

## BÀI THỰC HÀNH SỐ 3

### Xử lý ngoại lệ - Generics - Collections

#### I. Mục tiêu

Eff 2 2 2 d2 2 e2e 2 2k lý ngo i l (Exceptions) trong Java, các cách s d ng Java Generics và Collections.

#### II. Thời gian

- E d2a 2e d2 2e 2 e2 f1 22f 2
- f2 2 2e 2 2f 2

#### III. Hướng dẫn chung

##### Phần 1: Xử lý ngoại lệ trong Java.

Câu 1: Sf 2e d2 e2e 2 2 2 d2 2 e2 2e 2 2 2 2 d 2a d2 2 2e 2V 2 2 2a d2khối lệnh try...catch 2 2 2 d f2 2 2e 2 2 2 f2 d2 2e 2 2e 2 2 2 d2e 2 2 2 d2 2e 2 2 2 the d2e f2 2 d 2a d2

Hướng dẫn: 2a d2 2 2 2 2 d f2 2 ab L Lc a B b f 2 2 B b f 2 2 2 2 d f2 2 2 2 ab2 2a 2e 2 2

```
// Tạo một đối tượng Scanner
Scanner scanIn=new Scanner(System.in);
// Tạo mảng số nguyên
int[] soNguyen = new int[5];
int so;
int i =0;
try{
    while(true){
        System.out.print("Nhập số nguyên: ");
        so = scanIn.nextInt();
        soNguyen[i] = so;
        i++;
    }
} catch (ArrayIndexOutOfBoundsException aie){
    System.out.println("Bạn đã nhập quá số lượng phần tử");
}
catch (Exception e){
    System.out.println("Vui lòng nhập số nguyên");
}
//In ra mảng vừa nhập
System.out.println("Mảng vừa nhập");
for (int j = 0; j < soNguyen.length; j++) {
    System.out.println(soNguyen[j]);
}
```

**Câu 2:** Sử dụng exception để kiểm tra điều kiện của mảng

Cho mảng `int[] arr = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}`. Viết một chương trình để kiểm tra xem một giá trị `x` có nằm trong mảng hay không. Nếu `x` nằm trong mảng, hãy in ra giá trị của `x`. Nếu `x` không nằm trong mảng, hãy ném ra một `Exception` với thông điệp "Giá trị `x` không nằm trong mảng".

### Hướng dẫn:

Phương pháp `nextInt()` của lớp `Scanner` sẽ ném ra một `Exception` nếu giá trị nhập vào nằm ngoài phạm vi cho phép. Bạn có thể sử dụng điều này để kiểm tra xem một giá trị có nằm trong phạm vi cho phép hay không.

```
public static void exceptionThrows() throws ArrayIndexOutOfBoundsException, Exception{
    //Tạo mảng 10 phần tử các số nguyên ngẫu nhiên
    int randomNums[] = new int[10];
    Random rand = new Random();
    for (int i = 0; i < randomNums.length; i++) {
        randomNums[i] = rand.nextInt(100);
    }

    //Nhập vào chỉ số mảng
    Scanner scanIn=new Scanner(System.in);
    System.out.print("Nhập vào vị trí phần tử muốn in giá trị: ");
    int index = scanIn.nextInt();
    if (index < 0)
        throw new Exception();
    System.out.println("OK, phần tử mảng thứ " + index + " có giá trị " + randomNums[index]);
}
```

Trong chương trình chính, bạn có thể gọi phương pháp `exceptionThrows()` như sau:

```
public static void main(String[] args) {
    // TODO code application logic here
    while (true){
        try{
            exceptionThrows();
        }
        catch (ArrayIndexOutOfBoundsException e){
            System.out.println("Bạn đã nhập quá chỉ số của mảng.");
            break;
        }
        catch (Exception e){
            System.out.println("Chỉ số index phải lớn hơn 0");
        }
    }
}
```

## Phần 2: Generic và collection:

**Câu 1:** Sử dụng `ArrayList` để lưu trữ các số nguyên. Thêm các số 1, 2, 3, 4, 5 vào danh sách.

Cho danh sách `ArrayList<Integer> list = new ArrayList<>()`. Viết một chương trình để thêm các số 1, 2, 3, 4, 5 vào danh sách.

Hướng dẫn: `Đb bf` `xf` `xuatMang`.

```
public static <T> void xuatMang(T[] arr) {
    for (int i = 0; i < arr.length; i++) {
        System.out.println(arr[i]);
    }
}

public static void main(String[] args) {
    // TODO code application logic here

    Integer[] arrInt = {1,2,3,4};
    String[] arrString = {"A","B","C"};
    Double[] arrDouble = {10.0, 9.0, 8.5, 5.5};
    xuatMang(arrInt);
    xuatMang(arrString);
    xuatMang(arrDouble);
}
```

Câu 2: P `Đ` `aba` `fia` `a` `Đ` `e` `Đ` `e` `f` `e` `Đ` `Đb bf` `e` `e` `e` `Đ` `e` `Đf` `Đ` `Đ` `Đ`

\_\_\_\_\_ `e` `extends`

SA `public static double` `extends`

```
public static double getAverage(List<? extends Number> numList) {
    double total = 0.0;
    for (Number num : numList)
        total += num.doubleValue();
    return total/numList.size();
}

public static void main(String[] args) {
    // TODO code application logic here

    List<Integer> integerList = new ArrayList<Integer>();
    integerList.add(3);
    integerList.add(30);
    integerList.add(60);
    System.out.println(getAverage(integerList));

    List<Double> doubleList = new ArrayList<Double>();
    doubleList.add(3.0);
    doubleList.add(50.0);
    doubleList.add(33.0);
    System.out.println(getAverage(doubleList));
}
```

**Câu 3:** A d Db b f f Mb J db f f a e e P ab I b b

### Hướng dẫn:

Q i P ab e e fa b db e b b ab b P f d

Q i b b e e fa b i e d e b b ab b P f d

Q i Db b f Mb J db e

```
public class PersonManager <T> {

    private ArrayList<T> arr = new ArrayList<T>();

    //Thêm vào ArrayList
    public void add(T obj){
        arr.add(obj);
    }
    //Hiển thị danh sách
    public void display(){
        for (T obj : arr){
            System.out.println(obj);
        }
    }
}
```

f f Mb J db I b b Mb J db P ab e ab b ab e df b b ab

```
public static void main(String[] args) {
    // TODO code application logic here

    PersonManager<Lecturer> lec = new PersonManager<Lecturer>();
    lec.add(new Lecturer(1,"Lecturer 1", 5000.0));
    lec.add(new Lecturer(1,"Lecturer 2", 1000.0));
    lec.display();

    System.out.println("/-----/");

    PersonManager<Student> stu = new PersonManager<Student>();
    stu.add(new Student(1,"Student 1",20));
    stu.add(new Student(1,"Student 2",22));
    stu.display();
}
```

**Câu 4:** He f2 2 2 f 2 2 i 2 f 2he f2 2 f hbaIf 2hf 2a 2f 2 2 P f d2P 2 2

e 2 2 e 2 2 2 2 d2 b2e 2f 2f 2 ab2e 2ef 2 2 2 2 2

Ef 2e2 2e 2 2 d2 f 2 d2 e2 2a d2 b 2

Qe 2 f2 2h 2e 2 2 d2 f 2eb 2 2 2 d2 f 2 e 2 2 2e 2

e 2 2 e 2 2 d2 f 2e e2e 2 2

P 2 2 f 2eb 2e 2 2 d2a 2

2 2 e 2 2 d2 f 2 2 2 2 2 2 2 2 2

2 f 2 2a d2 f b 2

### Hướng dẫn:

Q 2a e2 e2 2 2 f 2 2 i 2 f 2he f2 2 f hbaIf 2

```
List<String> listCars = new LinkedList<String>();
listCars.add("BMW");
listCars.add("Ford");
listCars.add("Chevrolet");
listCars.add("Toyota");
listCars.add("Nissan");
listCars.add("Honda");
listCars.add("Mazda");
listCars.add("Volkswagen");
listCars.add("Volvo");
listCars.add("Mercedes");
listCars.add("Hyundai");
listCars.add("Kia");
```

Ef 2e2 2e 2 2 d2 f 2 d2 e2a d2 b 2

```
// In list dùng Iterator
Iterator<String> it = listCars.iterator();
System.out.println("Danh sách xe: ");
while (it.hasNext()){
    System.out.println(it.next());
}
```

Qe 2 f2 2h 2e 2 2 d2 f 2eb 2 2 2 d2 f 2 e 2 2 2e 2

```
System.out.println("Nhập vào phần tử mới: ");
String temp = scanner.nextLine();
System.out.println("Nhập vào vị trí của phần tử cần thay đổi: ");
int index = scanner.nextInt();
listCars.set(index, temp);
```

```

    e    ? ? e    ? ?    d? f ? e    e? e ? ?
    ?
    ListIterator<String> ls = listCars.listIterator();
    while (ls.hasNext()){
        String str = ls.next();
        ls.set(str.toUpperCase());
    }

```

P ? ? f ? eb ? e ? ? d? ?

```

Collections.sort(listCars);

```

```

    ? ? e    ? ?    d? f ? ? ? ? ? ?    ? ?    ?Ke ? ?    ? e    ?
    ?
    System.out.print("Nhập vào vị trí đầu: ");
    int firstInd = scanner.nextInt();
    System.out.print("Nhập vào vị trí cuối: ");
    int lastInd= scanner.nextInt();
    // Sử dụng phương thức clear()
    listCars.subList(firstInd, lastInd).clear();

```

? f ? ? ? d? f ? b ?

```

ListIterator lsI = listCars.listIterator(listCars.size());
System.out.println("list sau khi đảo ngược: ");
while (lsI.hasPrevious()) {
    System.out.println(lsI.previous());
}

```

**Câu 5:** Sf ? e d? ? e? ? f? d? e ? ? e? e? ? ? ? ?e ? f ? ? e ?

df ? f? ? f ? ? ? d? ? ? ? ? ? ?

\_\_\_\_\_ A d?TreeSet ? ? ? f? d? ?

Q ? f? d?

```

TreeSet<String> treel = new TreeSet<>();
TreeSet<String> tree2 = new TreeSet<>();
//Tạo mảng 1
treel.add("Java");
treel.add("PHP");
treel.add("C#");
treel.add("Python");
treel.add("HTML");
treel.add("Spring");
//Tạo mảng 2
tree2.add("Java");
tree2.add("ASP.Net");
tree2.add("C#");
tree2.add("R");
tree2.add("CSS");
tree2.add("Lavarel");

```

Me 21f 2a d2 retainAll()

Me 2e f2a d2 addAll()

2

Me 2 2a d2 removeAll()

#### IV. Thực hành

**Câu 1:** Q 1 Sinhvien 2 2 2f e2f 2l 2 2e 2 e2 2hf 2ong 2b 2  
hf 2String 2 2afb 2 hf 2double) 2 d2 f2 2 e d2 e 2db b 2 b b 2  
2 P f d2Ke 2a 2f 2e 2 2f e2f 2 2 2e 2 2f 2e d2f 2f e2f 2  
2 e 2 2 2e e2V 2 2a d2 2 2 d f2 2 2 2a d2 2e 2hf 2 2 f 2  
e 2a 2f 2 2 2e 2 e2 2f e2f 2 2 f2 2ef 2e d2 2S 2a 2 e 2  
f 2

Q 2 2eb 2 2 f 2e f2eb 2 2e 2 dñ 2 2 2 2e 2 dñ 2 2e 2 2 ef 2 2 f 2e f2

**Câu 3:** P 2a d2 iib f 2e 2ef 2 2 2 2 2 2 e2 d2 iib f 2 2e 2 ef 2

d2 2 2 2 2 2e 2 f 2 2 e2 2 f2e d2 2e 2 f 2 2 2 e 2 d 2 ef 2 2e 2 2e 2 2 d2 f 2 2 2 2e 2 f 2 2e 2

d2 2a 2 e2 2 2 2 2e 2 f2 2 e2 d2 2e 2 f 2 2 2 e 2he d2 2 d2e 2 f 2 2 d2 2a e2 e2 2 2e 2e 2 d2 2

d2 2 2a d2 2e 2 f 2e 2 f 2e 2 2 2 2e 2 2e 2 f2 2 e2 2e 2 f 2 2 d2 d2e 2ef 2e 2 f 2 2 d2 2 2 2e 2 2

Q d2 2 de 2 2 d2 2 2 2e 2e 2e 2 f 2 f2a 2 e2 2 2 2 2 2 2f 2e 2 2 d f2 dñ 2 2 f 2 2 dñ 2a 2 e2 2

J 2he e2 d2 2 d2 2 2 2a e2 2 2 d2 d2 2 2 2 i 2a e2 e2 2eb 2e 2 2f 2a 2a e2 2 f 2 2 2a e2 e2 2

**Câu 4:** J 2 2 f2 2 2 f2 2 2 f2 2 2 2 f2 2 2 2 2 f2e 2CNH 2 2

f2 2 f2 2 2e 2 e2 2 2e 2 e 2f 2e d2 e2 2ef 2 f2e e2 2e 2he 2e 2 f2e 2 2 2

\_\_\_\_\_ f 2 2 2 2a d2 2shuffle 2 2 iib f 2 2ef 2 f2

I 2 2 2 2 f2

I 2 2 2 f2 2 2 2he d2 d2e 2 2e d2e 2 2 f2

**Câu 5:** Sf 2e d2 e2 2 2 f 2f d2 e 2P 2a d2 iib f 2Q bbj 2 2 2 2 2 2a d2 2 2 2

**Lớp Từ** 2 2 2 2 2f d2 e2 f2 2e 2 e2 2f d2 e2 de 2 f2 2 2 e 2ef2e 2 2 2e d2e 2 2 2e d2f 2 2 2 2

**Lớp Từ điển** d 2 2e d2e 2Qe 2 2 2 2a e2 e2 2Q 2 2