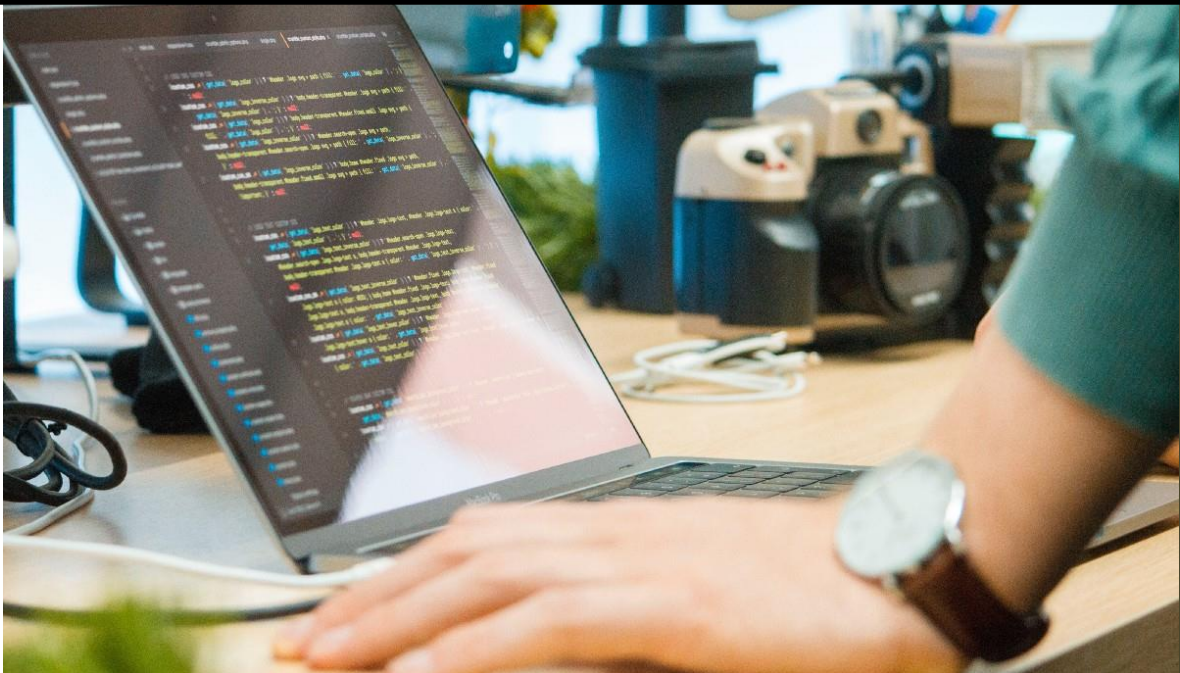




Build Your IT Skill

១) ណែនាំស្គាល់ពី Array Java

២) ណែនាំស្គាល់ពី Collection Java



វៀរវាងនិយោជក

គ្រូរោងចក្របច្ចេកទេស

អនុបណ្ឌិត RUPP, TKU

(Software

Engineering)

២០២២

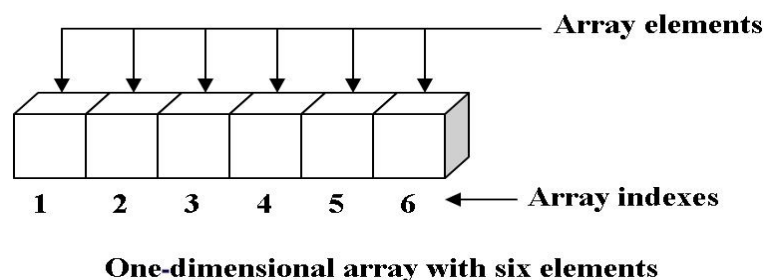
Java Array

I. ជូនម្ដេចទៅដែលហៅថា Array?

Array សំដៅលើប្រភេទអញ្ញាតិពិសេសមួយដែលអាចមានលទ្ធភាពធ្វើការ Store ទិន្នន័យបានច្រើនស្ថិតក្នុងប្រភេទទិន្នន័យរួមគ្នាមួយហើយមានលក្ខណៈបណ្តុះអាសន្នពេលគឺវាផ្ទុកនៅលើ RAM។ Array ក្នុង Java ត្រូវបានគេបែងចែកជា ៣ប្រភេទគឺ ៖

១). Array One Dimensional(Array មួយវិមាត្រ)

Array 1D គឺជាប្រភេទ Array ដែល Store ទុកទិន្នន័យមានលក្ខណៈ ជា Records ដែលយើងអាច Accessing វាបានទៅតាម Index នៃ Record នីមួយៗរបស់វា។



ឧទាហរណ៍ ១៖

```

Start Page DemoArray.java * x
class DemoArray{
public static void main(String args[]){
    int a[]=new int[5]; //declaration and instantiation
    a[0]=10; //initialization
    a[1]=20;
    a[2]=70;
    a[3]=40;
    a[4]=50;
    //printing array
    for(int i=0;i<a.length;i++) //length is the property of array
        System.out.println(a[i]);
}
}
    
```

លទ្ធផលទទួលបាន៖

```

General Output
-----Configurati
10
20
70
40
50

Process completed.
    
```

ឧទាហរណ៍ ២៖

```
Start Page DemoArray.java X
class DemoArray{
public static void main(String args[]){
    String[] stringArray = new String[5];
    stringArray[0] = "ETEC";
    stringArray[1] = "IT";
    stringArray[2] = "Professional";
    stringArray[1] = "Training";
    stringArray[2] = "Center";
    System.out.println("stringArray output");
    for (int i=0; i<stringArray.length; i++)
    {
        System.out.println(stringArray[i]);
    }
}
```

General Output

```
-----Configurat
stringArray output
ETEC
Training
Center
null
null

Process completed.
```

លទ្ធផលទទួលបាន៖

ឧទាហរណ៍ ៣៖

```
Start Page DemoArray.java X
class DemoArray{
public static void main(String args[]){
    int[] intArray = new int[] {100,500,800,50,60,73,88};
    System.out.println("intArray output (version 2)");
    for (int i=0; i<intArray.length; i++)
    {
        System.out.println(intArray[i]);
    }
}
```

General Output

```
-----Configuratio
intArray output (version 2)
100
500
800
50
60
73
88
```

លទ្ធផលទទួលបាន៖

ឧទាហរណ៍ ៤៖

```
Start Page DemoArray.java x
import java.util.Random;
class DemoArray{
public static void main(String args[]){
    int[] a = new int[100];
    Random ran=new Random();
    int n=20;
    System.out.println("intArray output (version 2)");
    //Random 1-100 with N number
    for (int i=0; i<n; i++){
        {
            a[i]=ran.nextInt(100) + 1;
        }
        //Output Array
        for (int i=0; i<n; i++){
            {
                System.out.print (a[i] + " ");
            }
        }
    }
}
```

លទ្ធផលលទ្ធផលច្បាស់៖

```
General Output
-----Configuration: <Default>-----
intArray output (version 2)
84 74 22 99 88 8 56 33 29 7 66 10 93 74 61 75 45 61 75 96
Process completed.
```



**“តស៊ូការលំបាករៀនបន្តិចទើបលទ្ធផលលទ្ធផល
ច្បាស់មានតំលៃបំផុតក្នុង
ជីវិតអ្នករៀនម្នាក់...”**

—គ្រូអាយធីចិត្តល្អ

លំហាត់អនុវត្តទី១

១). ចូរបង្កើតនូវ Array ចំនួន ៤ ដូចជា Id(String), Name(String) និង Score(double) បន្ទាប់មកអោយគេបញ្ចូលចំនួន N ដងហើយបង្ហាញលទ្ធផលទាំងនោះចេញមកក្រៅវិញ?

លំហាត់អនុវត្តទី២

២) ចូរបង្កើតនូវ Array Score(double) ចំនួន N ធាតុ បន្ទាប់មកចាប់តំលៃចែងនៅចំណុច ១ ដល់ 100 ហើយបោះលទ្ធផលចេញមកក្រៅវិញ ជាមួយនិង TotalScore, Average និង Grade score នីមួយៗតាមលក្ខណៈដូចខាងក្រោម៖

មធ្យមភាព/Average	និទេស/Grade
90-100	A
80-90	B
70-80	C
60-70	D
50-60	E
0-50	F

ឧទាហរណ៍ ៦៖

Start Page DemoArray.java * x

```
import java.util.Random;
import java.util.*;
class DemoArray{
public static void main(String args[]){
    int[] a = new int[100];
    String st;
    Scanner objin=new Scanner(System.in);
    int n=0,op,i;
    do{
        System.out.print("1. Input\n");
        System.out.print("2. Output\n");
        System.out.print("3. Search\n");
        System.out.print("Please Choose One=");
        op=objin.nextInt();
        switch(op){
            {
                case 1:{
                    System.out.print("Input N=");
                    n=objin.nextInt();
                    for(i=0;i<n;i++){
                        System.out.print("Input Array=");
                        a[i]=objin.nextInt();
                    }
                }break;

                case 2:{
                    for(i=0;i<n;i++){
                        System.out.print(a[i]);
                    }
                }break;

                case 3:{
                    int svalue;
                    int b=0;;
                    System.out.println ("Input Value to Search=");
                    svalue=objin.nextInt();
                    for(i=0;i<n;i++){
                        { if(a[i]==svalue)
                            System.out.println ("Search found");
                            b=1;
                            break;
                        }
                    }
                    if(b==0) System.out.println ("Search not found");
                }break;
            }
        }
        System.out.print("Press Yes to Continue...!");
        st=objin.next();
    }while(st.equals("yes"));
    }
}
```

General Output

-----Configuration: <Default>-----
1. Input
2. Output
3. Search
Please Choose One=

លក្ខណៈលក្ខណៈលក្ខណៈ

ឧទាហរណ៍ ៧៖

```

Start Page DemoArray.java X
import java.util.Random;
import java.util.*;
class DemoArray{
    int i;
    Scanner objin=new Scanner(System.in);
    void Input(int a[],int n)
    {
        for(i=0;i<n;i++)
        {
            System.out.print("Input Array=");
            a[i]=objin.nextInt();
        }
    }
    void Output(int a[],int n)
    {
        for(i=0;i<n;i++)
        {
            System.out.print(a[i]+ "  ");
        }
    }
    int search(int a[],int n)
    {
        int svalue;
        int index=-1;
        System.out.println("Input Value to Search=");
        svalue=objin.nextInt();
        for(i=0;i<n;i++)
        { if(a[i]==svalue)
          {
              index=i;
              break;
          }
        }
        return index;
    }

    void Update(int a[],int n)
    {
        int nvalue;
        int index=search(a,n);
        if(index!=-1)
        {
            System.out.println(".....Search not found.....");
        }
        else
        {
            System.out.println("Input New Value=");
            nvalue=objin.nextInt();
            a[index]=nvalue;
            System.out.println("Update Completed.....!");
        }
    }

    void Delete(int a[],int n)
    {
        int index=search(a,n);
        if(index!=-1)
        {
            System.out.println(".....Search not found.....");
        }
        else
        {
            for(i=index;i<n;i++)
            {
                a[i]=a[i+1];
            }
            n=n-1;
            Output(a,n);
        }
    }
}

```

```
void Sort(int a[],int n)
{
    int t,j;
    for(i=0;i<n;i++)
    {
        for(j=i+1;j<n;j++)
        {
            t=a[i];
            a[i]=a[j];
            a[j]=t;
        }
    }
}

public DemoArray()
{
    int[] a = new int[100];
    String st;

    int n=0,op,i;
    do{
        System.out.print("1. Input\n");
        System.out.print("2. Output\n");
        System.out.print("3. Search\n");
        System.out.print("4. Update\n");
        System.out.print("5. Delete\n");
        System.out.print("6. Sort\n");
        System.out.print("Please Choose One=");
        op=objin.nextInt();

        switch(op)
        {
            case 1:{
                System.out.print("Input N=");
                n=objin.nextInt();
                Input(a,n);
            }break;

            case 2:{
                Output(a,n);
            }break;
            case 3:{
                int index=search(a,n);
                if(index!=-1)
                    System.out.println ("Search not found");
                else
                    System.out.println ("Search found at " + index);
            }break;
            case 4:{
                Update(a,n);
            }break;
            case 5:{
                Delete(a,n);
            }break;
            case 6:{
                Sort(a,n);
            }break;
        }
        System.out.print("\nPress Yes to Continue...!");
        st=objin.next();
    }while(st.equals("yes"));
}

public static void main(String args[]){
    new DemoArray();
}
```


២). Array Two Dimensional(Array ពីរទំហំ)

គឺជាប្រភេទ Array ដែលអាចផ្ទុកទិន្នន័យមានលក្ខណៈជា Row និង Column ពេល វាជាតារាងទិន្នន័យមួយដែលអ្នកអាច Access ទិន្នន័យបាន តាមរយៈ Index របស់ Row និង index Column។

	Column 0	Column 1	Column 2	Column 3
Row 0	a[0][0]	a[0][1]	a[0][2]	a[0][3]
Row 1	a[1][0]	a[1][1]	a[1][2]	a[1][3]
Row 2	a[2][0]	a[2][1]	a[2][2]	a[2][3]

ឧទាហរណ៍ ១៖

```

Start Page DemoArray.java * x
import java.util.*;
class DemoArray{
public static void main(String args[]){
    int i,j;
    String[][] names = { {"Sam", "Smith"},
                          {"Robert", "Delgro"},
                          {"James", "Gosling"},
                          };

    //i read row
    for(i=0;i<3;i++)
    { //j Read Col
        for(j=0;j<2;j++)
        {
            System.out.print(names[i][j]);
        }
        System.out.println ("");
    }
}
}

```

លទ្ធផលទទួលបាន៖

```

General Output
-----Configurati
SamSmith
RobertDelgro
JamesGosling

Process completed.

```

ឧទាហរណ៍ ២៖

```

Start Page DemoArray.java x
import java.util.*;
class DemoArray{
public static void main(String args[]){
    int[][] board = new int[3][3];
    for (int i = 0; i < board.length; i++)
    {
        for (int j = 0; j < board[i].length; j++)
        {
            board[i][j] = i + j;
        }
    }
    for (int[] a : board) {
        for (int i : a) {
            System.out.print(i + "\t");
        }
        System.out.println("\n");
    }
}
}

```

លទ្ធផលទទួលបាន៖

General Output			
-----Conf			
0	1	2	
1	2	3	
2	3	4	



“រៀនអត់ដឹងទេវណា, រៀនអត់ផែនការ, រៀនអត់
សកម្មភាព , រៀនអត់ពេលវេលាច្បាស់ទេ
និងអត់អនុវត្តន៍អីសោះ អ្នកស្ទើរចេះអោយសោះ...”
_គ្រូអោយចិច្ចល្អ

ឧទាហរណ៍ ៣៖

```
Start Page DemoArray.java x
import java.util.*;
class DemoArray{
public static void main(String args[]){
    int k=0;
    int row, col, i, j;
    int arr[][] = new int[10][10];
    Scanner scan = new Scanner(System.in);
    System.out.print("Enter Number of Row for Array (max 10) : ");
    row = scan.nextInt();
    System.out.print("Enter Number of Column for Array (max 10) : ");
    col = scan.nextInt();
    System.out.print("Enter " + (row*col) + " Values : \n");
    for(i=0; i<row; i++)
    {
        for(j=0; j<col; j++)
        {
            arr[i][j] = scan.nextInt();
            k=k+arr[i][j];
        }
    }
    for(i=0; i<row; i++)
    {
        for(j=0; j<col; j++)
        {
            System.out.print(arr[i][j]+ " ");
        }
        System.out.println();
    }
    //Sum Array
    System.out.println("Total is:" +k);
}
}
```

ឧទាហរណ៍ ៤៖

```
Start Page DemoArray.java x
import java.util.*;
class DemoArray{
public static void main(String args[]){
    int[][] source = {
        {1, 2, 3, 4},
        {5, 6},
        {0, 2, 42, -4, 5}
    };

    int[][] destination = new int[source.length][];

    for (int i = 0; i < source.length; ++i) {

        // allocating space for each row of destination array
        destination[i] = new int[source[i].length];
        System.arraycopy(source[i], 0, destination[i], 0, destination[i].length);
    }

    // displaying destination array
    System.out.println(Arrays.deepToString(destination));
}
}
```

លទ្ធផលទទួលបាន៖

```
General Output
-----Configuration: <Default>-----
[[1, 2, 3, 4], [5, 6], [0, 2, 42, -4, 5]]

Process completed.
```

លំហាត់អនុវត្តទី១

១) ចូរបង្កើត Array មួយឈ្មោះ Product ជា ពីរវិមាត្រ ដែលមាន ១០ Rows និង 3 Column ហើយបញ្ចូលទិន្នន័យលក្ខណៈ ID, Name, Price ចូលក្នុង Array ទាំងនេះ និងបង្ហាញចេញ មកក្រៅវិញតាមជួរដេក និងជួរឈរច្បាស់លាស់?

លំហាត់អនុវត្តទី២

២) ចូរបង្កើតនូវ Array 2 វិមាត្រអាចអោយគេបញ្ចូលព័ត៌មានចំនួន NRow និង NCol បន្ទាប់មកបង្ហាញធាតុទាំងនោះចេញមកក្រៅវិញតែធាតុណាធាតុគូរ?



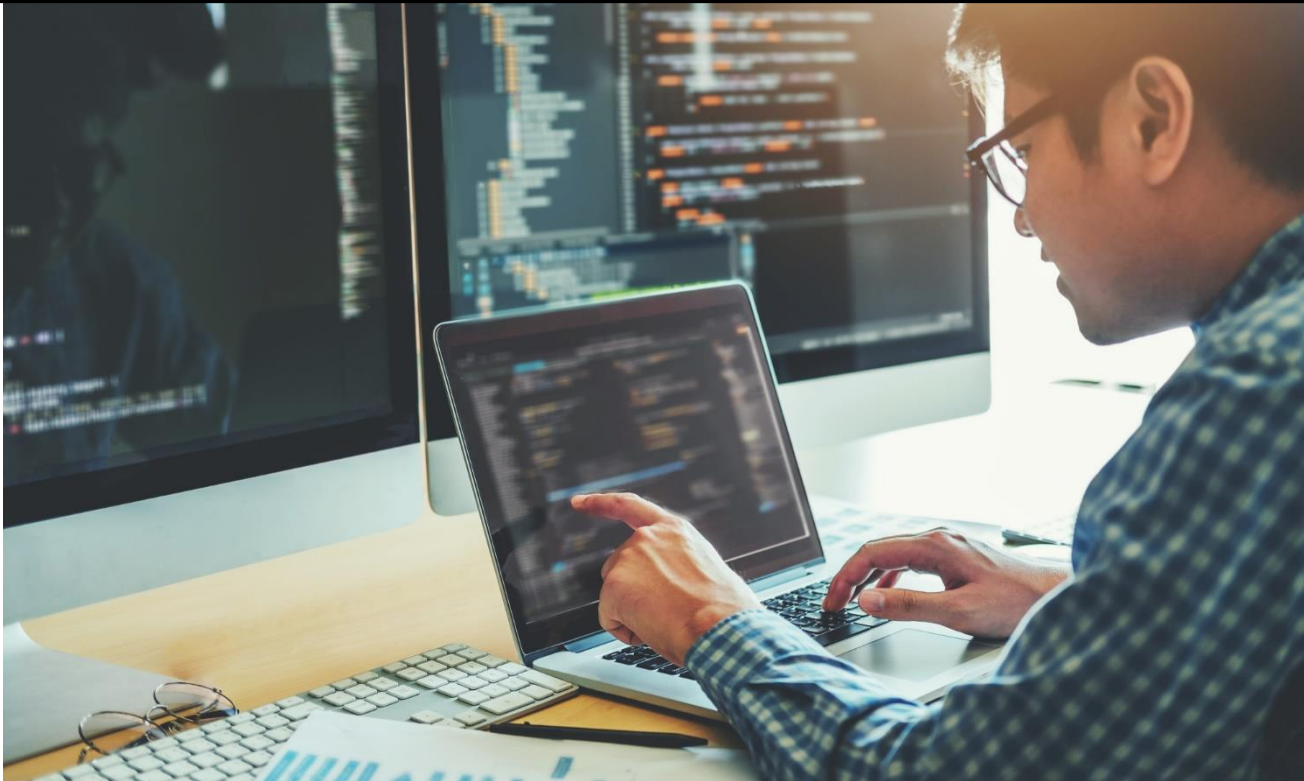
**“មើចង់ចេះក្នុង គ្រូចៀនស្រលាញ់, អនុវត្តន៍
តាមរយៈចេញម្តងមួយបន្ទាត់ តែយូរៗ ទៅ
អាចសរសេរបាន រាប់រយបន្ទាត់អត់ Error សោះក៏
មានដែរ....!”**

—គ្រូអោយចិត្តល្អ



Build Your IT Skill

ណែនាំស្នាមដី Collection Java



វៀបរាងសោយ៖
គ្រូរោងបរិច្ចាគ
អនុបណ្ឌិត RUPP, TKU
(Software Engineering)

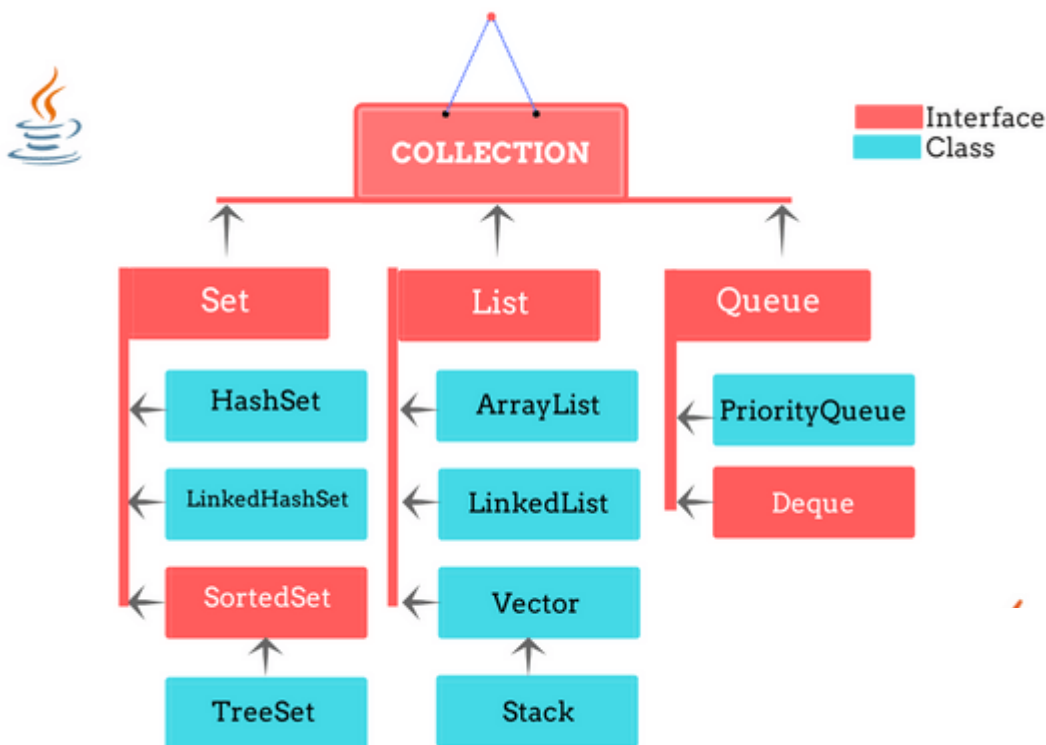
២០២២

Java Collection

II. ជូនម្ដេចទៅដែលហៅថា Collection?

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

លក្ខណៈ Class Framework របស់ Collection ៖

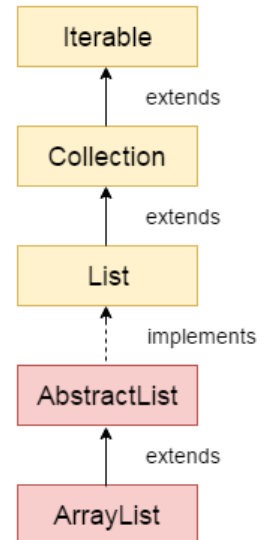


នៅក្នុងចំណុចនៃមេរៀនទាំងនេះយើងលើកតែ ២ ចំណុចមកសិក្សាមួយជា ArrayList(Indexing) និង Hashtable(ជាប្រភេទ HashSet ដែលប្រើប្រាស់នូវ key ដើម្បី Accessing) ។

2.1. ArrayList: គឺជាប្រភេទ List ដែលអាចផ្ទុកទិន្នន័យជាបណ្តោះអាសន្នបានច្រើនប្រភេទ និងច្រើនធាតុថែមទៀត។ ArrayList ត្រូវ Extend ពី AbstractList Class និង Implement ពី interface List។

ចំណុចសំខាន់ៗអំពីថ្នាក់ Java ArrayList គឺ:

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



ឧទាហរណ៍ ១ : (ការប្រើប្រាស់ 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។

ឧទាហរណ៍

```
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    }
```

លំហាត់អនុវត្ត

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

បញ្ជាក់៖ អោយគេអាចបញ្ចូល និង ធ្វើការច្នៃប្រឌិត

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 ខាងលើចូរបន្ថែមនូវ Menu Search និង Update បន្ថែមទៅលើ code មានស្រាប់?



“រៀនក្នុងយុវវ័យ កាន់តែជាក់ សម្បាយជាង
មើលក្នុង ឬ តុក្កតាទៀត, មើមិនជឿទេ
សាករៀនតាមដានក្នុង យុវវ័យ និងលះដូចគ្រូច្នេះ...!”
_គ្រូអាយធីចិត្តល្អ

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 ជាមួយនិង 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, Boostatpr",
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 }
```

លំហាត់អនុវត្ត

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