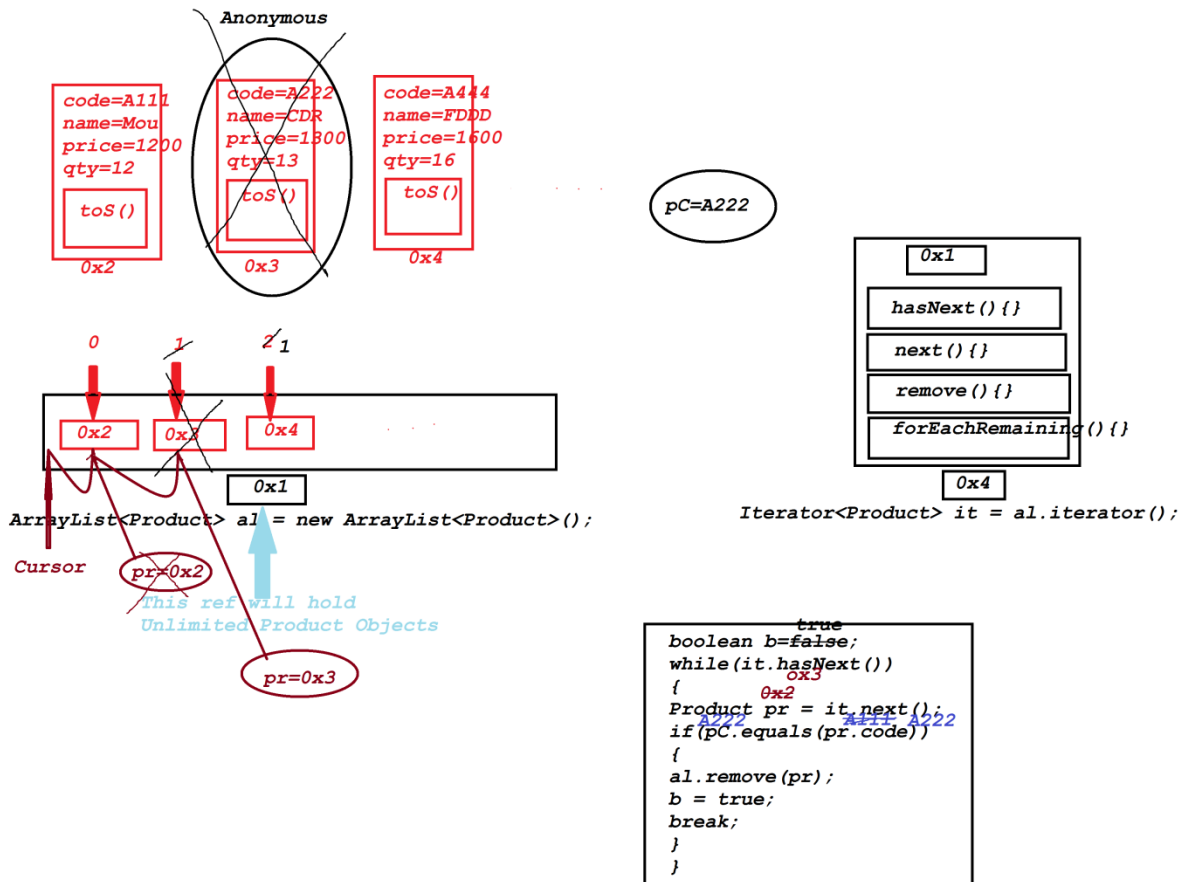


Dt : 7/10/2023

Diagram:



faq:

define sort() method?

=>sort() method is used to perform sorting process on List<E> objects
and which is introduced by Java8 version.

Method Signature:

```
public default void sort(java.util.Comparator<? super E>);
```

Ex:

p1 : Product.java

```
package p1;
public class Product extends Object
{
    public String code,name;
    public float price;
    public int qty;
    public Product(String code,String name,float
price,int qty)
    {
        this.code=code;
        this.name=name;
        this.price=price;
        this.qty=qty;
    }
    @Override
    public String toString()
    {
        return code+"\t"+name+"\t"+price+"\t"+qty;
    }
}
```

p1 : SortByCode.java

```
package p1;
import java.util.*;
@SuppressWarnings("rawtypes")
public class SortByCode implements Comparator
{
    public int compare(Object o1,Object o2)
    {
```

```

        Product p1 = (Product)o1;
        Product p2 = (Product)o2;
        int z = p1.code.compareTo(p2.code);
        if(z==0) return 0;
        else if(z>0) return 1;
        else return -1;
    }
}

```

p1 : SortByName.java

```

package p1;
import java.util.*;
@SuppressWarnings("rawtypes")
public class SortByName implements Comparator
{
    public int compare(Object o1, Object o2)
    {
        Product p1 = (Product)o1;
        Product p2 = (Product)o2;
        int z = p1.name.compareTo(p2.name);
        if(z==0) return 0;
        else if(z>0) return 1;
        else return -1;
    }
}

```

p1 : SortByPrice.java

```

package p1;
import java.util.*;
@SuppressWarnings("rawtypes")
public class SortByPrice implements Comparator
{
    public int compare(Object o1, Object o2)
    {
        Product p1 = (Product)o1;
        Product p2 = (Product)o2;
        if(p1.price==p2.price) return 0;
    }
}

```

```

        else if(p1.price>p2.price) return 1;
        else return -1;
    }
}

```

p1 : SortByQty.java

```

package p1;
import java.util.*;
@SuppressWarnings("rawtypes")
public class SortByQty implements Comparator
{
    public int compare(Object o1, Object o2)
    {
        Product p1 = (Product)o1;
        Product p2 = (Product)o2;
        if(p1.qty==p2.qty) return 0;
        else if(p1.qty>p2.qty) return 1;
        else return -1;
    }
}

```

p2 : DemoList2.java(MainClass)

```

package p2;

import p1.*;

import java.util.*;

public class DemoList2 {

    @SuppressWarnings("unchecked")

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        try(s){

```

```

try {

    ArrayList<Product> al =

        new ArrayList<Product>();

    System.out.println("Enter the number of Products:");

    int n = Integer.parseInt(s.nextLine());

    System.out.println("Enter "+n+" ProductDetails:");

    for(int i=1;i<=n;i++) {

        System.out.println("===ProductsDetails-"+i+"===");

        System.out.println("Enter the ProdCode:");

        String pC = s.nextLine();

        System.out.println("Enter the ProdName:");

        String pN = s.nextLine();

        System.out.println("Enter the ProdPrice:");

        float pP = Float.parseFloat(s.nextLine());

        System.out.println("Enter the ProdQty:");

        int pQ = Integer.parseInt(s.nextLine());

        al.add(new Product(pC,pN,pP,pQ));

        System.out.println("Product Added to List...");

    }//end of loop

    System.out.println("****AllProducts****");

    al.spliterator().forEachRemaining((k)->

    {

```

```
        System.out.println(k.toString());
    });

    while(true) {

        System.out.println("****Choice****");

        System.out.println("\t1.SortByCode"
            + "\n\t2.SortByName"
            + "\n\t3.SortByPrice"
            + "\n\t4.SortByQty"
            + "\n\t5.Exit");

        System.out.println("Enter the Choice:");

        switch(Integer.parseInt(s.nextLine()))
        {

        case 1:

            al.sort(new SortByCode());

            System.out.println("****SortByCode****");

            al.spliterator().forEachRemaining((k)->
            {

                System.out.println(k.toString());

            });

            break;

        case 2:

            al.sort(new SortByName());
```

```
        System.out.println("****SortByName****");
al.splitterator().forEachRemaining((k)->
{
    System.out.println(k.toString());
});
    break;
case 3:
    al.sort(new SortByPrice());
    System.out.println("****SortByPrice****");
al.splitterator().forEachRemaining((k)->
{
    System.out.println(k.toString());
});
    break;
case 4:
    al.sort(new SortByQty());
    System.out.println("****SortByQty****");
al.splitterator().forEachRemaining((k)->
{
    System.out.println(k.toString());
});
    break;
```

case 5:

System.out.println("Program Stopped...");

System.exit(0);

default:

System.out.println("Invalid choice...");

}//end of switch

}//end of loop

}catch(Exception e) {e.printStackTrace();}

}//end of try with resource

}

}

o/p:

Enter the number of Products:

3

Enter 3 ProductDetails:

===ProductsDetails-1===

Enter the ProdCode:

C123

Enter the ProdName:

Mou

Enter the ProdPrice:

1200

Enter the ProdQty:

12

Product Added to List...

===ProductsDetails-2===

Enter the ProdCode:

A123

Enter the ProdName:

CDR

Enter the ProdPrice:

1000

Enter the ProdQty:

9

Product Added to List...

===ProductsDetails-3===

Enter the ProdCode:

B123

Enter the ProdName:

Fddd

Enter the ProdPrice:

1700

Enter the ProdQty:

22

Product Added to List...

******AllProducts******

C123 Mou 1200.0 12

A123 CDR 1000.0 9

B123 Fddd 1700.0 22

******Choice******

1.SortByCode

2.SortByName

3.SortByPrice

4.SortByQty

5.Exit

Enter the Choice:

1

******SortByCode******

A123 CDR 1000.0 9

B123 Fddd 1700.0 22

C123 Mou 1200.0 12

******Choice******

1.SortByCode

2.SortByName

3.SortByPrice

4.SortByQty

5.Exit

Enter the Choice:

3

******SortByPrice******

A123 CDR	1000.0	9
C123 Mou	1200.0	12
B123 Fddd	1700.0	22

******Choice******

1.SortByCode

2.SortByName

3.SortByPrice

4.SortByQty

5.Exit

Enter the Choice:

2

******SortByName******

A123 CDR	1000.0	9
B123 Fddd	1700.0	22
C123 Mou	1200.0	12

******Choice******

1.SortByCode

2.SortByName

3.SortByPrice

4.SortByQty

5.Exit

Enter the Choice:

4

******SortByQty******

A123 CDR 1000.0 9

C123 Mou 1200.0 12

B123 Fddd 1700.0 22

******Choice******

1.SortByCode

2.SortByName

3.SortByPrice

4.SortByQty

5.Exit

Enter the Choice:

5

Program Stopped...

=====