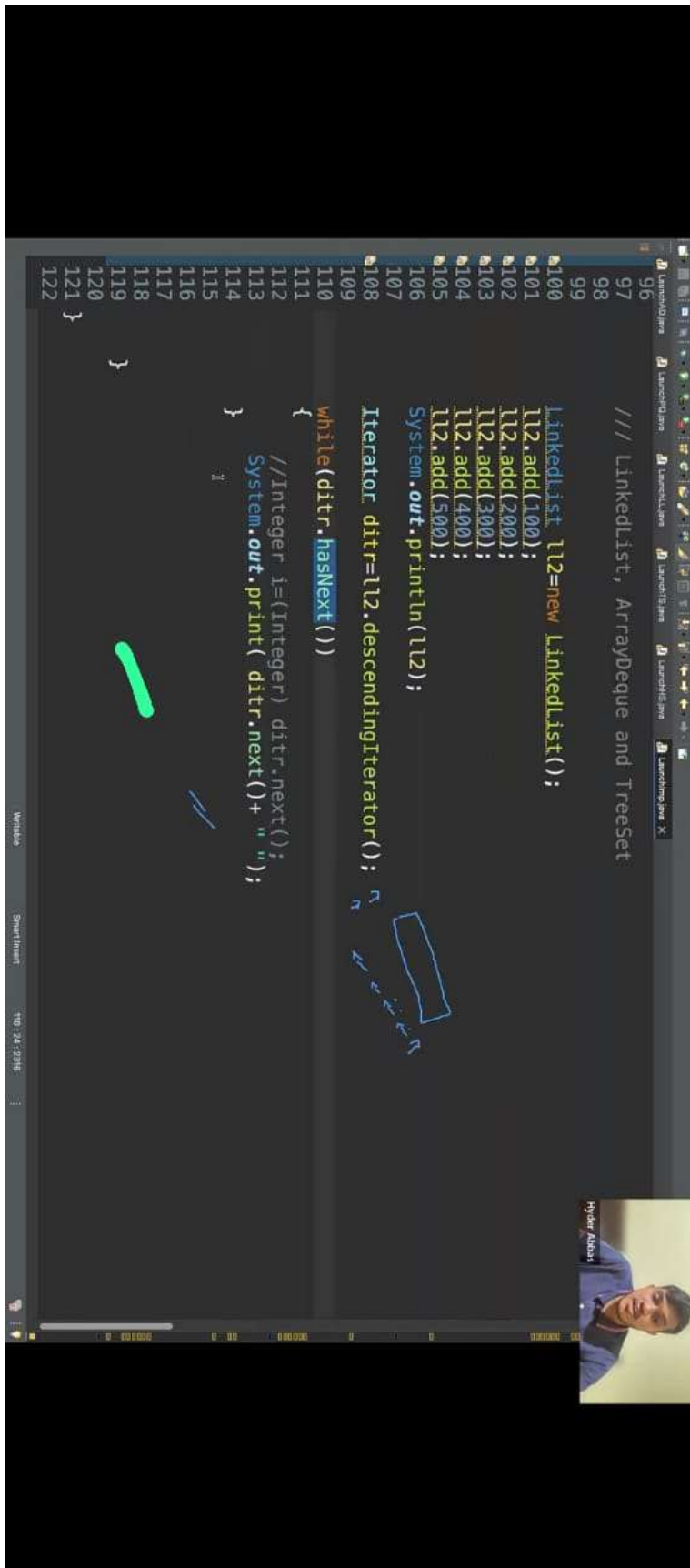


# Java Collection Part-3



The screenshot shows an IDE with a Java file named `LinkedList.java`. The code implements a `LinkedList` class with methods `add`, `print`, and `descendingIterator`. A `while` loop is used to iterate through the list in descending order. A green line is drawn under the `while` loop, and a blue line is drawn under the `descendingIterator` method. A diagram of a linked list structure is drawn in blue, showing a sequence of nodes connected by arrows, with the last node pointing to `null`.

```
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
```

```
/// LinkedList, ArrayDeque and TreeSet  
  
LinkedList ll2=new LinkedList();  
ll2.add(100);  
ll2.add(200);  
ll2.add(300);  
ll2.add(400);  
ll2.add(500);  
System.out.println(ll2);  
  
Iterator ditr=ll2.descendingIterator();  
while(ditr.hasNext())  
{  
    //Integer i=(Integer) ditr.next();  
    System.out.print( ditr.next()+" ");  
}
```

Diagram of a linked list structure:

```
graph LR  
    n1[n1] --> n2[n2]  
    n2 --> n3[n3]  
    n3 --> n4[n4]  
    n4 --> n5[n5]  
    n5 --> null[null]
```





Package Explorer X

LaunchV.java

```
1. import java.util.Enumeration;
2. import java.util.Vector;
3
4. public class LaunchV {
5
6.     // TODO Auto-generated method stub
7
8
9
10
11     Vector v=new Vector();
12     v.add(100);
13     v.add(200);
14     v.add(300);
15     v.add(400);
16     v.add(500);
17
18     Enumeration em=v.elements();
19     while(em.hasMoreElements())
20     {
21         System.out.println(em.nextElement());
22     }
23
24 }
25
```

Workspace

Smart IDE

14:30:24

Hyder Abbas

4

```
26 while(itr.hasNext())
27 {
28     System.out.println(itr.next());
29     //al.add(123);
30 }
31
32 //Fail safe
33
34 CopyOnWriteArrayList cal=new CopyOnWriteArrayList();
35
36 cal.add(1000);
37 cal.add(2000);
38 cal.add(3000);
39 cal.add(4000);
40
41 Iterator itr=cal.iterator();
42 while(itr.hasNext())
43 {
44     System.out.println(itr.next());
45     cal.add(12345);
46 }
47
48
49
50
51 }
```

← 0 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

55



IDE screenshot showing Java code and a video feed of Syder Abbas.

**Code Snippets:**

```
al.add(50);
al.add(150);
al.add(25);
al.add(75);
al.add(125);
al.add(175);
System.out.println(al);

//Collection vs Collections
Collections.sort(al);
System.out.println(al);

ArrayList al2=new ArrayList();
al2.add(28);
al2.add("Hyder");
al2.add("ineuron");
al2.add("Najafi code");
al2.add(560075);
Collections.sort(al2);
System.out.println(al2);
```

**Video Feed:** Syder Abbas

**Green Markers:** A large green number '6' is drawn over the code. A green dot is visible in the top right corner of the IDE window.

The screenshot shows an IDE with a project named 'Lamochi' and a package named 'Lamochi'. The code is as follows:

```
36  
37 ArrayList<String> a12=new ArrayList<String>(  
38 // a12.add(28); error  
39 a12.add("Hyder");  
40 a12.add("ineuron");  
41 a12.add("Najafi code");  
42 //error a12.add(560025);  
43  
44 Collections.sort(a12);  
45 System.out.println(a12);  
46  
47 ArrayList<Integer> a13=new ArrayList<Integer>();  
48 a13.add(1000);  
49 a13.add(200);  
50 // a13.add("GF"); error  
51 Collections.sort(a13);  
52 System.out.println(a13);  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62 }
```


A green arrow points to the line `Collections.sort(a13);` at line 52.



IDE screenshot showing Java code in Eclipse. The code is in a file named `LambdaTest.java`. A large red circle highlights the code block starting at line 41.

```
41 a12.add("Najafi code");
42 //error a12.add(560025);
43
44 Collections.sort(a12);
45 System.out.println(a12);
46
47 ArrayList<Integer> a13=new ArrayList<Integer>();
48 a13.add(1000);
49 a13.add(200);
50 // a13.add("GF"); error
51 Collections.sort(a13);
52 System.out.println(a13);
53
54 //few more important inbuilt methods of Collections class:
55 ArrayList a14=new ArrayList();
56 a14.add(10);
57 a14.add(20);
58 a14.add(30);
59 a14.add(40);
60 a14.add(50);
61 Collections.binarySearch(a14, 40);
62
63
64
65
66
67 }
```

The IDE interface includes a Package Explorer on the left showing a project named `LambdaTest` with several Java files. The bottom status bar shows the time as 8:14:13 AM.





IDE screenshot showing Java code and a video feed of a presenter.

**Code Snippets:**

```
42 //error a12.add(560025);
43
44 Collections.sort(a12);
45 System.out.println(a12);
46
47 ArrayList<Integer> a3=new ArrayList<Integer>();
48 a3.add(1000);
49 a3.add(200);
50 // a3.add("GF"); error
51 Collections.sort(a13);
52 System.out.println(a13);
53
54 //few more important inbuilt methods of Collections class:
55
56 ArrayList a14=new ArrayList();
57 a14.add(10);
58 a14.add(20);
59 a14.add(30);
60 a14.add(40);
61 a14.add(50);
62 int index= Collections.binarySearch(a14, 40);
63 System.out.println("Index " + index);
64
65
66
67
68 }
```

**Project Explorer:**

- src
  - arraylist
  - arraylist2
  - arraylist3
  - arraylist4
  - arraylist5
  - arraylist6
  - arraylist7
  - arraylist8
  - arraylist9
  - arraylist10
  - arraylist11
  - arraylist12
  - arraylist13
  - arraylist14
  - arraylist15
  - arraylist16
  - arraylist17
  - arraylist18
  - arraylist19
  - arraylist20
  - arraylist21
  - arraylist22
  - arraylist23
  - arraylist24
  - arraylist25
  - arraylist26
  - arraylist27
  - arraylist28
  - arraylist29
  - arraylist30
  - arraylist31
  - arraylist32
  - arraylist33
  - arraylist34
  - arraylist35
  - arraylist36
  - arraylist37
  - arraylist38
  - arraylist39
  - arraylist40
  - arraylist41
  - arraylist42
  - arraylist43
  - arraylist44
  - arraylist45
  - arraylist46
  - arraylist47
  - arraylist48
  - arraylist49
  - arraylist50
  - arraylist51
  - arraylist52
  - arraylist53
  - arraylist54
  - arraylist55
  - arraylist56
  - arraylist57
  - arraylist58
  - arraylist59
  - arraylist60
  - arraylist61
  - arraylist62
  - arraylist63
  - arraylist64
  - arraylist65
  - arraylist66
  - arraylist67
  - arraylist68
  - arraylist69
  - arraylist70
  - arraylist71
  - arraylist72
  - arraylist73
  - arraylist74
  - arraylist75
  - arraylist76
  - arraylist77
  - arraylist78
  - arraylist79
  - arraylist80
  - arraylist81
  - arraylist82
  - arraylist83
  - arraylist84
  - arraylist85
  - arraylist86
  - arraylist87
  - arraylist88
  - arraylist89
  - arraylist90
  - arraylist91
  - arraylist92
  - arraylist93
  - arraylist94
  - arraylist95
  - arraylist96
  - arraylist97
  - arraylist98
  - arraylist99

10

```
47 ArrayList<Integer> a3=new ArrayList<Integer>
48 a3.add(1000);
49 a3.add(200);
50 // a3.add("6F"); error
51 Collections.sort(a3);
52 System.out.println(a3);
53
54 //few more important inbuilt methods of Collections class
55 ArrayList a4=new ArrayList();
56 a4.add(10);
57 a4.add(20);
58 a4.add(30);
59 a4.add(40);
60 a4.add(50);
61 int index=Collections.binarySearch(a4, 40);
62 System.out.println("Index " + index);
63
64 Collections.shuffle(a4);
65 System.out.println(a4);
66
67 System.out.println(Collections.frequency(a4, 40));
68
69 }
70
71 }
72 }
73 }
```

Hyder Abbas

FileViewToolsBrushesShapesSizeColors

20°C  
Broadly cloudy

100%  
1480 x 3200px

Q Search

100%  
ENG  
IN  
22:13  
13/12/2022

Untitled - Paint

class Plane {  
 static String s = "-";  
 public static void main(String[] args) {  
 new Plane().s1();  
 System.out.println(s);  
 }  
 void s1() {  
 try { s2(); }  
 catch (Exception e) { s += "c"; }  
 }  
 void s2() throws Exception {  
 s3(); s += "a";  
 }  
 void s3() throws Exception {  
 s3(); s += "2b";  
 throw new Exception();  
 }  
}

s - c

Exception

==

```
try { s2(); }  
catch (Exception e) { s += "c"; }  
}  
void s2() throws Exception {  
    s3(); s += "2";  
    s3(); s += "2b";  
}  
void s3() throws Exception {  
    throw new Exception();  
}  
}  
What is the result?
```

- A. -
- B. -c
- C. -c2
- D. -2c
- E. -c22b
- F. -2c2b
- G. -2c2bc
- H. Compilation fails

Answer: B

2



- C. -c2
- D. -2c
- E. -c22b
- F. -2c2b
- G. -2c2bc
- H. Compilation fails

Answer: B

Given:

```
try { int x = Integer.parseInt("two"); }
```

Which could be used to create an appropriate catch block? (Choose all that apply.)

- A. ClassCastException
- B. IllegalStateException
- C. NumberFormatException
- D. IllegalArgumentException
- E. ExceptionInitializerError
- F. ArrayIndexOutOfBoundsException

answer: C,D

1



```
1. class Loopy {  
2. public static void main(String[] args) {  
3.     int[] x = {7,6,5,4,3,2,1};  
4.     // insert code here  
5.     System.out.println(" ");  
6. }  
7. }
```

Which, inserted independently at line 4, compiles? (Choose all that apply.)

- A. for(int y : x) {
- B. for(x : int y) {
- C. int y = 0; for(y : x) {
- D. for(int y=0, z=0; z<x.length; z++) { y = x[z];
- E. for(int y=0, int z=0; z<x.length; z++) { y = x[z];
- F. int y = 0; for(int z=0; z<x.length; z++) { y = x[z];

answer: A, D, F



Q>

Given:

```
class Emu {  
    static String s = "-"; // -ic mc mf of  
    public static void main(String[] args) {
```

```
        try
```

```
        {
```

```
            throw new Exception();
```

```
        }
```

```
        catch (Exception e) {
```

```
            try
```

```
            {
```

```
                try
```

```
                {
```

```
                    throw new Exception();
```

```
                }
```

```
                catch (Exception ex) {
```

```
                    s += "ic ";
```

```
                }
```

```
            }  
            throw new Exception();
```

51





13.12.2022, enigma\_session - Notepad

File Edit View

```
{
    throw new Exception();
}
catch (Exception e) {
    try
    {
        throw new Exception();
    }
    catch (Exception ex) {
        s += "lc ";
    }
    throw new Exception();
}
catch (Exception x)
{
    s += "mc ";
}
}
finally
```

2

20°C Mostly cloudy

Ln 11, Col 1

Q Search

100% Windows [CTRL]

ENG IN UTT 6 22:33 13.12.2022

Nishu Marjani

File Edit View

```
}  
System.out.println(s);  
}
```

What is the result?

- A. -ic of
- B. -mf of
- C. -mc mf
- D. -ic mf of
- E. -ic mc mf of
- F. -ic mc of mf
- G. Compilation fails

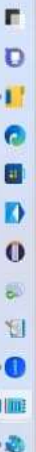
Answer: E

5



Ln 111, Col 1  
20°C Mostly Sunny

Search



100%

Windows [CTRL]

UTF-8

ENG IN 22:33 13.12.2022

File Edit View

Given:

3. class SubException extends Exception { } //ParentException type
4. class SubSubException extends SubException { } //ChildException type
- 5.
6. public class CC { void doStuff() throws SubException { } } //Parent
- 7.
8. class CC2 extends CC { void doStuff() throws SubSubException { } } //Child
- 9.
10. class CC3 extends CC { void doStuff() throws Exception { } } //Child::CE(violate the rule of overriding)
- 11.
12. class CC4 extends CC { void doStuff(int x) throws Exception { } } //child
- 13.
14. class CC5 extends CC { void doStuff() { } } //Child

What is the result? (Choose all that apply.)

- A. Compilation succeeds
- B. Compilation fails due to an error on line 8
- C. Compilation fails due to an error on line 10
- D. Compilation fails due to an error on line 12
- E. Compilation fails due to an error on line 14

Answer: C

1

81

Ln:118 Col:1

20°C  
Brisley, Kenya

Q Search



90%

Windows [CTRL]

UTF-8

ENG  
IN  
22:40  
13-12-2022



Given:

```
3. public class Ebb {  
4.     static int x = 7; // x = 7, 8, 9, 10, 11  
5.     public static void main(String[] args) {  
6.         String s = "", // s = 9 10 10 d 13  
7.         for(int y = 0; y < 3; y++) { // y = 0, 1, 2, 3  
8.             x++;  
9.             switch(x) {  
10.                 case 8: s += "8 ";  
11.                 case 9: s += "9 ";  
12.                 case 10: { s += "10 "; break; }  
13.                 default: s += "d ";  
14.                 case 13: s += "13 ";  
15.             }  
16.             System.out.println(s);  
17.         }  
18.     }  
19.     static { x++; }  
20. }
```

What is the result?

A. 9 10 d  
B. 8 9 10 d  
C. 9 10 10 d  
D. 9 10 10 d 13

51



13.12.2022, support\_session - Notepad

File Edit View

```
9.      switch(x) {
10.          case 8: s += "8 ";
11.          case 9: s += "9 ";
12.          case 10: { s += "10 "; break; }
13.          default: s += "d ";
14.          case 13: s += "13 ";
15.      }
16.      }
17.      System.out.println(s);
18.  }
19.      static { x++; }
20.  }

What is the result?
A. 9 10 d
B. 8 9 10 d
C. 9 10 10 d
D. 9 10 10 d 13
E. 8 9 10 10 d 13
F. 8 9 10 9 10 10 d 13
G. Compilation fails

answer: D
```

20

Notepad

13.12.2022, support\_session - Notepad

File Edit View

90%

Windows (CTRL)

UTF-8

22:50

13.12.2022

ENG

IN

20°C

Cloudy

File Edit View

answer: D

Given:

```
3. class Infinity {  
4. public class Beyond extends Infinity {  
5.     static Integer i; // i = null  
6.     public static void main(String[] args) {  
7.         int sw = (int)(Math.random() * 3);  
8.         switch(sw) {  
9.             case 0: { for(int x = 10; x > 5; x++)  
10.                    if(x > 100000000) x = 10;  
11.                    break; }  
12.             case 1: { int y = 7 * i; break; }  
13.             case 2: { infinity inf = new Beyond();  
14.                    Beyond b = (Beyond)inf;  
15.                    }  
16.             }  
17. }
```

And given that line 7 will assign the value 0, 1, or 2 to sw, which are true? (Choose all that apply.)

- A. Compilation fails
- B. A ClassCastException might be thrown
- C. A StackOverflowError might be thrown
- D. A NullPointerException might be thrown
- E. An IllegalStateException might be thrown



File Edit View

```
6. public static void main(String[] args) {
7.     int sw = (int)(Math.random() * 3);
8.     switch(sw) {
9.         case 0: { for(int x = 10; x > 5; x++)
10.             if(x > 10000000) x = 10;
11.             break; }
12.         case 1: { int y = 7 * i; break; }
13.         case 2: { infinity inf = new Beyond();
14.             Beyond b = (Beyond)inf; }
15.     }
16. }
17. }
```

And given that line 7 will assign the value 0, 1, or 2 to sw, which are true? (Choose all that apply.)

- A. Compilation fails
- B. A ClassCastException might be thrown
- C. A StackOverflowError might be thrown
- D. A NullPointerException might be thrown
- E. An IllegalStateException might be thrown
- F. The program might hang without ever completing
- G. The program will always complete without exception

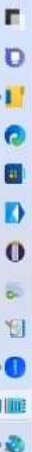
Answer: D,F

I

Ln: 192, Col: 1

20°C  
Cloudy

Q Search



90%

Windows [CTRL]

| UTM: 8

ENG  
IN  
22:57  
13.12.2022