Collection arralist with iteratot

import java.util.\*;  
public class ta5{  
 public static void main(String[] args) {  
 System.*out*.println("Hello ANK");  
 Scanner s = new Scanner(System.*in*);  
 ArrayList<String> a = new ArrayList<>();  
 a.add("Ahmad");  
 a.add("Noor");  
 a.add("Khan");  
 System.*out*.println("Name is "+a);  
 System.*out*.println(" ");  
 System.*out*.println("Is Khan name is present");  
 System.*out*.println(a.contains("Khan"));  
 System.*out*.println("Size of Array is ");  
 System.*out*.println(a.size());  
 System.*out*.println("Remove Khan");  
 System.*out*.println(a.remove("Khan"));  
 System.*out*.println("Now after removing Khan Name is "+a);  
 Iterator<String> iterate=a.iterator();  
 while(iterate.hasNext()){  
 System.*out*.println(iterate.next());  
 }  
 ArrayList<String> c = new ArrayList<>();  
 c.add("ANK");  
 c.add("CEO of Fiducia Software");  
 a.addAll(c);  
 System.*out*.println("Complete Name is "+a);  
 System.*out*.println("Name is " + a);  
 ArrayList<Integer> b = new ArrayList<>();  
 b.add(3841);  
 System.*out*.println("Roll No is " + b);  
 }  
}

* and if we add object datatype instead of string and integer so it will be genrics
* on the above program if we change the word ArrayList into HashSet so it will be into HashSet.

Collection linked list

import java.util.ArrayList;  
import java.util.Collections;  
import java.util.LinkedList;  
import java.util.Scanner;  
  
public class ta5{  
 public static void main(String[] args) {  
 Scanner s = new Scanner (System.*in*);  
 LinkedList<String> khan = new LinkedList<>();  
 khan.add("4");  
 khan.add("6");  
 khan.add("1");  
 khan.addFirst("8");  
 System.*out*.println(khan);  
 khan.addLast("2");  
 System.*out*.println("2 is added at last"+khan);  
 System.*out*.println("1st element is "+khan.getFirst());  
 System.*out*.println("Last element is "+khan.getLast());  
 System.*out*.println("Index 4 is removed "+khan.remove(4));  
 System.*out*.println("Before Sorting"+khan);  
 Collections.*sort*(khan);  
 System.*out*.println("After sorting "+khan);  
 }  
}

* we can change string into object in linked list.

For multiple data types arraylist and hashset.

import java.util.ArrayList;  
import java.util.HashSet;  
import java.util.Iterator;  
public class ta5{  
 public static void main(String[] args) {  
 HashSet<Object>hs=new HashSet<>();  
 hs.add("taha");  
 hs.add("jawaid");  
 hs.add(67);  
 hs.add(75);  
 hs.add(88);  
 Iterator<Object>it=hs.iterator();  
 while(it.hasNext()){  
 Object obj=it.next();  
 System.*out*.println(obj);  
 }  
 } }

Generics

public class Gen <T> {  
 T a;  
 Gen(T a1){  
 a= a1;  
 }  
 void get() {  
 System.*out*.println(a);  
 }  
public static void main(String[] args) {  
  
 Gen<Integer> b = new Gen<Integer>(2);  
  
 b.get();  
 }}

for two variables

class Gen <T , V> {  
 T a;  
 V b;  
 Gen(T a1 , V b1){  
 a= a1;  
 b= b1;  
 }  
 void get() {  
 System.*out*.println(a.getClass().getName()+a);  
 System.*out*.println(b.getClass().getName()+b);  
 }  
public static void main(String[] args) {  
  
 Gen<Integer , Double> b = new Gen<>(2 , 2.0);  
  
 b.get();  
 }  
 }

Bounded type generics

public class ta4{  
 public static <T extends Number> double div(T d1,T d2){  
 if(d2.doubleValue()==0){  
 throw new ArithmeticException("Division by zero"); }  
 return d1.doubleValue()/d2.doubleValue(); }  
 public static <T extends Number> double sub(T d1,T d2){  
 return d1.doubleValue()-d2.doubleValue(); }  
 public static <T extends Number> double add(T d1,T d2){  
 return d1.doubleValue()+d2.doubleValue(); }  
 public static <T extends Number> double mul(T d1,T d2){  
 return d1.doubleValue()\*d2.doubleValue(); }  
 public static void main(String[] args) {  
 int x=3,y=4;  
 double p=35.0,q=16.0;  
 System.*out*.print("Division = "+*div*(p,y)+" ");  
 System.*out*.println("Subtraction = "+*sub*(p,q));  
 System.*out*.print("Addition = "+*add*(x,y)+" ");  
 System.*out*.println("Multiplication = "+*mul*(x,q));  
 } }