

Demo004

Il metodo `getTallest()` implementa una riduzione seguita da un filtro, due operazioni fondamentali della programmazione funzionale

La prima fase applica una riduzione sull'array per trovare l'altezza massima. Parte da un valore iniziale (`max=0`) e lo aggiorna iterativamente confrontando ogni elemento, riducendo l'intero array a un singolo valore numerico: l'altezza massima (192). Questo cambia la cardinalità da n elementi a 1 solo valore.

La seconda fase applica un filtro usando l'altezza massima trovata come predicato. Scorre nuovamente l'array e mantiene solo le persone che soddisfano la condizione `p.height==max`, scartando le altre. Il risultato è un sottoinsieme dell'array originale ($m \leq n$ elementi), dove la cardinalità è ridotta ma il tipo rimane `Person`.

Il metodo usa un `HashSet<Person>` come contenitore risultato, che garantisce l'unicità degli elementi. Importante: quando esegui `res.add(p)`, aggiungi al Set un riferimento all'oggetto originale nell'array, non una copia. Le variabili in Java sono puntatori agli oggetti, quindi `res` contiene gli stessi oggetti `Person` già presenti in memoria.

```
public class DemoWithSetList {
    public static void main(String[] args) {
        Person[] people = {
            new Person("Giovanni", 175),
            new Person("Bryton", 172),
            new Person("Carlo", 174),
            new Person("Adriano", 192),
            new Person("Andrea", 192)
        };

        // filtro, entrano n persone, escono m persone
        // con m<=n, escono le più alte
        Set<Person> tallest = getTallest(people);
    }
}
```

```
private static Set<Person> getTallest(Person[] people) {  
    // 1 - trovare la statura più alta  
    int maxHeight = 0;  
    for(Person p : people) {  
        if(p.getHeight() > maxHeight) {  
            maxHeight = p.getHeight();  
        }  
    }  
  
    // 2 - filtrare tutti quelli con quella statura  
    Set<Person> tallestPeople = new HashSet<>();  
    for(Person p : people) {  
        if(p.getHeight() == maxHeight) {  
            tallestPeople.add(p);  
        }  
    }  
  
    return tallestPeople;  
}
```