

Bài 08 COLLECTIONS

Company: DEVPRO VIỆT NAM

Website: devpro.edu.vn

Design by Minh An

Nội dung

- Giới thiệu
- Một số class thông dụng
 - ArrayList
 - LinkedList
 - HashMap
 - ...

Design by Minh An

1. Collections trong java

- Collections cung cấp một bộ các class để làm việc với một nhóm nhiều các đối tượng (danh sách).
- Các lớp này cung cấp các phương thức để xử lý danh sách: thêm mới, sửa, xóa, chèn thêm, thay thế, sắp xếp, tìm kiếm, ...
- Collections khắc phục nhược điểm của mảng là cố định số lượng các phần tử mảng khi chương trình đang được thực thi.

Design by Minh An

2. ArrayList class

- ArrayList hỗ trợ mảng động (có thể thêm hay bớt số phần tử khi chương trình đang thực thi).
- Khởi tạo (Constructor)

```
ArrayList[<ClassName>] list =  
    new ArrayList([<ClassName>]);
```

- Ví dụ:

```
ArrayList list_1 = new ArrayList();  
ArrayList<Person> list =  
    new ArrayList(<Person>);
```

Design by Minh An

2.1. Một số phương thức xử lý ArrayList

`list.size()`: trả về số lượng phần tử của list

`list.add(obj)`: thêm đối tượng obj vào cuối list

`list.add(i, obj)`: chèn đối tượng obj vào vị trí có chỉ số i trong list

`list.addAll(collection)`: Thêm collection vào cuối list

`list.addAll(i, collection)`: Chèn collection vào list tại vị trí chỉ số i

`list.clear()`: Xóa mọi phần tử trong list

`list.clone()`: Trả về một đối tượng là nhân bản của list

`list.contains(obj)`: Trả về true nếu obj thuộc list, ngược lại trả về false

`list.get(i)`: Trả về phần tử tại vị trí chỉ số i

Design by Minh An

Một số phương thức xử lý ArrayList



`list.indexOf(obj)`: Trả về chỉ số tại vị trí đầu tiên chứa obj, nếu không có obj trong list trả về - 1

`list.lastIndexOf(obj)`: Trả về chỉ số tại vị trí cuối cùng chứa obj, nếu không có obj trong list trả về - 1

`list.remove(i)`: Xóa phần tử tại vị trí chỉ số i

`list.set(i, obj)`: Thay thế phần tử tại vị trí chỉ số i bằng obj.

Design by Minh An

2.2. Duyệt ArrayList



- Sử dụng for (thường) với list.size() và list.get(i)
- Sử dụng for each
- Sử dụng Iterator

```
Iterator<Emp> itor = list.iterator();
while (itor.hasNext()) {
    itor.next().display();
}
```

Design by Minh An

Sort ArrayList by Collections.sort()



- Sắp danh sách nhân sự theo lương

```
public static sort(ArrayList<Emp> list){
    Collections.sort(list, new Comparator<Emp>(){
        @Override
        public int compare(Emp e1, Emp e2){
            if (e1.getSalary() > e2.getSalary()){
                return 1;
            }
            else if (e1.getSalary() == e2.getSalary()){
                return 0;
            }
            else{
                return -1;
            }
        }
    });
}
```

Design by Minh An

Ví dụ - Sử dụng ArrayList



Cho thông tin về học viên gồm: Mã học viên, họ đệm, tên, giới tính, ngày sinh, tên khóa học tham gia.

Yêu cầu:

- Tạo danh sách học viên (lưu trữ bằng ArrayList – sử dụng phương thức add để thêm học viên vào ds).
- Sử dụng các phương thức của ArrayList thực hiện các thao tác xử lý danh sách học viên: thêm, chèn, sửa, xóa, tìm kiếm, sắp xếp, v.v...

Design by Minh An

3. LinkedList trong java



- Package: java.util.LinkedList
- LinkedList là một cấu trúc dữ liệu được sử dụng để lưu trữ danh sách tuyến tính gọi là danh sách liên kết.
- LinkedList được sử dụng khi số lượng phần tử trong danh sách cần lưu trữ lớn.
- Khởi tạo LinkedList: tương tự ArrayList.
- Ngoài các phương thức tương tự như ArrayList thì LinkedList có thêm các phương thức:

```
l1list.addFirst(obj);  
l1list.addLast(obj);  
l1list.getFirst();  
l1list.getLast();  
l1list.removeFirst();  
l1list.removeLast();
```

Design by Minh An

Duyệt LinkedList



- Sử dụng for (thường) với list.size() và list.get(i)
- Sử dụng for each
- Sử dụng Iterator

```
Iterator<Students> itor = llist.iterator();  
while (itor.hasNext()) {  
    itor.next().display();  
}
```

Design by Minh An

Sort LinkedList by Collections.sort()



- Sắp danh sách học viên theo tuổi

```
public static sort(LinkedList<Students> list) {  
    Collections.sort(list, new Comparator<Students>() {  
        @Override  
        public int compare(Emp e1, Emp e2) {  
            if (e1.getAge() > e2.getAge()) {  
                return 1;  
            }  
            else if (e1.getAge() == e2.getAge()) {  
                return 0;  
            }  
            else {  
                return -1;  
            }  
        }  
    });  
}
```

Design by Minh An

LinkedList – Ví dụ



Cho thông tin về học viên gồm: Mã học viên, họ đệm, tên, giới tính, ngày sinh, tên khóa học tham gia.

Yêu cầu:

- Tạo danh sách học viên (lưu trữ bằng LinkedList).
- Sử dụng các phương thức của LinkedList thực hiện các thao tác xử lý danh sách học viên: thêm vào đầu, cuối, chèn, sửa, xóa, tìm kiếm, sắp xếp, v.v...

Design by Minh An

4. HashMap



- Là một triển khai của Map Interface
- Lưu trữ dữ liệu dưới dạng cặp key – value
- HashMap chỉ chứa các key duy nhất
- Có thể có 1 key là null, và nhiều value là null
- Duy trì các phần tử không theo thứ tự chèn
- Khởi tạo HashMap: `HashMap(<K, V>)`
- K là kiểu của key, V là kiểu của value
- Ví dụ:

```
HashMap<Integer, String> hashMap =  
    new HashMap<Integer, String>();
```

Design by Minh An

HashMap – Một số phương thức



- `boolean equals(Object obj)`: Trả về true nếu obj có các key và value bằng với các key và value của đối tượng đang gọi
- `Object clone()`: Trả về một đối tượng là nhân bản của HashMap
- `boolean containsKey(Object key)`: Trả về true nếu HashMap chứa element có key bằng với key đã cho
- `Object put(Object key, Object value)`: chèn một element vào map hiện tại.
- `Object get(Object key)`: trả về value ứng với key được chỉ định
- `Object remove(Object key)`: xóa element có key được chỉ định
- `void clear()`: Xóa bỏ tất cả elements
- `boolean containsValue(Object value)`: Trả về true nếu HashMap chứa element có value bằng với value đã cho.

Design by Minh An

HashMap – Một số phương thức



- `Set entrySet()`: Trả về một tập hợp các element
- `Set keySet()`: Trả về đối tượng Set có chứa tất cả các keys
- `boolean isEmpty()`: Trả về true nếu HashMap không chứa element nào
- `Set keySet()`: Trả về tập các key được chứa trong HashMap
- `putAll(Map m)`: Sao chép tất cả elements của map đã xác định vào HashMap.
- `int size()`: Trả về số element chứa trong HashMap
- `Collection values()`: Trả về tập các value của các elements chứa trong HashMap.

Design by Minh An

Duyệt HashMap



- Sử dụng phương thức `keySet()`

```
for (Integer key : hashMap.keySet()) {  
    Student value = hashMap.get(key);  
    System.out.println(key + " = " + value);  
}
```

- Sử dụng phương thức `entrySet()`.

```
for (Entry<Integer, Student> entry : hashMap.entrySet())  
{  
    Integer key = entry.getKey();  
    Student value = entry.getValue();  
    System.out.println(key + " = " + value);  
}
```

Design by Minh An

HashMap – Ví dụ



- Xây dựng từ điển với các yêu cầu
 - Tạo mới từ điển
 - Tra cứu từ trong từ điển
 - Thêm từ
 - Sửa từ trong từ điển
 - Xóa từ
 - ...

Design by Minh An

Bài tập: Sử dụng ArrayList hoặc LinkList



- Chương trình quản lý bán hàng gồm các bảng dữ liệu:

LoaiHang
ID
TenLoai

KhachHang
ID
TenKhachHang

Hang
ID
ID_LoaiHang
TenHang
SoLuong
DonGia

GioHang
ID
ID_KhachHang
Danh_Sach_Hang

Design by Minh An

Bài tập: Sử dụng ArrayList hoặc LinkList



- Cài đặt chương trình thực hiện thao tác:
 - Cập nhật danh sách loại hàng: Xem, Thêm, sửa, xóa, sắp xếp theo tên loại hàng.
 - Cập nhật danh sách hàng (xem, thêm, sửa, xóa, sắp xếp theo tên hàng) trong đó loại hàng chỉ được chọn trong danh sách loại hàng.
 - Cập nhật danh sách khách hàng: Xem, thêm, sửa, xóa, sắp xếp theo tên khách hàng.
 - Tạo 1 giỏ hàng mới cho khách (khách hàng có trong DS khách hàng).
 - Cập nhật giỏ hàng: Thêm, sửa, xóa hàng trong giỏ (hàng trong giỏ phải có trong danh sách hàng. Khi thêm một hàng vào giỏ cần kiểm tra số lượng hàng trong danh sách có đủ để mua hay không? Trừ đi số lượng hàng trong danh sách hàng nếu hàng được mua). Hiển thị chi tiết giỏ hàng (có thông tin tổng số tiền phải thanh toán).
 - Hiển thị danh sách các giỏ hàng.
 - Các chức năng tìm kiếm: Loại hàng, hàng, khách hàng, giỏ hàng.

Design by Minh An

Bài tập: HashMap



- Cài đặt chương trình giới thiệu các địa điểm du lịch:
 - Tạo danh sách ban đầu
 - Cho phép thêm / sửa / xóa địa điểm
 - Cho phép tìm kiếm địa điểm