JAVA BEGINNER

COLLECTIONS FRAMEWORK

Website	http://www.antkh.com
Tel	010 / 016 66 66 53

Prepared By	Tum Sakal
Tel	087 36 31 30
Mail	tumsakal.ts@gmail.com
Facebook	sak kal

JAVA COLLECTIONS

ANT
Abstract of New Technology

- Java Collection គឺជាសំណុំនៃ interfaces និង classes សំរាប់ជំនួយដល់ការគ្រប់គ្រង និងការរក្សាទុកទិន្នន័យក្នុងទំរង់នៃ data structure ណាមួយ។
- អត្ថប្រយោជន៍:
  - ដំនើការលឿនជាង
  - កាត់បន្ថយការសេរក្លូដ:

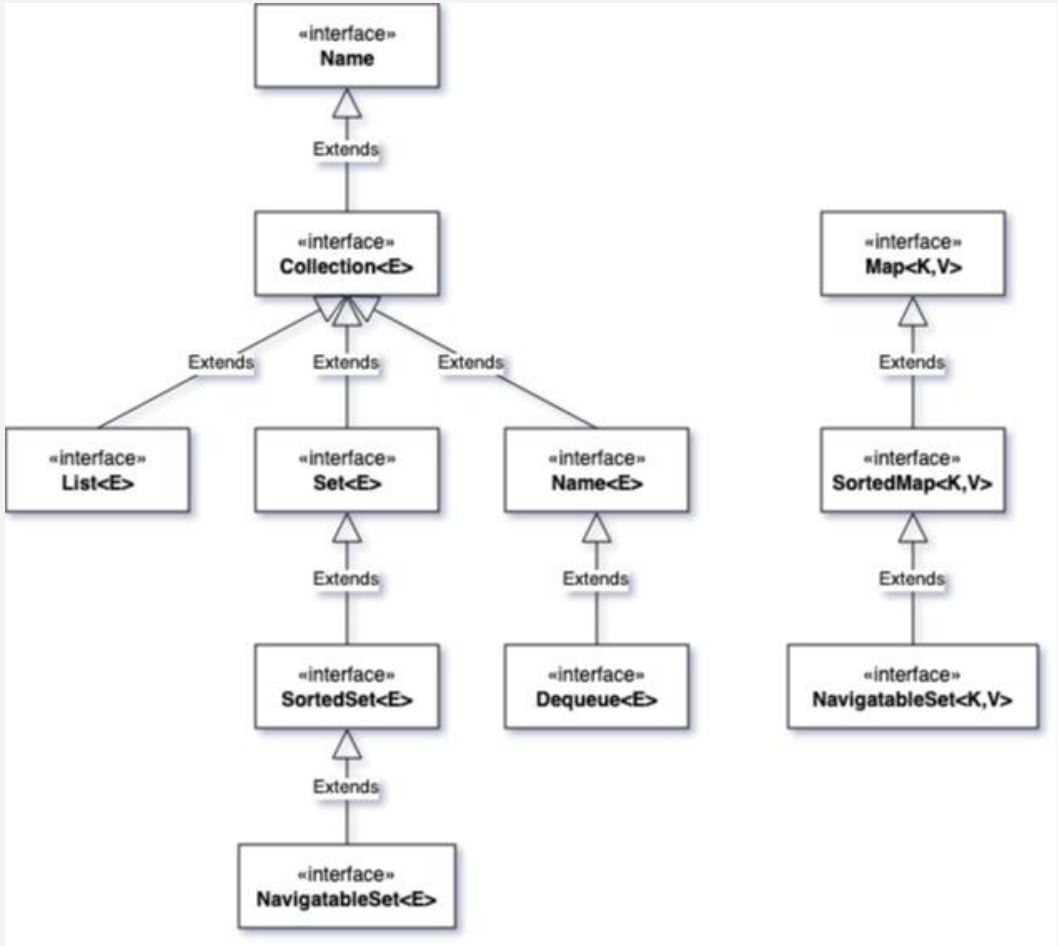
JAVA COLLECTION

- Java Collection បែងចែកជាបីផ្នែកគឺ
  - Interface
  - Implementation
  - និង Algorithm

CORE INTERFACE

ANT
Abstract of New Technology

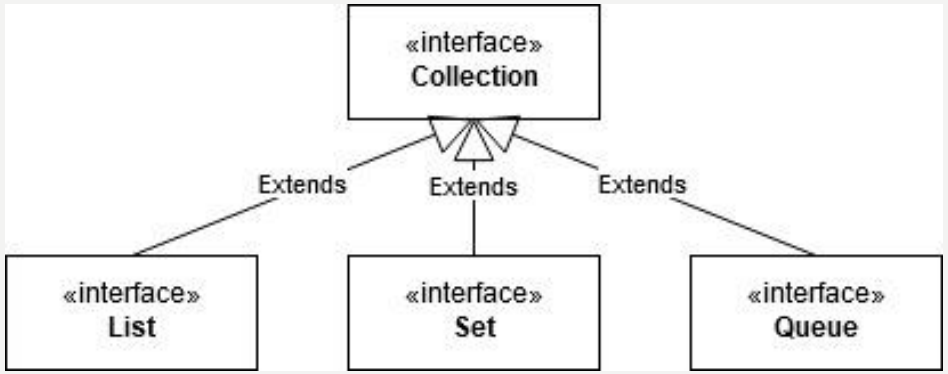
- Java Collections មាន Interfaces សំខាន់ដូចជា
  - Collection<E>
    - List<E>
    - Set<E>
    - Queue<E>
      - Dequeue<E>
  - និង Map<K, V>



COLLECTION<E>

ANT
Abstract of New Technology

- Collection<E> គឺជា Root Interface នៃ
  - List
  - Set
  - Queue



- Collection<E> Interface រក្សាទុកទិន្នន័យក្នុងទំរង់
  - Sequence of elements

data	data	data	data	data

  - Dynamic Size: can growth and shrink

COLLECTION

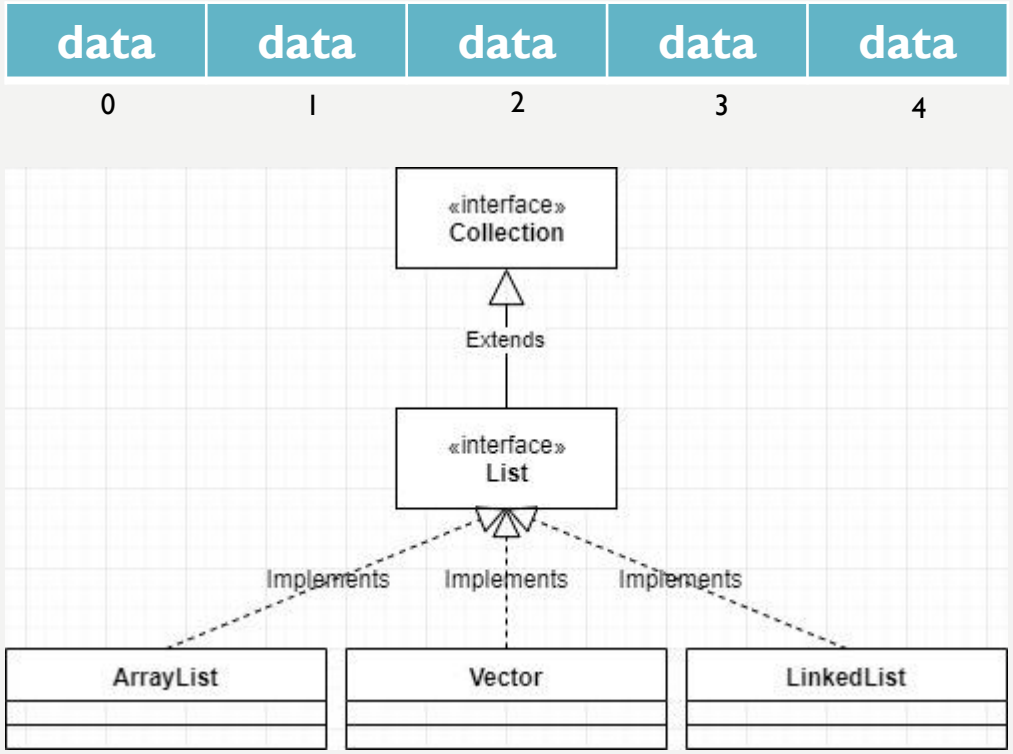
ANT
Abstract of New Technology

- **Return** **Method**
- boolean add(E element)
- boolean addAll(Collection<? extends E> coll)
- void clear()
- boolean contains(E element)
- boolean containsAll(Collection<? extends E> coll)
- boolean isEmpty()
- boolean remove(E element)
- boolean removeAll(Collection<?> coll)
- int size()
- Object[] toArray()
- T[] toArray(T[] array)

LIST<E>

ANT
Abstract of New Technology

- List គឺជា Collection Interface ដែល:
  - រក្សាទុកទិន្នន័យមានលំដាប់(Insertion-Order)
  - អាចផ្ទុកទិន្នន័យស្ទួន
  - អាចផ្ទុកទិន្នន័យទទេរ
  - Zero-Based Index
- Implementation Class: ArrayList, LinkedList



LIST

Abstract of New Technology

Return	Method
boolean	add(int index, E element)
boolean	addAll(int index, Collection<? extends E> coll)
E	get(int index)
int	indexOf(Object obj)
int	lastIndexOf(Object obj)
E	remove(int index)
E	set(int index, E element)
List<E>	subList(int fromIndex, int toIndex)

ARRAYLIST

Abstract of New Technology

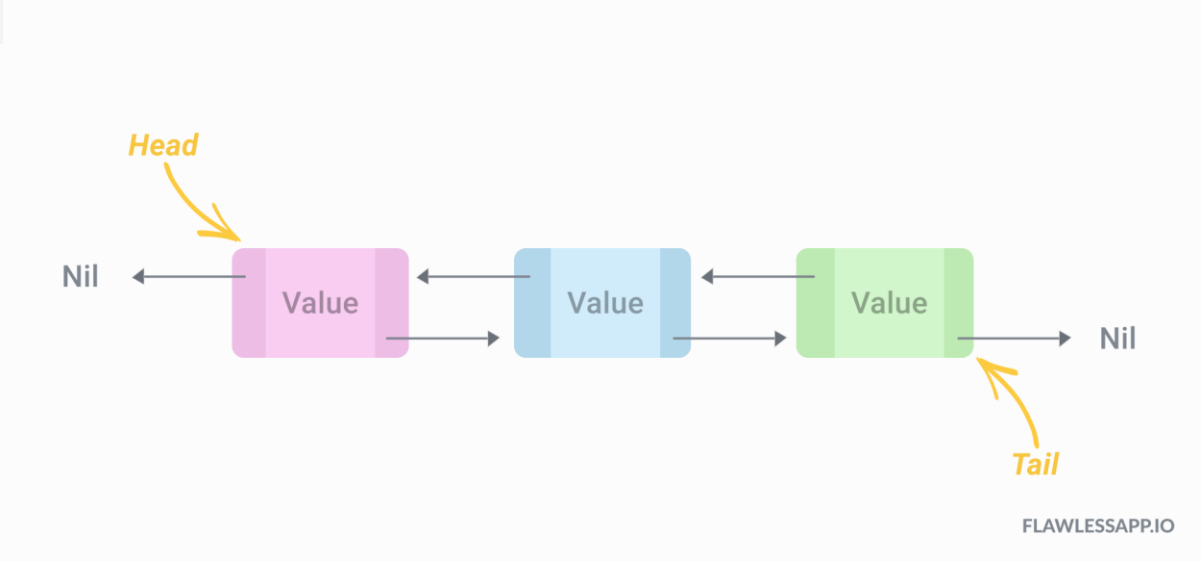
- ArrayList គឺជាប្រភេទ List ដែលរក្សាទុកទិន្នន័យក្នុងទំរង់ Resizable Array មានន័យថាជាប្រភេទ Array ដែលមានចំនួនធាតុ អាចកើន អាចថយបាន នៅពេលដែលយើងបន្ថែម រឺលុបធាតុចេញពីវា

```
List<E>      objectName =  new ArrayList<>();
ArrayList<E> objectName =  new ArrayList<>();
```

LINKEDLIST

ANT
Abstract of New Technology

- LinkedList គឺជាប្រភេទ List ដែលរក្សាទុកទិន្នន័យក្នុងទំរង់ Doubly Linked List



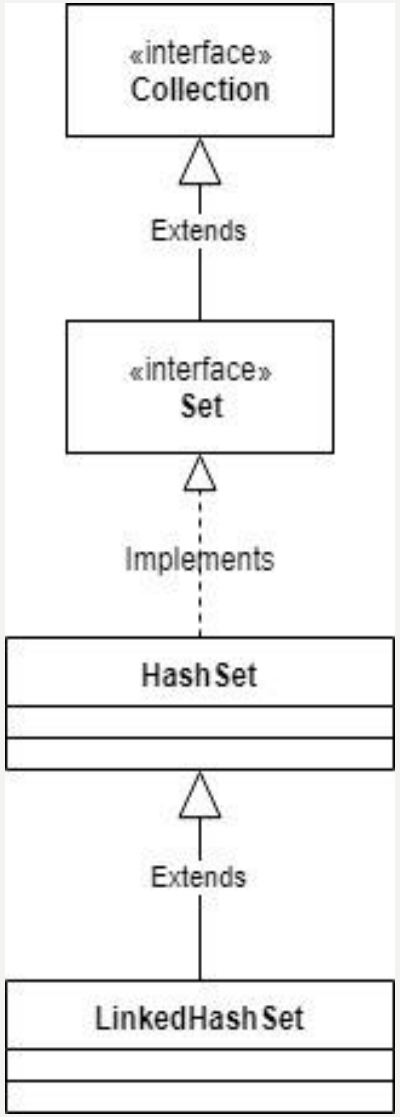

```
List<E>       objectName =  new LinkedList<>();

LinkedList<E> objectName =  new LinkedList<>();
```

SET<E>

- Set ធ្វើការ Inherit Collection Interface ទាំងស្រុង:
- លក្ខណៈរបស់ Set:

data	data	data	data	data

  - រក្សាទុកទិន្នន័យមិនស្ងួន
  - មិនប្រើប្រាស់ Index
- Implementation Class:
  - HashSet<E>:
  - LinkedHashSet<E>:



HASHSET

- HashSet<E> គឺជា ប្រភេទ Set ដែលរក្សាទុកទិន្នន័យក្នុងទំរង់ Hashtable

```
HashSet<E> objectName = new HashSet<>();

Set<E>     objectName = new HashSet<>();
```

HASHSET

Abstract of New Technology

```
Set<Integer> set = new HashSet<>();
```

```
set.add(10);
```

Size = 1

```
set.add(20);
```

Size = 2

```
set.add(30);
```

Size = 3

```
set.add(40);
```

Size = 4

```
set.add(50);
```

Size = 5

```
set.remove(20);
```

Size = 4

LINKEDHASHSET

ANT
Abstract of New Technology

- LinkedHashSet<E> គឺជា ប្រភេទ Set ដែលរក្សាទុកទិន្នន័យក្នុងទំរង់ Hashtable ជាមួយនិង Doubly Linked List
- LinkedHashSet រក្សាទុកទិន្នន័យគិតលំដាប់នៃការបញ្ចូល

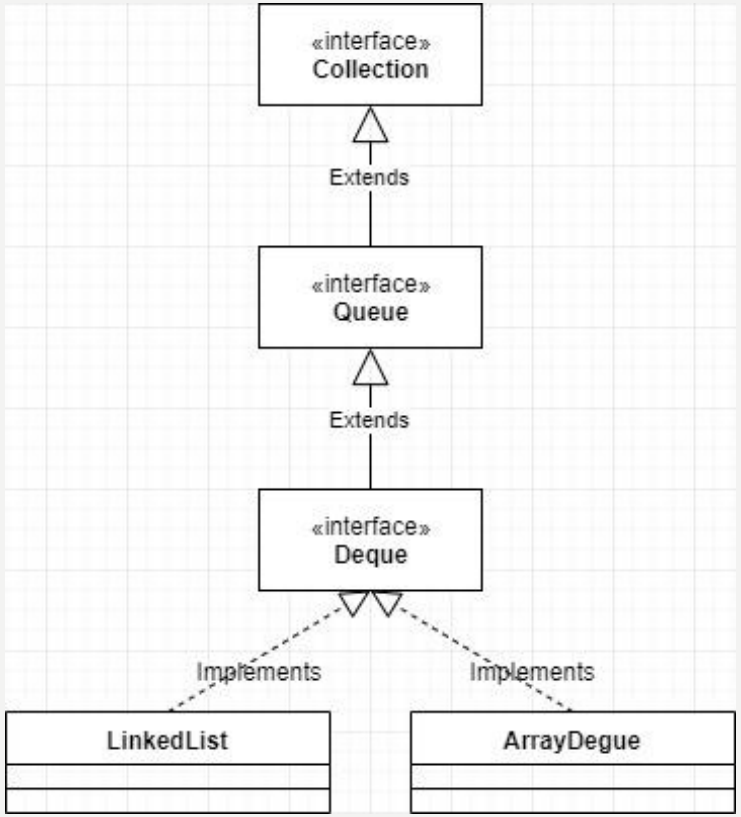
```
Set<E>           objName = new LinkedHashSet<>();
LinkedHashSet<E> objName = new LinkedHashSet<>();
```

QUEUE & DEQUEUE

ANT
Abstract of New Technology

- Queue គឺជា Collection Interface ដែល :
  - រក្សាទុកទិន្នន័យក្នុងទំរង់ F.I.F.O
  - ទិន្នន័យអាចស្ទួន និងទទេរ
  - មិនប្រើ Index
- Deque extends Queue Interface
  - រក្សាទុកទិន្នន័យក្នុងទំរង់ F.I.F.O & L.I.F.O
- Implementation Class:
  - LinkedList និង ArrayDeque

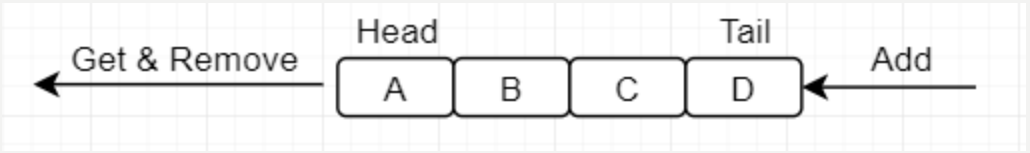
data	data	data	data	data



- Queue – First In First Out:
- បន្ថែមធាតុទៅកាន់ទីតាំងចុងក្រោយ

  boolean add(E e)

  void addLast(E e)



- ដកយកធាតុពីទីតាំងដំបូង

  boolean remove(E e)

  void removeFirst(E e)

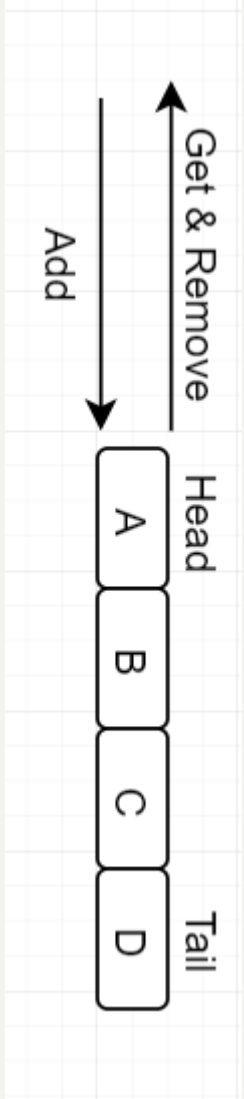
- Stack – Last In First Out:
- បន្ថែមធាតុទៅកាន់ទីតាំងដំបូង

  void push(E e)

  void addFirst(E e)
- ដកយកធាតុពីទីតាំងដំបូង

  E pop()

  E removeFirst()



ARRAYDEQUE

ANT
Abstract of New Technology

- ArrayDeque រក្សាទុកទិន្នន័យក្នុងទំរង់ Array ដែលមិនកំនត់ចំនួនធាតុ ហើយអនុញ្ញាតអោយយើងបញ្ចូល និងលុបធាតុក្នុងទំរង់ណាមួយនៃ L.I.F.O រឺ F.I.F.O

```
Deque<E>      objName =  new ArrayDeque<>();

ArrayDeque<E> objName =  new ArrayDeque<>();
```

LINKEDLIST

ANT
Abstract of New Technology

- LinkedList រក្សាទុកទិន្នន័យក្នុងទំរង់ Doubly Linked List ហើយអនុញ្ញាតអោយយើងបញ្ចូល និងលុបធាតុក្នុងទំរង់ណាមួយនៃ L.I.F.O រឺ F.I.F.O

```
Deque<E>       objectName =  new LinkedList<>();

LinkedList<E> objectName =  new LinkedList<>();
```

MAP<K,V>

ANT
Abstract of New Technology

- លក្ខណៈពិសេសរបស់ Map<K,V> Interface គឺ:
  - រក្សាទុកទិន្នន័យក្នុងទំរង់គូរ Key និង Value
  - ប្រើប្រាស់ Key ជំនួស Index
  - មិនអាចមាន Key ស្ទួន
- Class ដែល Implement Set Interface មាន [illegible]
  LinkedHashMap, …។

Key	Values
Key1	Value1 of Key1
Key2	Value of Key2
…	…
KeyN	Value of KeyN

- **Return** **methodName([parameter])**
- void clear()
- boolean containsKey(Object key)
- boolean containsValue(Object value)
- V get(Object key)
- V getOrDefault(Object key, V defaultValue)
- boolean isEmpty()
- Set<V> keySet
- V put(K key, V value)
- V remove(Object key)
- int size()

ANT
Abstract of New Technology

HASHMAP

ANT
Abstract of New Technology

- HashMap<K,V> គឺជាប្រភេទ Map ដែលរក្សាទុកទិន្នន័យក្នុងទំរង់ Hashtable

```
HashMap<K,V> objectName = new HashMap<>();

Map<K,V>     objectName = new HashMap<>();
```

HASHMAP

Abstract of New Technology

```
Map<Integer, String> numDictionary = new HashMap<>();
numDictionary.put(“one”, 1);
numDictionary.put(“two”, 2);
numDictionary.put(“three”, 3);
numDictionary.put(“four”, 4);
numDictionary.put(“five”, 5);
numDictionary.remove(“two”);
```

“one”	1
“two”	2
“three	3
“four	4
“five”	5