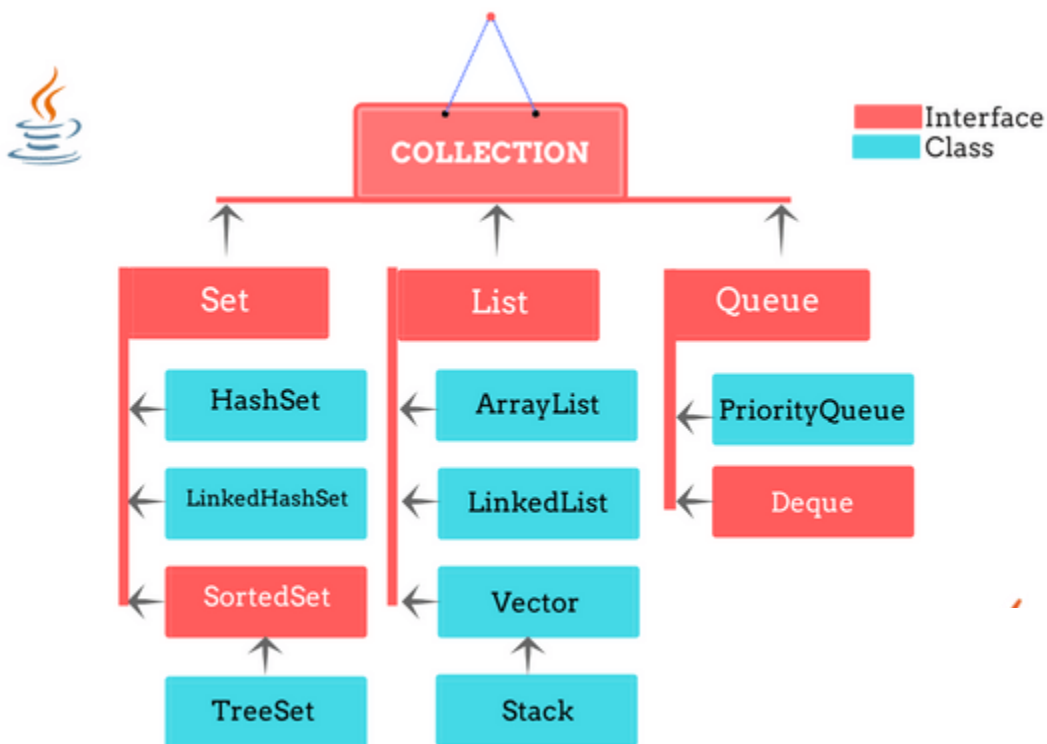


Java Collection

I. ដូចម្តេចទៅដែលហៅថា Collection?

Collection គឺជាប្រភេទ Framework ដែលមានស្រាប់សំរាប់អាចអោយគេផ្ទុកទិន្នន័យបាន ច្រើនប្រភេទលើ ឈ្មោះតែមួយ។ Collection ជាប្រភេទ Dynamic temporary storage ដែលអាចផ្ទុកនូវធាតុ បានច្រើនប្រភេទហើយមិនចាំបាច់កំណត់ទំហំដូច Array នោះទេ។ Collection វាមាននូវ Property, Method ស្រាប់ៗជាច្រើនដែលគ្រាន់រៀនប្រើប្រាស់របស់ទំនោះជាការស្រេច។

លក្ខណៈ Class Framework របស់ Collection ៖



នៅក្នុងចំណុចនៃមេរៀនទាំងនេះយើងលើកតែ ២ ចំណុចមកសិក្សាមួយជា ArrayList(Indexing) និង Hashtable(ជាប្រភេទ HashSet ដែលប្រើប្រាស់នូវ key ដើម្បី Accessing) ។

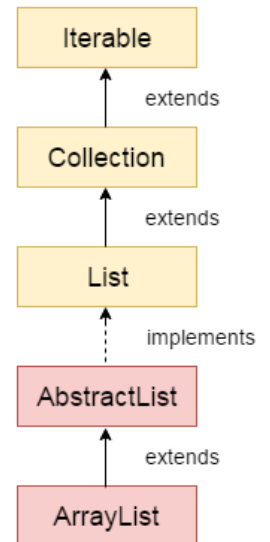
1.1. ArrayList: គឺជាប្រភេទ List

ដែលអាចផ្ទុកទិន្នន័យជាបណ្តោះអាសន្នបានច្រើនប្រភេទ និងច្រើនធាតុថែមទៀត។

ArrayList ត្រូវ Extend ពី AbstractList Class និង Implement ពី interface List។

ចំណុចសំខាន់ៗអំពីថ្នាក់ Java ArrayList គឺ:

- Java ArrayList Class អាចមានធាតុស្ទួន។
- Java ArrayList Class រក្សាលំដាប់នៃការបញ្ចូលធាតុ។
- នៅក្នុង Class Java ArrayList ការរៀបចំគឺយឺតដោយសារមានការផ្លាស់ប្តូរជាច្រើនដែលត្រូវការប្រសិនបើធាតុណាមួយត្រូវបានយកចេញពី List។



ឧទាហរណ៍ ១ : (ការទាញទិន្នន័យចេញពី iterator Class)

```
1
2 import java.util.*;
3 class TestJavaCollection1{
4 public static void main(String args[]){
5 ArrayList<String> list=new ArrayList<String>();//Creating arraylist
6 list.add("ETEC");//Adding object in arraylist
7 list.add("IT");
8 list.add("Professional");
9 list.add("Training");
10 list.add("Center");
11 //Traversing list through Iterator
12 Iterator itr=list.iterator();
13 while(itr.hasNext()){
14 System.out.println(itr.next());
15 }
16 }
17 }
```

ឧទាហរណ៍ ២:

```
1
2 import java.util.*;
3 class TestJavaCollection1{
4 public static void main(String args[]){
5 ArrayList<String> list=new ArrayList<String>();//Creating arraylist
6 list.add("ETEC");//Adding object in arraylist
7 list.add("IT");
8 list.add("Professional");
9 list.add("Training");
10 list.add("Center");
11 System.out.println(list);
12 }
13 }
14
```

ឧទាហរណ៍ ៣:

```
1
2 import java.util.*;
3 class TestJavaCollection1{
4 public static void main(String args[]){
5 ArrayList<String> list=new ArrayList<String>();//Creating arraylist
6 list.add("ETEC");//Adding object in arraylist
7 list.add("IT");
8 list.add("Professional");
9 list.add("Training");
10 list.add("Training");
11 //Traversing List through Iterator
12 for (int i = 0; i < cars.size(); i++) {
13     System.out.println(list.get(i) + "\n");
14 }
15 }
16 }
```

ខាងក្រោមជា Methods ដែលប្រើប្រាស់ជាមួយនិង ArrayList :

- 1) Add(): គឺជាប្រភេទ Method ដាក់ទិន្នន័យបញ្ចូលក្នុង List ឬ អាចអោយគេ Insert ទៅតាមទីតាំង Index មួយច្បាស់លាស់ ។
- 2) Clear(): សំរាប់ធ្វើការសំអាតទិន្នន័យចេញ ពី List
- 3) Containt(): ជា Method សំរាប់ ត្រួតពិនិត្យមើលថាទិន្នន័យមានរឺទេក្នុង List។
- 4) get(): គឺជាប្រភេទ Method ដែលអាចទាញទិន្នន័យចេញពី Collection មកក្រៅតាម Index ។
- 5) set(): គឺជាប្រភេទ method ដែលអាចកែទិន្នន័យតាមរយៈ Index របស់ ArrayList។
- 6) remove(): គឺជាប្រភេទ Method ដែលអាចលុបទិន្នន័យចេញ ពី List ។
- 7) size(): គឺជាប្រភេទ Method ដែលប្រាប់ពីចំនួនធាតុសរុបរបស់ ArrayList
- 8) Sort(): គឺជាប្រភេទ Method ដែលអាចតំរៀបធាតុរបស់ List។

```
1 import java.util.*;
2 class TestJavaCollection1{
3 public static void main(String args[]){
4 Scanner objin=new Scanner(System.in);
5 ArrayList<String> list=new ArrayList<String>();
6 list.add("ETEC");//Adding object in arraylist
7 list.add("IT");
8 list.add("Professional");
9 list.add("Training");
10 list.add("Training");
11 string svalue;
12 int b=0;
13 System.out.print("Input value to Delete=");
14 svalue=objin.next();
15 for (int i = 0; i < cars.size(); i++) {
16     if(svalue.equals(list.get(i)))
17     {
18         System.out.println(list.get(i) + "\n");
19         list.remove(i);
20         System.out.print(svalue+"Remove Completed!");
21         b=1;
22         break;
23     }
24     if(b==0) System.out.print("Error Remove!");
25 }
26 }
```

លំហាត់អនុវត្តន៍

```
1  =====Menu=====
2  1)> Add
3  2)> View
4  3)> Search
5  4)> Update
6  5)> Remove
7  6)> Sort
8  7)> Exit
9
```

អោយគេអាចបញ្ចូល និង ធ្វើការបានច្រើនដង

ការប្រើប្រាស់នូវ ArrayList ជាមួយនិង Class

ឧទាហរណ៍ ៖

```
1
2 import java.util.*;
3 //Creating Class
4 class Student{
5     int rollno;
6     String name;
7     int age;
8     Student(int rollno,String name,int age){
9         this.rollno=rollno;
10        this.name=name;
11        this.age=age;
12    }
13 }
14
```

```
15
16  class ArrayList5{
17  public static void main(String args[]){
18      //Creating user-defined class objects
19      Student s1=new Student(101,"Sok daravan",23);
20      Student s2=new Student(102,"GechLeang",21);
21      Student s2=new Student(103,"Van Serey",25);
22      //creating arraylist
23      ArrayList<Student> al=new ArrayList<Student>();
24      al.add(s1); //adding Student class object
25      al.add(s2);
26      al.add(s3);
27      //Getting Iterator
28      Iterator itr=al.iterator();
29      //traversing elements of ArrayList object
30  while(itr.hasNext()){
31      Student st=(Student)itr.next();
32      System.out.println(st.rollno+" "+st.name+" "+st.age);
33  }
34  }
35  }
```

លំហាត់អនុវត្តន៍

១) តើ Class ខាងលើចូរបង្កើតនូវ Menu Search និង Update បន្ថែមទៅលើ code មានស្រាប់?

Java HashTable

គឺជាប្រភេទ List មួយបែបទៀតដែលការផ្ទុកទិន្នន័យមិនប្រើប្រាស់តាមរយៈ Index ពេលគឺវាប្រើប្រាស់នូវ key ដើម្បីដំណើរការវិញ។ List ប្រភេទនេះគេអាចប្រើប្រាស់នូវ Key អាចជាអក្សរ ឬ លេខគត់ដើម្បីកំណត់វាបាន ។

```
1  
2 Syntax  
3  
4 public class Hashtable<K,V> extends Dictionary<K,V>  
5     implements Map<K,V>, Cloneable, Serializable
```

ឧទាហរណ៍ ១៖

```
1 import java.util.*;  
2 class Hashtable1{  
3     public static void main(String args[]){  
4         Hashtable<Integer,String> hm=new Hashtable<Integer,String>();  
5  
6         hm.put(100,"RUPP");  
7         hm.put(102,"SETEC");  
8         hm.put(101,"ITC");  
9         hm.put(103,"Norton");  
10        hm.put(104,"AEU");  
11        hm.put(105,"NIPTEC");  
12  
13        for(Map.Entry m:hm.entrySet()){  
14            System.out.println(m.getKey()+" "+m.getValue());  
15        }  
16    }  
17 }
```

ឧទាហរណ៍ ២៖

```
1 import java.util.*;
2 class Hashtable1{
3     public static void main(String args[]){
4         Hashtable<Integer,String> hm=new Hashtable<Integer,String>();
5
6         hm.put(100,"RUPP");
7         hm.put(102,"SETEC");
8         hm.put(101,"ITC");
9         hm.put(103,"Norton");
10        hm.put(104,"AEU");
11        hm.put(105,"NIPTEC");
12
13        for(Map.Entry m:hm.entrySet()){
14            System.out.println(m.getKey()+" "+m.getValue());
15        }
16
17        // Output with
18
19        System.out.println("Before remove: "+ hm);
20        // Remove value for key 102
21        hm.remove(102);
22        System.out.println("After remove: "+ hm);
23    }
24 }
```


ឧទាហរណ៍ ៣៖

```
1 import java.util.*;
2 class hashTabledemo {
3     public static void main(String[] arg)
4     {
5         // creating a hash table
6         Hashtable<String, String> h =
7             new Hashtable<String, String>();
8         Hashtable<String, String> h1 =
9             new Hashtable<String, String>();
10        h.put("ETEC1001", "Sok Dara");
11        h.put("ETEC1002", "Va Sythong");
12        h.put("ETEC1003", "Song Tona");
13        // create a clone or shallow copy of hash table h
14        h1 = (Hashtable<String, String>)h.clone();
15        // checking clone h1
16        System.out.println("values in clone: " + h1);
17        // clear hash table h
18        h.clear();
19        // checking hash table h
20        System.out.println("after clearing: " + h);
21    }
22 }
23
```

ខាងក្រោមគឺជា ប្រភេទ Method ដែលមានស្រាប់របស់ Hashtable ដូចជា៖

- ១). Put(): គឺជាប្រភេទ Method ដាក់ទិន្នន័យចូលក្នុង List.
- ២). Getvalue(): គឺជាប្រភេទ Method ទិន្នន័យចេញពី List
- ៣) getKey(): គឺជាប្រភេទ method ទាញ Key ចេញ.
- ៤) clone(): គឺជាប្រភេទ Method Copy ទិន្នន័យពី Object មួយទៅកាន់ Object មួយផ្សេងទៀត។
- ៥) clear(): គឺជា Method សំរាប់ សំអាតទិន្នន័យចេញពី List។
- ៦) remove(): សំរាប់លុបធាតុចេញពី List.



៧) get(): សំរាប់ទាញទិន្នន័យចេញ ពី List តាមរយៈ Key ។

៨) Containts(): សំរាប់ស្វែងរកតំលៃណាមួយក្នុង List ។

៩) Size(): សំរាប់ប្រាប់ពីទំហំនៃចំនួនធាតុរបស់ List ។

ឧទាហរណ៍ ៤ ៖ ការប្រើប្រាស់ Class ជាមួយនឹង Hashtable

```
1 import java.util.*;
2 class Book {
3     int id;
4     String name,author,publisher;
5     int quantity;
6     public Book(int id, String name, String author,
7         String publisher, int quantity) {
8         this.id = id;
9         this.name = name;
10        this.author = author;
11        this.publisher = publisher;
12        this.quantity = quantity;
13    }
14 }
15
16
```



```

1 public class HashtableExample {
2     public static void main(String[] args) {
3         //Creating map of Books
4         Map<Integer,Book> map=new Hashtable<Integer,Book>();
5         //Creating Books
6         Book b1=new Book(101,"C/C++","Kro IT Chet Laor","ETEC1",8);
7         Book b2=new Book(102,"HTML,CSS, Boostratpr",
8             "Kro IT Chet Laor","ETEC2",4);
9         Book b3=new Book(103,"Java Programming","Kro IT Chet Laor","ETEC3",6);
10        //Adding Books to map
11        map.put(1,b1);
12        map.put(2,b2);
13        map.put(3,b3);
14        //Traversing map
15        for(Map.Entry<Integer, Book> entry:map.entrySet()){
16            int key=entry.getKey();
17            Book b=entry.getValue();
18            System.out.println(key+" Details:");
19            System.out.println(b.id+" "+b.name+" "+b.author+" "+b.publisher+" "
20                +b.quantity);
21        }
22    }
23 }

```

លំហាត់អនុវត្តន៍

- 1) ចូរបង្កើត Class មួយដែលអាចផ្ទុកទិន្នន័យ ពាក្យ Word(String), Speech(String) និង Description(String) បន្ទាប់មកបង្កើត Object អោយគេអាច Add, Output, Search និង update ពាក្យទាំងនោះ? (ប្រើប្រាស់នូវ HashTable)