 8
1 /* 2 interface Comparable{
2     public int compareTo(Object obj);
3 }
4
5
6 If we keep objects inside TreeSet, internally JVM uses compareTo() and it sorts the
7 object, based on the sorting result the object will be stored in TreeSet.
8 */
9 class Test
10 {
11     public static void main(String[] args)
12     {
13         TreeSet ts = new TreeSet();
14         ts.add("A");
15         ts.add("Z");
16         System.out.println("A".compareTo("Z"));
17         System.out.println("Z".compareTo("K"));
18         System.out.println("Z".compareTo("Z"));
19         System.out.println("Z".compareTo(null));
20     }
21 }
```

In 15 col 19 27 00 PC ANSI

For Help press F1

20C Clear

Editor - (D:\Whisper classes\TreeSet.java)

File Edit View Search Document Project Tools Browser Emmet Window Help

Directory Object Functions

[D:] New Volume D:\ Whisper classes

1 object, based on the sorting result the object will be stored in TreeSet.

2 obj1.compareTo(obj2)

3 => returns -ve value, if obj1 has to come before obj2

4 => returns +ve value, if obj1 has to come after obj2

5 => returns 0 value, if both obj1 and obj2 are equal

6 */

7

8

9

10

11

12

13

14 class Test

15 {

16 public static void main(String[] args)

17 {

18 TreeSet ts = new TreeSet();

19 ts.add("A");

20 ts.add("Z");

21 ts.add("K");

22 System.out.println(ts);

23

24 System.out.println("A".compareTo("Z"));

25 System.out.println("Z".compareTo("K"));

26 System.out.println("Z".compareTo("Z"));

27 System.out.println("Z".compareTo(null));

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

990

991

992

993

994

995

996

997

998

999

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1020

1021

1022

1023

1024

1025

1026

1027

1028

1029

1020

1021

1022

1023

1024

1025

1026

1027

1028

1029

1030

1031

1032

1033

1034

1035

1036

1037

1038

1039

1030

1031

1032

1033

1034

1035

1036

1037

1038

1039

1040

1041

1042

1043

1044

1045

1046

1047

1048

1049

1040

1041

1042

1043

1044

1045

1046

1047

1048

1049

1050

1051

1052

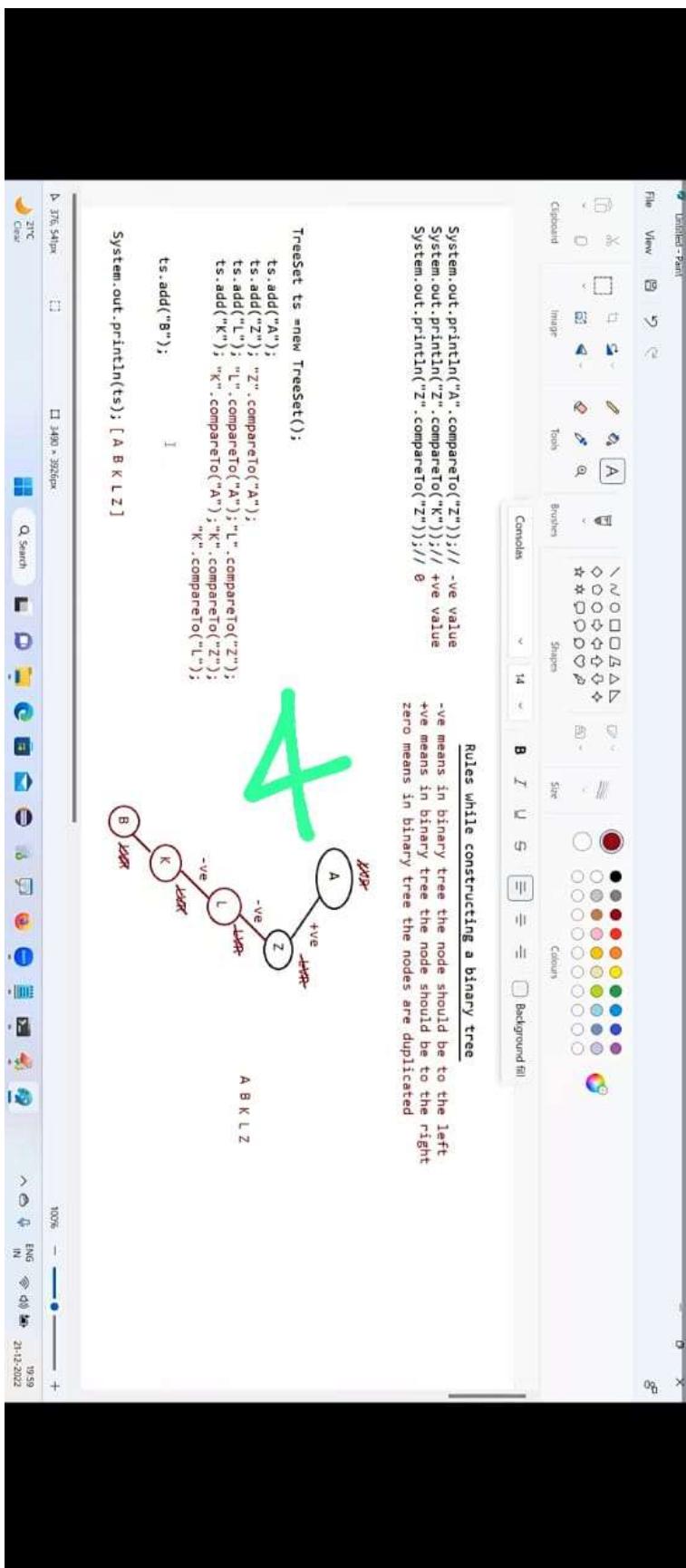
1053

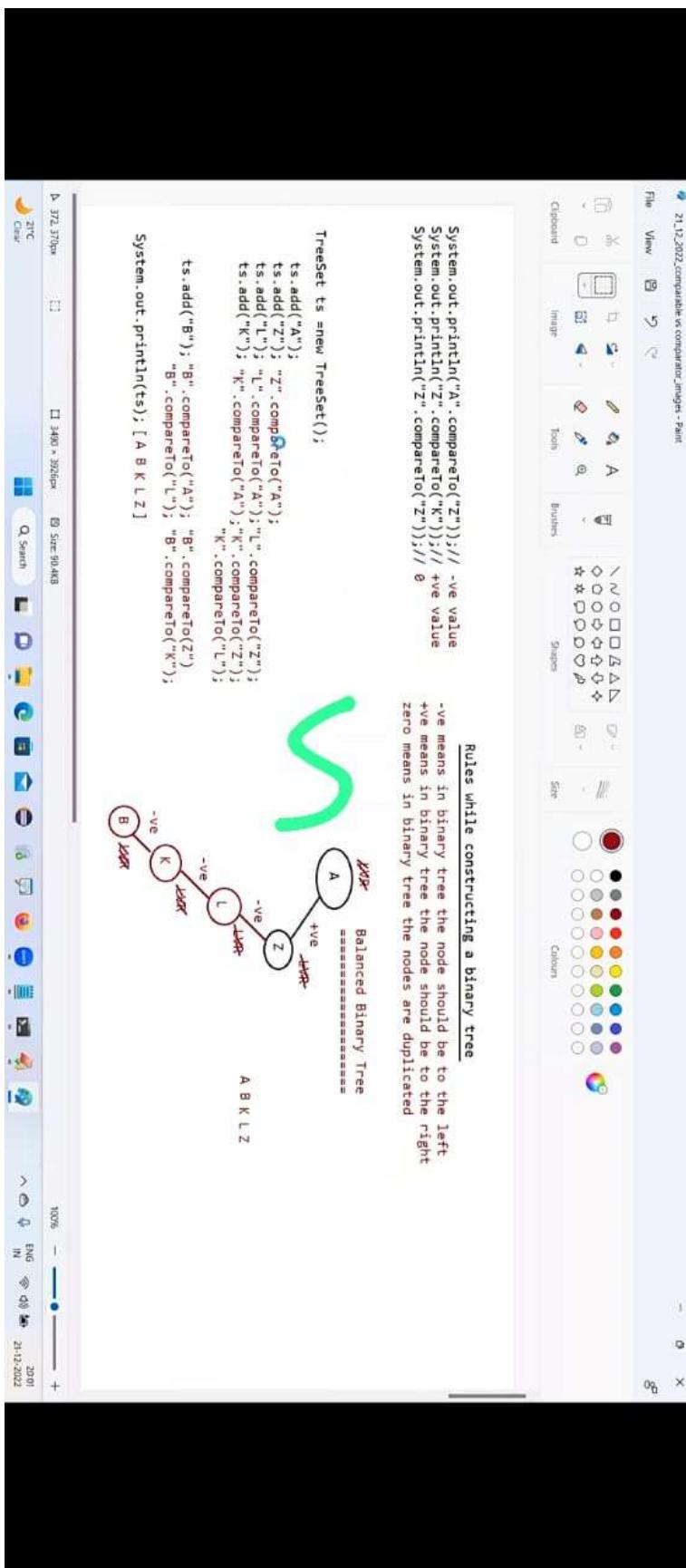
1054

1055

1056

1057





2NC
Chair

Q. Search

Minimize Maximize Close

Command prompt

```
D:\Wrapper classes>javap -java.util.Comparator
Compiled from "Comparator.java"
public interface java.util.Comparator<T> {
    public abstract int compare(T, T);
    public abstract boolean equals(java.lang.Object);
    public java.util.Comparator<T> reversed();
    public <U> java.util.Comparator<T> thenComparing(java.util.Function<T, U>);
    public <U> extends java.lang.Comparable<U> super U> java.util.Comparator<T> thenComparingFunction();
    public java.util.Comparator<T> thenComparingInt(int);
    public java.util.Comparator<T> thenComparingLong(long);
    public java.util.Comparator<T> thenComparingDouble(double);
    public java.util.Comparator<T> thenComparingBoolean(boolean);
    public static <T> extends java.lang.Comparable<T> super T> java.util.Comparator<T> reversed();
    public static <T> extends java.util.Comparator<T> nullSafe();
    public static <T, U> java.util.Comparator<T> nullSafeFunction();
    public static <T, U> java.util.Comparator<T> comparing(java.util.Function<T, U>);
    public static <T, U> java.util.Comparator<T> comparingFunction();
    public static <T, U> java.util.Comparator<T> comparingInt(int);
    public static <T, U> java.util.Comparator<T> comparingLong(long);
    public static <T, U> java.util.Comparator<T> comparingDouble(double);
}
```

D:\Wrapper classes>

21.12.2022_comparable_vs_comparator_class_notes - Notepad

At Line 1 if we are Not Passing Comparator Object as an Argument then Internally JVM will Call compareTo().

In this Case the Output is [0, 5, 10, 15, 20].

III. ፳፻፲፭. የትምህር አገልግሎት ስርዓት ተስፋዣ

At Line 1 if we are Passing Comparator Object then JVM will Call compare() Instead of compareTo(). Which Is Meant for Customized Sorting (can be Ascending /Descending Order). In this Case the Output is [20, 15, 10, 5, 0]

In this Case the Output is [20, 15, 10, 5, 0]

In this Case the Output is [20, 15, 10, 5, 0]

卷之三

16

21-12-2022-comparable vs comparator-class-notes - Notepad

File Edit View

```
TreeSet t = new TreeSet();
t.add("K");
t.add("Z"); "Z".compareTo("K");
t.add("A"); "A".compareTo("K");
t.add("A"); "A".compareTo("A");
System.out.println(t); // [A,K,Z] => Sorting is ascending order
```

Note:

For String default natural sorting order is "Ascending order".

For Number default natural sorting order is "Ascending order".

Comparator()

Note: If we are Not satisfied with Default Natural Sorting Order OR if Default Natural Sorting Order is Not Already Available then we can Define Our Own Sorting by using Comparator Object.

```
public interface java.util.Comparator<T> {
    public abstract int compare(T, T);
    public abstract boolean equals(java.lang.Object);
}
```

21-12-2022-comparable vs comparator-class-notes - Notepad

File Edit View

Q. Search

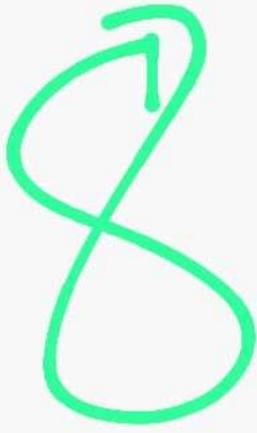


100% Windows (C:\EF7) UTF-8
21-12-2022

```
*21.12.2022 Comparable vs comparator_Ganes_Notes - Notepad
File Edit View
public interface java.util.Comparator<T> {
    public abstract int compare(T, T);
    public abstract boolean equals(java.lang.Object);
}

class MyComparator implements Comparator{
    @Override
    public int compare(Object obj1, Object obj2){
        ;;;;
    }
}

compare(Object obj1, Object obj2)
|=> returns -ve value, if obj1 has to come before obj2
|=> returns +ve value, if obj1 has to come after obj2
|=> returns 0 value, if both obj1 and obj2 are equal
```



```
21-12-2022-comparable vs comparator-class-notes - Notepad  
File Edit View  
public interface java.util.Comparator<T> {  
    public abstract int compare(T, T);  
    public abstract boolean equals(java.lang.Object);  
}  
  
Comparator():  
This Interface Present in java.util Package.  
Methods: It contains 2 Methods compare() and equals().  
  
public int compare(Object obj1, Object obj2);  
Returns -ve if and Only if obj1 has to Come Before obj2.  
Returns +ve if and Only if obj1 has to Come After obj2.  
Returns 0 if and Only if obj1 and obj2 are Equal.  
  
public boolean equals(Object o);  
  
Whenever we are implementing Comparator Interface Compulsory we should Provide Implementation for compare().  
Implementing equals() is Optional because it is Already Available to Our Class from Object Class through Inheritance.
```



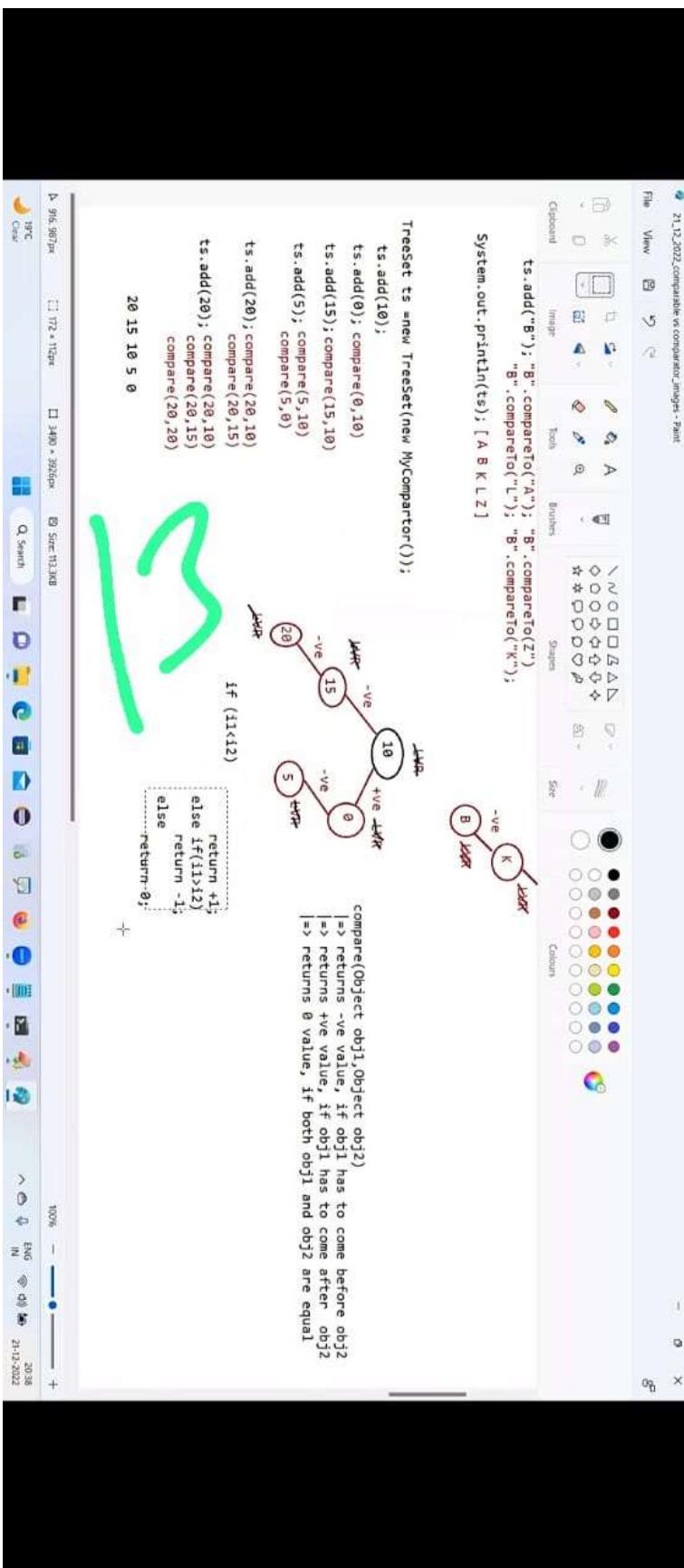
```
//Customized sorting
interface Comparator{
    public int compare(Object obj1, Object obj2);
    public boolean equals(Object obj);
}

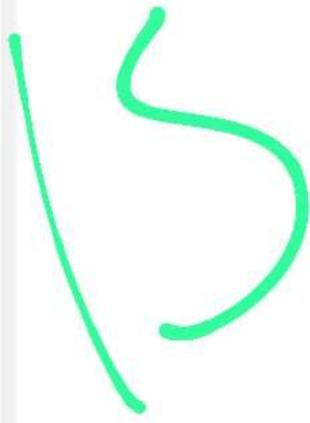
compare(Object obj1, Object obj2)
|=> returns -ve value, if obj1 has to come before obj2
|=> returns +ve value, if obj1 has to come after obj2
|=> returns 0 value, if both obj1 and obj2 are equal
```

10

```
D:\Wrapper\classes>javap java.util.TreeSet
Compiled from "TreeSet.java"
public class java.util.TreeSet<E> extends java.util.AbstractSet<E> implements java.util.NavigableSet<E>, java.lang.Cloneable,
java.io.Serializable {
    public java.util.TreeSet();
    public java.util.TreeSet(java.util.NavigableMap<E, java.lang.Object>);
    public java.util.TreeSet(java.util.Comparator<? super E>);
    public java.util.TreeSet(Collection<? extends E>);
    public java.util.TreeSet(TreeSet<E>);
    public java.util.TreeSet(TreeSet<E> iterator());
    public java.util.Iterator<E> descendingIterator();
    public java.util.NavigableSet<E> descendingSet();
    public boolean isEmpty();
    public int size();
    public boolean contains(Object);
    public boolean add(E);
    public boolean remove(Object);
    public void clear();
    public boolean addAll(Collection<? extends E>);
    public boolean subSet(E, boolean, E, boolean);
    public java.util.NavigableSet<E> headSet(E, boolean);
    public java.util.NavigableSet<E> tailSet(E, boolean);
    public java.util.SortedSet<E> subSet(E, E);
    public java.util.SortedSet<E> headSet(E);
    public java.util.SortedSet<E> tailSet(E);
    public java.util.Comparator<? super E> comparator();
    public E first();
    public E last();
    public E lower(E);
    public E floor(E);
    public E ceiling(E);
```

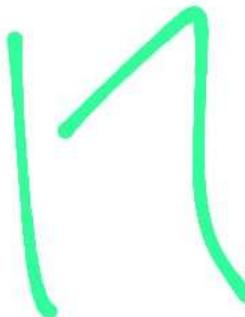
```
Editor - D:\Whisper classes\Net.java
File Edit View Search Document Project Icons Browser Ethernet Window Help
D:\ New Volume D:\ Whisper classes
Item Test.java
Test.java
24
25
26 */
27 class MyComparator implements Comparator<Object> {
28     @Override
29     public int compare(Object obj1, Object obj2) {
30         //logic coming soonnnnn.....
31         Integer i1 = (Integer) obj1;
32         Integer i2 = (Integer) obj2;
33
34
35         if (i1 < i2)
36             return +1;
37         else if (i1 > i2)
38             return -1;
39     }
40
41
42
43     }
44 class Test
```





```
Editor - D:\Whisper classes\setinterval
File Edit View Search Document Project Icons Browser Ethernet Window Help
D:\ New Volume D:\ Whisper classes
ItemTest.java
1
2
3
4
5
6
7
8
46 }
47 class Test
48 {
49     public static void main(String[] args)
50     {
51
52         TreeSet ts = new TreeSet(new MyComparator());
53         ts.add(10);
54         ts.add(0);
55         ts.add(15);
56         ts.add(5);
57         ts.add(20);
58         ts.add(20);
59         System.out.println(ts);
60
61     }
62 }
63
64
65
66
67
Java(TM) SE Runtime Environment
Java(TM) SE Runtime Environment
For Help press F1
IPPC
Clear
Q. Search
In 53 col 20 60 31 PC ANSI
^ ⌂ ⌄ ENG ⇢ CHS
IN 21/12/2022
```

The code in the editor is a Java program named ItemTest.java. It defines a class Test with a main method. Inside the main method, a TreeSet is created using a custom comparator (MyComparator). Elements are added to the set, and then it is printed to the console. A green hand-drawn heart shape is overlaid on the code area.



```
Editor - D:\Whisper classes\Sort.java
File Edit View Search Document Project Tools Browser Emett Window Help
D:\ New Volume D:\ Whisper classes Item Test.java Test.java
28 */
29 class MyComparator implements Comparator<
30 {
31     @Override
32     public int compare(Object obj1, Object obj2){
33         System.out.println("Every comparison the compare() is called");
34         Integer i1 =(Integer)obj1;
35         Integer i2 =(Integer)obj2;
36
37         return -i2.compareTo(i1); //Ascending order
38
39         // i1.compareTo(i2) ===> Ascending order
40         // -i1.compareTo(i2) ===> Descending order
41         // i2.compareTo(i2) ===> Descending order
42         // -i2.compareTo(i1) ===> Ascending order
43
44     }
45 }
46 class Test
47 {
48     public static void main(String[] args)
49     {
50     }
}
```



```
class TreeSetDemo {
    public static void main(String[] args) {
        TreeSet t = new TreeSet(new MyComparator());
        t.add("sachin");
        t.add("ponting");
        t.add("sangakara");
        t.add("fleming");
        t.add("ara");
        System.out.println(t);
    }
}

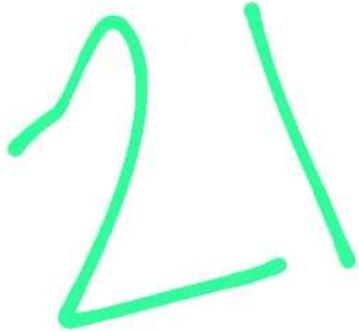
class MyComparator implements Comparator {
    public int compare(Object obj1, Object obj2) {
        String s1 = obj1.toString();
        String s2 = (String)obj2;
        return s2.compareTo(s1);
    }
}
```

Write a Program to Insert StringBuffer Objects into the TreeSet where Sorting Order is Alphabetical Order:



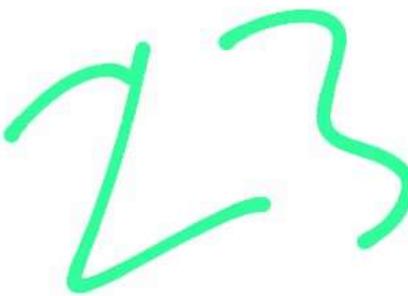
```
Editor - D:\Whisper classes\Java\src\java  
File Edit View Search Document Project Tools Browser Ethernet Window Help  
Directory Object Folders  
[D] New Volume  
D:\ Whisper classes  
36  
37 }  
38 }  
39 class Test  
40 {  
41     public static void main(String[] args)  
42     {  
43         TreeSet ts = new TreeSet(new MyComparator());  
44         ts.add("sachin");  
45         ts.add("ponting");  
46         ts.add("gayle");  
47         ts.add("devilliers");  
48         ts.add("jaysuriya");  
49         System.out.println(ts);//Ascending order  
50     }  
51 }  
52 }  
53 }  
54 }  
55 }  
56 }  
57 }  
Java(TM) SE Runtime Environment  
Java(TM) SE Runtime Environment  
For Help press F1  
IPC  
Clear
```

In 42 col 6 59 00 PC ANSI



```
Editor - D:\Whisper classes\Java\Java  
File Edit View Search Document Project Icons Browser Ethernet Window Help  
Directory Object Functions  
(D) New Volume  
D:\ Whisper classes  
Item Test.java  
Text Java  
4.1  
4.2 {  
4.3  
4.4 {  
4.5  
4.6  
4.7  
4.8  
4.9  
5.0  
5.1  
5.2  
5.3  
5.4  
5.5  
5.6  
5.7  
Java (Java)  
@ TestJava  
For Help press F1  
197C  
Clear  
Q. Search  
In 49 col 49 59 38 PC ANSI  
^ < > < > ENG ⇧ ⇩ ⇨ ⇨ 21/12/2022
```

```
36 }  
37 }  
38 }  
39 class Test  
40 {  
41     public static void main(String[] args)  
42     {  
43         TreeSet ts = new TreeSet(new MyComparator());  
44         ts.add(new StringBuffer("sachin"));  
45         ts.add(new StringBuffer("ponting"));  
46         ts.add(new StringBuffer("gavile"));  
47         ts.add(new StringBuffer("devilliers"));  
48         ts.add(new StringBuffer("jaysuriya"));  
49         System.out.println(ts); //Ascending order  
50     }  
51 }  
52 }  
53 }  
54 }  
55 }  
56 }  
57 }
```

```
Editor - Eclipse IDE for Java Developers (Metropolis)
File Edit View Search Document Project Tools Browser Emmet Window Help
1 2 3 4 5 6 7 8
Directory Object Functions
[D] New Volume
D\ Wiper classes
Item Test.java
Test.java
Java (*.java)
Java (*.jar)
for Help press F1
19PC Clear
Q. Search
In 35 col 33 5 00 PC ANSI
^ < > < > ENG ⇧ ⇩ ⌂ ⌂ 21/12/2022
```

```
28 class MyComparator implements Comparator<Object>
29 {
30     @Override
31     public int compare(Object obj1, Object obj2){
32         String s1 = obj1.toString();
33         String s2 = obj2.toString();
34         //natural sorting order
35         return s1.compareTo(s2);
36     }
37 }
38
39 class Test
40 {
41     public static void main(String[] args)
42 {
43
44     TreeSet<StringBuffer> ts = new TreeSet<StringBuffer>(new MyComparator());
45     ts.add(new StringBuffer("sachin"));
46     ts.add(new StringBuffer("ponting"));
47     ts.add(new StringBuffer("gavile"));
48     ts.add(new StringBuffer("debelliers"));
49 }
```




```
Editor - [D:\Wrapper classes\test.java]
File Edit View Search Project Tools Browser Errors Window Help
D:\ Wrapper classes
Directory Output fractions
[Output Volume]
Wrapper classes
ItemTest.java
Text.java
ItemTest.java
Text.java
Java (*.java)
Test.java
Test.java
File Help prev F1
19% Clear
Search
PC ANSI
21-12-2022 21:15
29: {
30:     @Override
31:     public int compare(Object obj1, Object obj2){
32:         String s1 = obj1.toString();
33:         String s2 = obj2.toString();
34:
35:         int i1 = s1.length();
36:         int i2 = s2.length();
37:
38:         if (i1 < i2)
39:             return -1;
40:         else if (i1 > i2)
41:             return +1;
42:         else
43:             return s1.compareTo(s2);
44:     }
45: }
46: class Test
47: {
48:     public static void main(String[] args)
49:     {
50:     }
51: }
```



```
21-12-2022-comparable vs comparator-class-notes - Notepad  
File Edit View  
  
int i1 = s1.length();  
int i2 = s2.length();  
if(i1 < i2) return -1;  
else if(i1 > i2) return 1;  
else return s1.compareTo(s2);  
}  
  
Note:  
if we are use TreeSet(), then the condition is  
a. Object should be homogenous.  
b. Object should be comparable(class should implement Comparable(l));  
  
if we are use TreeSet(Comparator c) then what is the condition?  
a. Object need not be homogenous.  
b. Object need not implement Comparable.
```

27/12/2022 Comparable & Comparable Class Notes - RichPad
File Edit View

if we are use TreeSet(Comparator c) then what is the condition?

- a. Object need not be homogenous.
- b. Object need not implement Comparable.

When to go for Comparable interface and When to go Comparator interface?

Ans. Predefined Comparable classes like String,Wrapper class =====> Default natural sorting is already available if we are not happy with natural sorting order, we want customization then we need to go for "Comparator()".

For Predefined Non-Comparable class like StringBuffer => Comparator() is used for both natural sorting order and and customized sorting order.

For userdefined class like Employee,Student =====> Developer if he comes up with own logic of sorting,then he should implement comparable()

27

Ln 272 Col 36 10% Windows (CET) UTF-8 ↻ ENG ⇔ DFL 27/12/2022 21:42:20
Q. Search Clear

21-12-2022_comparable vs comparator_class_notes - Notepad

File Edit View

For Predefined Non-Comparable class like StringBuffer => Comparator() is used for both natural sorting order and and customized sorting order.

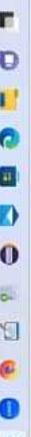
For userdefined class like Employee, Student =====> Developer if he comes up with own logic of sorting, then he should implement Comparable() and give it as a ready made logic.

```
class Employee implements Comparable  
{  
    int id;  
    String name;  
    int age;  
  
    public int compareTo(Object obj){  
        //sorting is done based on "id"  
        .....  
    }  
}
```

20

ABC
Clear

Q. Search



In 217_Cal13

100% Windows (CE/P)

UTF-8

⑤

21:49

21-12-2022

```
*21.12.2022 Comparable vs comparator class.java - Notepad  
File Edit View  
use "Comparator".
```

When we go for Comparable and When we go for Comparator:
Comparable Vs Comparator:

- => For Predefined Comparable Classes (Like String) Default Natural Sorting Order is Already Available. If we are Not satisfied with can Define Our Own Sorting by Comparator Object.
- => For Predefine Non- Comparable Classes (Like StringBuffer) Default Natural Sorting Order is Not Already Available.
If we want to Define Our Own Sorting we can Use Comparator Object.
- => For Our Own Classes (Like Employee) the Person who is writing Employee Class he is Responsible to Define Default Natural Sorting
Order by implementing Comparable Interface.
- => The Person who is using Our Own Class if he is Not satisfied with Default Natural Sorting Order he can Define his Own Sorting
If he is satisfied with Default Natural Sorting Order then he can Use Directly Our Class.

29

Editor - D:\Whisper classes\TetJava1

File Edit View Search Document Project Tools Brower Ethernet Window Help

Directory Object Functions

[D] New Volume

D\ Whisper classes

1
2
3
4
5
6
7
8

```
47 }  
48 }  
49 }  
50 class Test  
51 {  
52     public static void main(String[] args)  
53 {  
54     TreeSet ts = new TreeSet();  
55     ts.add(new Employee("sachin",10));  
56     ts.add(new Employee("ponting",14));  
57     ts.add(new Employee("gayle",99));  
58     ts.add(new Employee("develiers",17));  
59     ts.add(new Employee("develiers",17));  
60     }  
61     }  
62     }  
63     }  
64     }  
65     }  
66     }  
67     }
```

Java (*.java)

TetJava

For Help press F1

IBC Clear

Q. Search

In 57 col 9 60 74 PC ANSI

~ ⌂ ⌄ ENG ⇔ CHS 2012-2022

30

Editor: D:\Wiper\class [Test.java]

File Edit View Search Document Project Icons Browser Emett Window Help

Directory Object Functions

[D] New Volume D:\ Wiper\classes

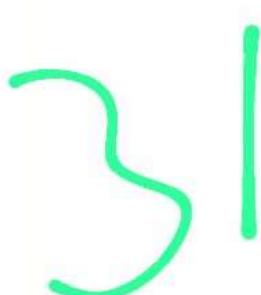
ItemTest.java

Test.java

In 55 col 20 69 45 PC ANSI

For Help press F1 IBC Clear

```
47 } 1
48 } 2
49 } 3
50 class Test 4
51 { 5
52     public static void main(String[] args) 6
53     { 7
54         TreeSet ts = new TreeSet(); 8
55         ts.add(new Employee("sachin",10)); 9
56         ts.add(new Employee("ponting",14)); 10
57         ts.add(new Employee("gayle",99)); 11
58         ts.add(new Employee("develiers",17)); 12
59         ts.add(new Employee("develiers",17)); 13
60         System.out.println(ts); 14
61     } 15
62 } 16
63 } 17
64 } 18
65 } 19
66 } 20
67 }
```



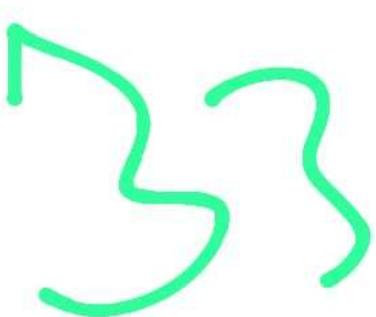
Editor - Employee class.java

```
File Edit View Search Document Project Icons Browser Ethernet Window Help
Directory Object Functions
[D] New Volume
D:\ Websphere
ItemTest.java
HelloWorld
Java (*.java)
* Test.java
For Help press F1
IBPC Clear
In 35 00:37 69 00 PC ANSI
Q. Search L D E C A S M G P N ^ & F ENG ⇢ CHS 21/12/2022
```

```
35
36 class Employee implements Comparable<Employee>
37 {
38     String name;
39     int eid;
40
41     Employee(String name, int eid){
42         this.name = name;
43         this.eid = eid;
44     }
45
46     public String toString(){
47         return name + " ==> " + eid;
48     }
49 }
50 class Test
51 {
52     public static void main(String[] args)
53     {
54
55         TreeSet ts = new TreeSet();
56         ts.add(new Employee("Rahul", 10));
57         ts.add(new Employee("Karan", 20));
58         ts.add(new Employee("Aman", 30));
59         ts.add(new Employee("Vishal", 40));
60
61         for (Employee e : ts)
62             System.out.println(e);
63     }
64 }
```

32

```
Editor - Eclipse IDE for Java Developers
File Edit View Search Document Project Tools Browser Ethernet Window Help
1 2 3 4 5 6 7 8
Directory Object Folders
[D] New Volume
D:\ Wizpaper classes
Item Test.java
Test.java
62 class Test
63 {
64     public static void main(String[] args)
65     {
66         TreeSet ts = new TreeSet();
67         Employee e1 = new Employee("sachin",10);
68         Employee e2 = new Employee("ponting",14);
69         Employee e3 = new Employee("gayle",99);
70         Employee e4 = new Employee("develliers",17);
71
72         ts.add(e1);
73         ts.add(e2);
74         ts.add(e3);
75         ts.add(e4);
76
77         System.out.println(ts);
78     }
79 }
80
81
82
```



Eclipse (jWrapper classes) [Test]
File Edit View Document Project Tools Browser Emmet Window Help
D:\ New volume D:
Wrapper classes
Directory Client functions
File Edit View Search Document Project Tools Browser Emmet Window Help
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71

```
50
51     @Override
52     public int compareTo(Object obj1){
53         //DNSO -> based on id, sort the objects
54
55         int id1 = this.eid;//id1 -> FirstEmployee object id
56         Employee emp = (Employee) obj1;
57         int id2 = emp.eid;// id2 -> SecondEmployee object id
58
59         if (id1<id2)
60             return -1;
61         else if(id1>id2)
62             return +1;
63         else
64             return 0;
65     }
66 }
67 class Test {
68     public static void main(String[] args) {
69         TreeSet ts = new TreeSet();
70         Employee e1 = new Employee("sachin", 10);
71         Employee e2 = new Employee("manoj", 11);
72         ts.add(e1);
73         ts.add(e2);
74     }
75 }
```

Java (1.java)
Test.java
For Help, press F1

ln 64 col 22 10 00 PC ANSI



Embarcadero RAD Studio XE7

File Edit View Search Document Project Tools Browser Embarcadero Window Help

Directory Object Functions

[D:\] New Volume D:\ WinZip classes

Item Test.java Test.java

26
27 */
28 class MyComparator implements Comparator
29 {
30 @Override
31 public int compare(Object obj1, Object obj2){
32 //Customization -> Sort based on name
33 Employee e1 = (Employee)obj1;
34 Employee e2 = (Employee)obj2;
35
36 String i1 = e1.name;
37 String i2 = e2.name;
38
39 return i1.compareTo(i2);
40 }
41 }
42
43 class Employee implements Comparable
44 {
45 String name;
46 int eid;

Java (*.java)
Test.java

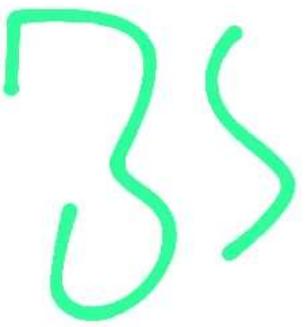
For Help press F1

IBC Clear

Q. Search

In 31 col 21 104 -72 PC ANSI

ENGLISH QWERTY IN 21/12/2022





```
D:\Wrapper\classes>javac Test.java
Note: Test.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

D:\Wrapper\classes>java Test
[sachin ==> 10, ponting ==> 14, develliers ==> 17, gayle ==> 99]
[develliers ==> 17, gayle ==> 99, ponting ==> 14, sachin ==> 10]

D:\Wrapper\classes>
```

Comparison of Comparable and Comparator:

Comparable()

Present in **java.lang** Package

It is Meant for Default Natural Sorting Order.

Defines Only One Method **compareTo()**

All Wrapper Classes and String Class implements Comparable Interface.

Comparator()

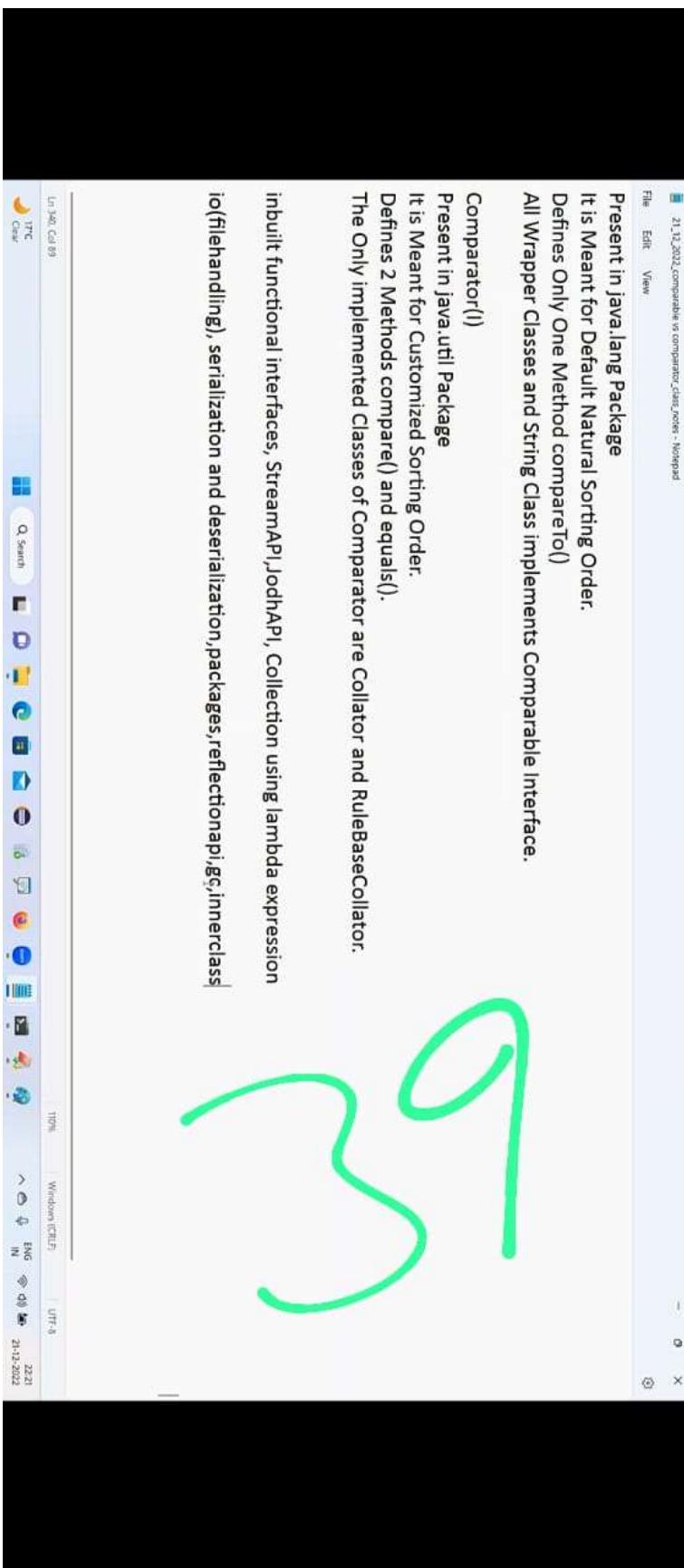
Present in **java.util** Package

It is Meant for Customized Sorting Order.

Defines 2 Methods **compare()** and **equals()**.

The Only implemented Classes of Comparator are Collator and RuleBaseCollator.

38



21-12-2022\snippets\session_2notes - Notepad

File Edit View

Question

Consider below code:

```
public class Test {  
    static Double d1; // d1 =null  
    static int x = d1.intValue(); // null.intValue() ---> NullPointerException  
  
    public static void main(String[] args) {  
        System.out.println("HELLO");  
    }  
}
```

On execution, does Test class print "HELLO" on to the console?

A:Yes HELLO is printed on the console
B.NO Hello is not printed on the console

Answer: B

40

21-12-2022\snippets\session_2022 - Notepad

File Edit View

Question

Consider below code:

```
public class Test {  
    static Double d1; // static variable =====> d1 =null  
    int x = d1.intValue(); //instance variable =====> only upon creating an object  
  
    public static void main(String[] args) {  
        System.out.println("HELLO"); //HELLO  
    }  
}
```

On execution, does Test class print "HELLO" on to the console?

A.Yes HELLO is printed on the console
B.NO Hello is not printed on the console

Answer: A

41

```
File Edit View  
E:\Java\Java\src\com\javapoint\exception  
public static void main(String[] args) {  
    Error obj = new Error();  
    boolean flag1 = obj instanceof RuntimeException; //Line n1  
    boolean flag2 = obj instanceof Exception; //Line n2  
    boolean flag3 = obj instanceof Error; //Line n3  
    boolean flag4 = obj instanceof Throwable; //Line n4  
    System.out.println(flag1 + "," + flag2 + "," + flag3 + "," + flag4);  
}  
  
A. Compilation Error  
B. false:false:true:true  
C. false:true:true:true  
D. true:true:true:true  
E. false:true:true:false  
  
String s = "sachin";  
System.out.println( s instanceof String );//true  
System.out.println( s instanceof StringBuffer );//CE  
System.out.println( s instanceof Runnable );//false  
System.out.println( null instanceof StringBuilder );//false
```

42

21/12/2022 snippets/session_2022 - Notepad

File Edit View

```
public static void main(String[] args) {
    Error obj = new Error();
    boolean flag1 = obj instanceof RuntimeException; //Line n1
    boolean flag2 = obj instanceof Exception; //Line n2
    boolean flag3 = obj instanceof Error; //Line n3
    boolean flag4 = obj instanceof Throwable; //Line n4
    System.out.println(flag1 + ":" + flag2 + ":" + flag3 + ":" + flag4);
}
```

A. Compilation Error
B. false:false:true:true
C. false:true:true:true
D. true:true:true:true
E. false:true:true:false

Note: Error and RunTimeException no relation in hierarchy as parent and child
Error and Exception no relation in hierarchy as parent and child

Answer: A

```
String s = "sachin";
System.out.println(s instanceof String); //true
```

Ln: 41 Col: 10 100% Windows (CEP) UTF-8 ↻ ENG ⇔ CHS 21/12/2022 22:59

ITPC Clear





```
D:\Wrapper\classes>javap java.lang.RuntimeException
Compiled from "RuntimeException.java"
public class java.lang.RuntimeException extends java.lang.Exception {
    static final long serialVersionUID;
    public java.lang.RuntimeException();
    public java.lang.RuntimeException(java.lang.String);
    public java.lang.RuntimeException(java.lang.String, java.lang.Throwable);
    public java.lang.RuntimeException(java.lang.Throwable);
    protected java.lang.RuntimeException(java.lang.String, java.lang.Throwable, boolean, boolean);
}

D:\Wrapper\classes>jav|
```

```
D:\Wrapper\classes>javap java.lang.RuntimeException
Compiled from "RuntimeException.java"
public class java.lang.RuntimeException extends java.lang.Exception {
    static final long serialVersionUID;
    public java.lang.RuntimeException();
    public java.lang.RuntimeException(java.lang.String);
    public java.lang.RuntimeException(java.lang.String, java.lang.Throwable);
    public java.lang.RuntimeException(java.lang.Throwable);
    protected java.lang.RuntimeException(java.lang.String, java.lang.Throwable, boolean, boolean);
}

D:\Wrapper\classes>javap java.lang.Exception
Compiled from "Exception.java"
public class java.lang.Exception extends java.lang.Throwable {
    static final long serialVersionUID;
    public java.lang.Exception();
    public java.lang.Exception(java.lang.String);
    public java.lang.Exception(java.lang.String, java.lang.Throwable);
    public java.lang.Exception(java.lang.Throwable);
    protected java.lang.Exception(java.lang.String, java.lang.Throwable, boolean, boolean);
}

D:\Wrapper\classes>javap java.lang.Throwable
Compiled from "Throwable.java"
public class java.lang.Throwable implements java.io.Serializable {
    static final boolean $assertionsDisabled;
    public java.lang.Throwable();
    public java.lang.Throwable(java.lang.String);
    public java.lang.Throwable(java.lang.String, java.lang.Throwable);
    public java.lang.Throwable(java.lang.Throwable);
    protected java.lang.Throwable(java.lang.String, java.lang.Throwable, boolean, boolean);
}
```

450

11:10 AM

```
public java.lang.Exception(java.lang.Throwable); java.lang.Throwable, boolean, boolean)
} protected java.lang.Exception(java.lang.String); java.lang.Throwable, boolean, boolean)

D:\Wrapper\classes>javap java.lang.Throwable
Compiled from "Throwable.java"
public class java.lang.Throwable implements java.io.Serializable {
    static final boolean $assertionsDisabled;
    public java.lang.Throwable();
    public java.lang.Throwable(java.lang.String);
    public java.lang.Throwable(java.lang.String, java.lang.Throwable);
    public java.lang.Throwable(java.lang.Throwable);
    protected java.lang.Throwable(java.lang.String, java.lang.Throwable, boolean, boolean);
    public java.lang.String getMessage();
    public java.lang.String getLocalizedMessage();
    public synchronized java.lang.Throwable getCause();
    public synchronized java.lang.Throwable initCause(java.lang.Throwable);
    public java.lang.String toString();
    public void printStackTrace();
    public void printStackTrace(java.io.PrintWriter);
    public synchronized java.lang.Throwable fillInStackTrace();
    public java.lang.StackTraceElement[] getStackTrace();
    public void setStackTrace(java.lang.StackTraceElement[]);
    native int getStackTraceDepth();
    native java.lang.StackTraceElement getElement(int);
    public final synchronized void addSuppressed(java.lang.Throwable);
    public final synchronized java.lang.Throwable[] getSuppressed();
}
}

D:\Wrapper\classes>
```

ITC
Clear



^ ⌂ ⌄ ENG ⇢ ⌂ 22:43
IN 21/12/2022

```
21-12-2022 snippets/session_notes - Notepad  
File Edit View  
for(Logger logger : loggers)  
    logger.log(); //Line n2  
}  
  
What will be the result of compiling and executing Test class?  
A. Line n1 causes compilation error  
B. Line n2 causes compilation error  
C. Exception is thrown at runtime  
D. No output is displayed but program terminates successfully.
```

valid

====

```
class MyThread1 implements Runnable{  
}  
class MyThread2 implements Runnable{  
}  
Runnable[] runnable = new Runnable[2];  
runnable[0] =new MyThread1();  
runnable[1] =new Mythread2();
```

47

21-12-2022 snippets/session_4 notes - Notepad

File Edit View

Question>

Given code of Test.java file:

```
interface ILogger {  
    void log();  
}
```

```
public class Test {  
    public static void main(String[] args) {  
        ILogger [] loggers = new ILogger[2]; //Line n1 ==> loggers[0]=null; loggers[1]=null;  
        for(ILogger logger : loggers)  
            logger.log(); //Line n2 ==> JVM==> NullPointerException  
    }  
}
```

What will be the result of compiling and executing Test class?

- A. Line n1 causes compilation error
- B. Line n2 causes compilation error
- C. Exception is thrown at runtime
- D. No output is displayed but program terminates successfully.

Answer: C

48

