

17. Scheda di lavoro (ArrayList)

23/11/2018

Lista: sequenza di elementi dello stesso tipo, ciascuno dei quali ha un solo precedente ed un solo successore, tranne il primo che non ha precedenti e l'ultimo che non ha successori.

Array → non possono cambiare la propria dimensione: il numero di elementi contenuti viene stabilito al momento della creazione e rimane immutato

Per superare questa limitazione Java mette a disposizione la classe ArrayList, contenuta nel package java.util che permette di rappresentare sequenze di oggetti di lunghezza variabile. Ciascun oggetto in un'istanza di ArrayList viene identificato da un numero intero, detto indice, che ne indica la posizione

ArrayList è una implementazione di List che memorizza gli elementi in un Array, ma che si occupa della gestione dinamica della dimensione in modo trasparente per lo sviluppatore.

ArrayList è quindi simile ad un **array**, le differenze principali sono due:

- La dimensione può variare durante l'esecuzione di un programma
- Gli elementi contenuti sono di un solo tipo: Object.

<https://docs.oracle.com/javase/9/docs/api/java/util/ArrayList.html>

- costruttori
 - ArrayList(): crea un vettore vuoto in cui la capacità iniziale non è specificata (costruttore di default).
 - ArrayList(int initialCapacity): crea un vettore con la capacità iniziale indicata. Si utilizza quando si ha un'idea, anche approssimata della dimensione massima che la lista raggiungerà.
- Inserimento di un elemento → metodo add
- Accedere ad un elemento → metodo get
- Conoscere il numero di elementi contenuti → size

```
1 package pkg4binfo;
2 import java.io.*;
3 import java.util.*;
4 public class Main {
5     public static void main(String[] args) {
6         ArrayList a = new ArrayList();
7         a.add("ugo");
8         a.add("gavino");
9         a.add("lia");
10        a.add(456);
11        for (int i=0; i<a.size(); i++)
12            System.out.println(a.get(i));
13    }
14 }
```

```
Output - 4Binfo (run)
run:
ugo
gavino
lia
456
```

bad operand types for binary operator "+"
 first type: int
 second type: Object

 (Alt-Enter shows hints)

```

1 package pkg4binfo;
2 import java.io.*;
3 import java.util.*;
4 public class Main {
5     public static void main(String[] args) {
6         ArrayList a = new ArrayList();
7         int s=0;
8         a.add(5);
9         a.add(3);
10        for ( int i=0; i<a.size(); i++) {
11            System.out.println(a.get(i));
12        }
13
14        for ( int i=0; i<a.size(); i++) {
15            s = s + a.get(i);
16        }
17        System.out.println("somma --> " + s);
18    }
19 }
    
```

Devo effettuare una somma fra interi!

casting

```

1 package pkg4binfo;
2 import java.io.*;
3 import java.util.*;
4 public class Main {
5     public static void main(String[] args) {
6         ArrayList a = new ArrayList();
7         int s=0;
8         a.add(5);
9         a.add(3);
10        for ( int i=0; i<a.size(); i++) {
11            System.out.println(a.get(i));
12        }
13
14        for ( int i=0; i<a.size(); i++) {
15            s = s + (int)a.get(i); //casting
16        }
17        System.out.println("somma --> " + s);
18    }
19 }
    
```

Ora ottengo il risultato corretto

```

: Output - 4Binfo (run)
run:
5
3
somma --> 8
BUILD SUCCESSFUL (total time: 0 seconds)
    
```