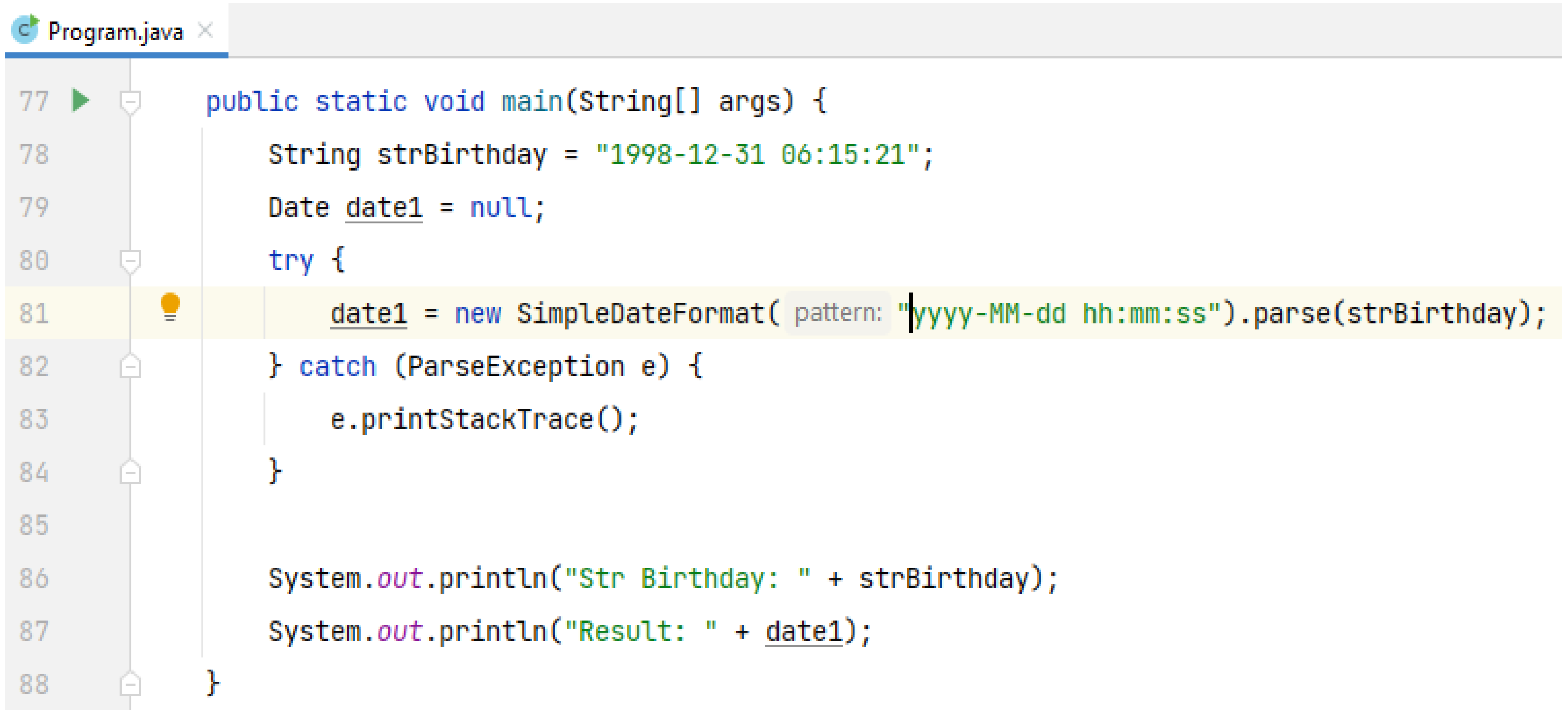
Nội Dung

Author: Nguyễn Văn Tuyên
Email: nguyenvantuyen6789@gmail.com
Phone: 0888.335.333
Fb: facebook.com/nguyenvantuyen6789

1. Create Date

Create Date



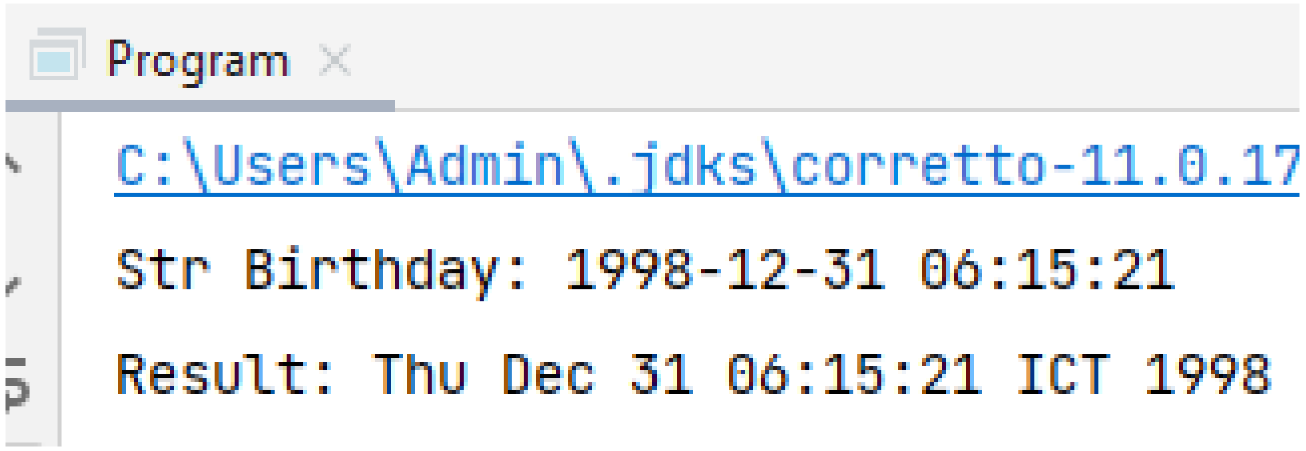
Program.java

```java
public static void main(String[] args) {
    String strBirthday = "1998-12-31 06:15:21";
    Date date1 = null;
    try {
        date1 = new SimpleDateFormat( pattern: "yyyy-MM-dd hh:mm:ss").parse(strBirthday);
    } catch (ParseException e) {
        e.printStackTrace();
    }

    System.out.println("Str Birthday: " + strBirthday);
    System.out.println("Result: " + date1);
}
```

1. Create Date

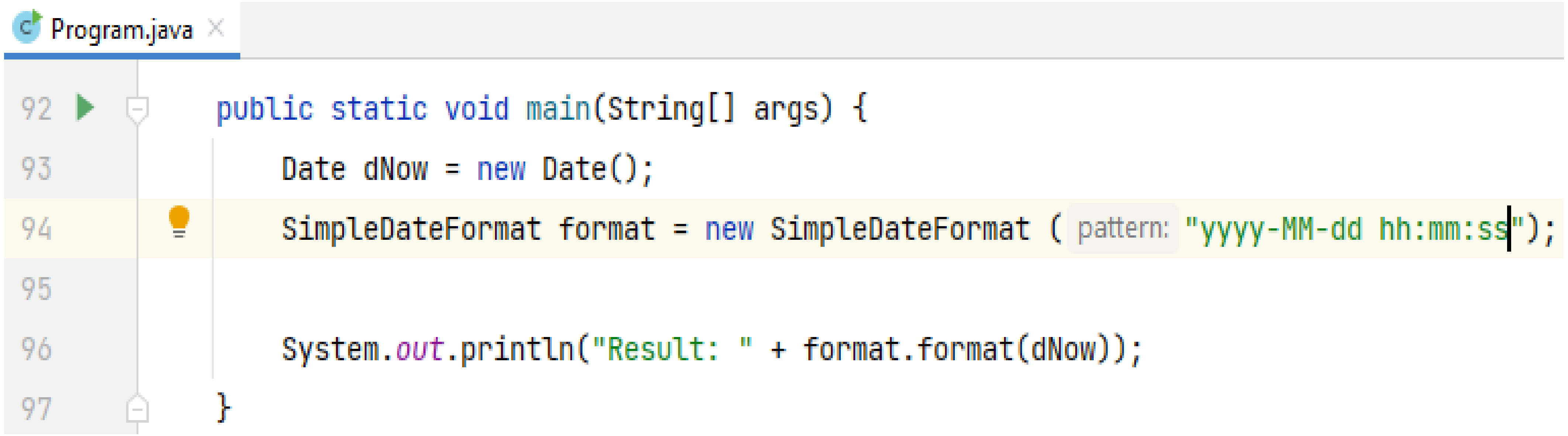
Kết Quả



```
C:\Users\Admin\.jdks\corretto-11.0.17
Str Birthday: 1998-12-31 06:15:21
Result: Thu Dec 31 06:15:21 ICT 1998
```

1. Create Date

Create Current Date



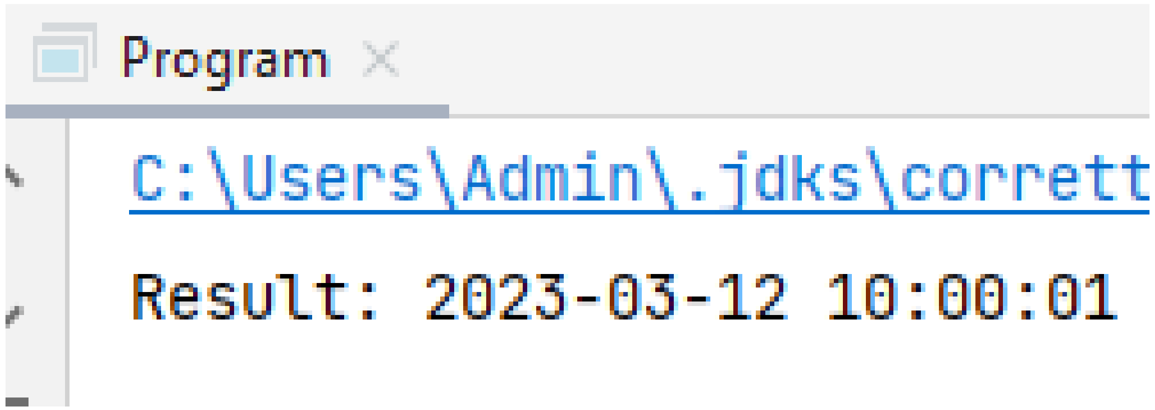
Program.java

```java
public static void main(String[] args) {
    Date dNow = new Date();
    SimpleDateFormat format = new SimpleDateFormat ( pattern: "yyyy-MM-dd hh:mm:ss");

    System.out.println("Result: " + format.format(dNow));
}
```

1. Create Date

Result

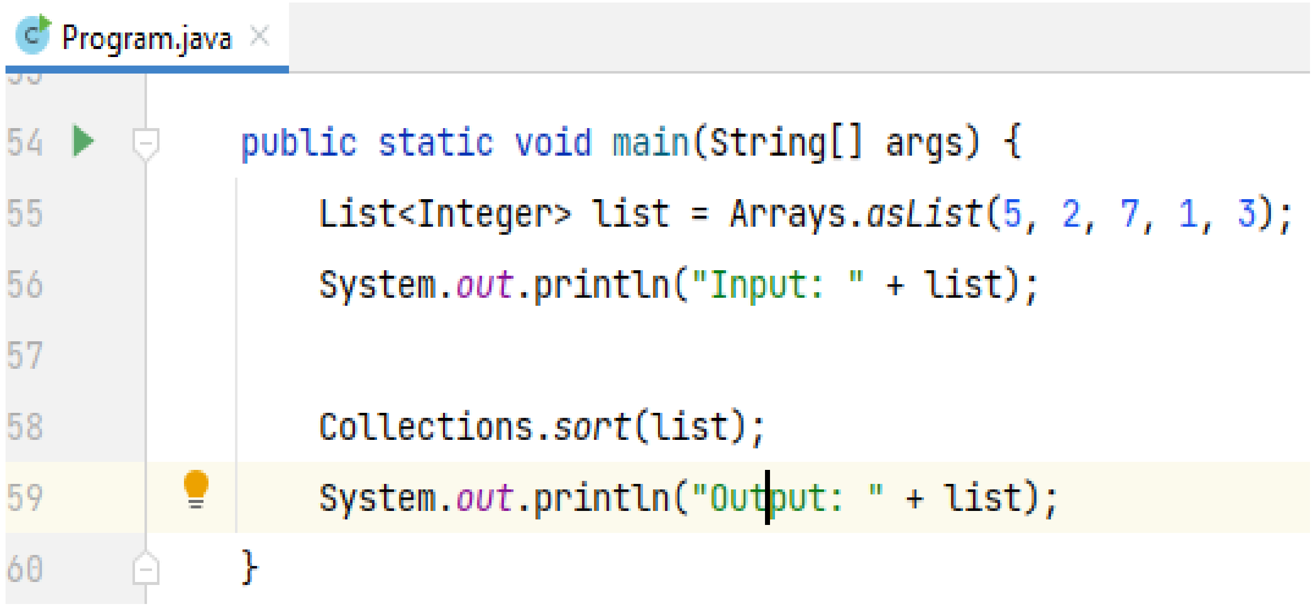


```
C:\Users\Admin\.jdks\corrett
Result: 2023-03-12 10:00:01
```

2. Sort Integer, String, Date

Sort: Integer, String, Date

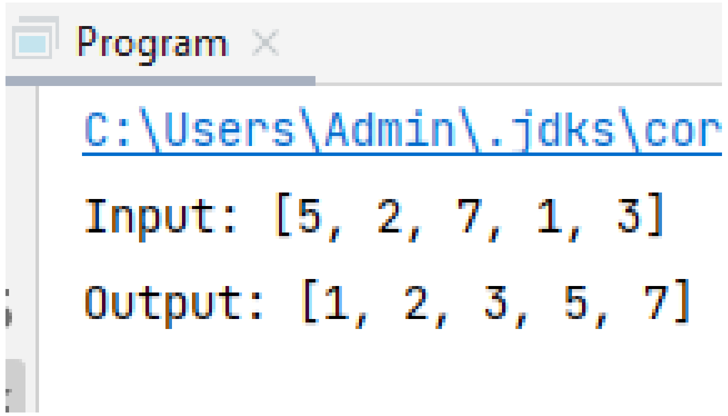
Integer



Program.java

```java
    public static void main(String[] args) {
        List<Integer> list = Arrays.asList(5, 2, 7, 1, 3);
        System.out.println("Input: " + list);

        Collections.sort(list);
        System.out.println("Output: " + list);
    }
```



Program

```
C:\Users\Admin\.jdks\cor
Input: [5, 2, 7, 1, 3]
Output: [1, 2, 3, 5, 7]
```

2. Sort Integer, String, Date

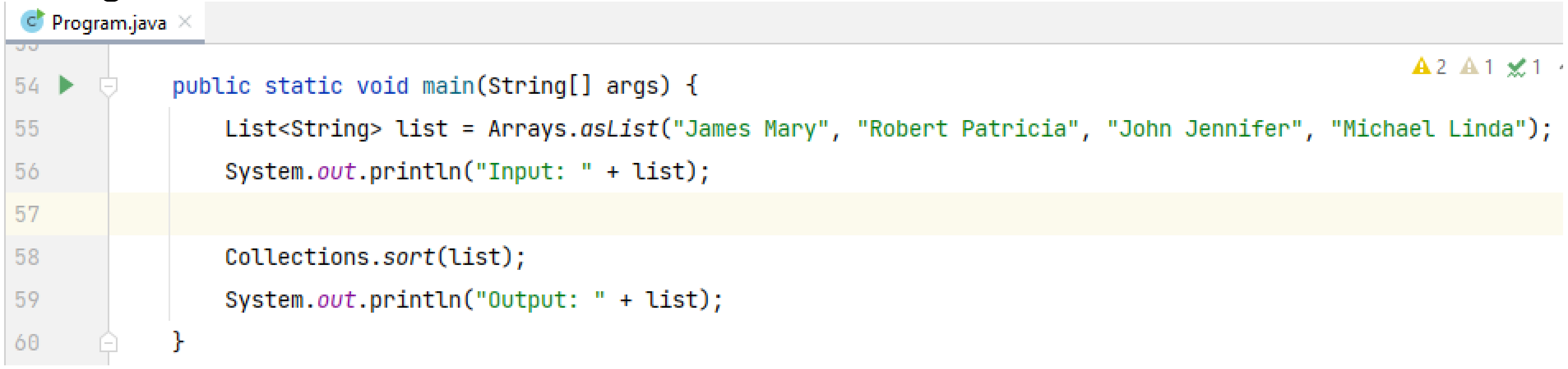
Sort Decending

```
Comparator<Integer> reverse = Collections.reverseOrder();
Collections.sort(list, reverse);
```

2. Sort Integer, String, Date

Sort: Integer, String, Date

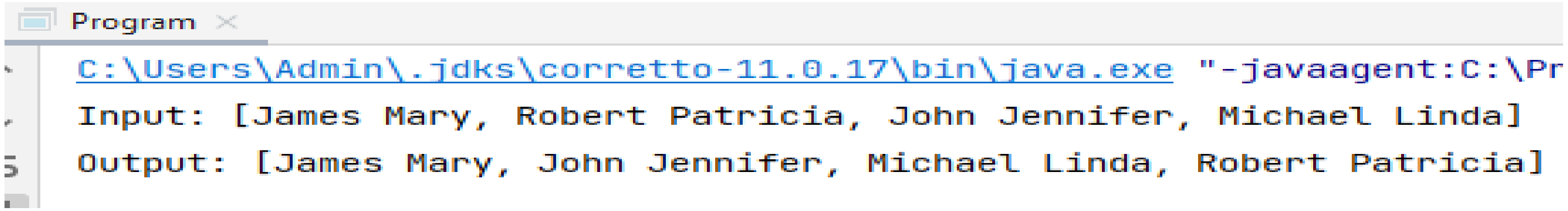
String



Program.java

```
    public static void main(String[] args) {
        List<String> list = Arrays.asList("James Mary", "Robert Patricia", "John Jennifer", "Michael Linda");
        System.out.println("Input: " + list);

        Collections.sort(list);
        System.out.println("Output: " + list);
    }
```



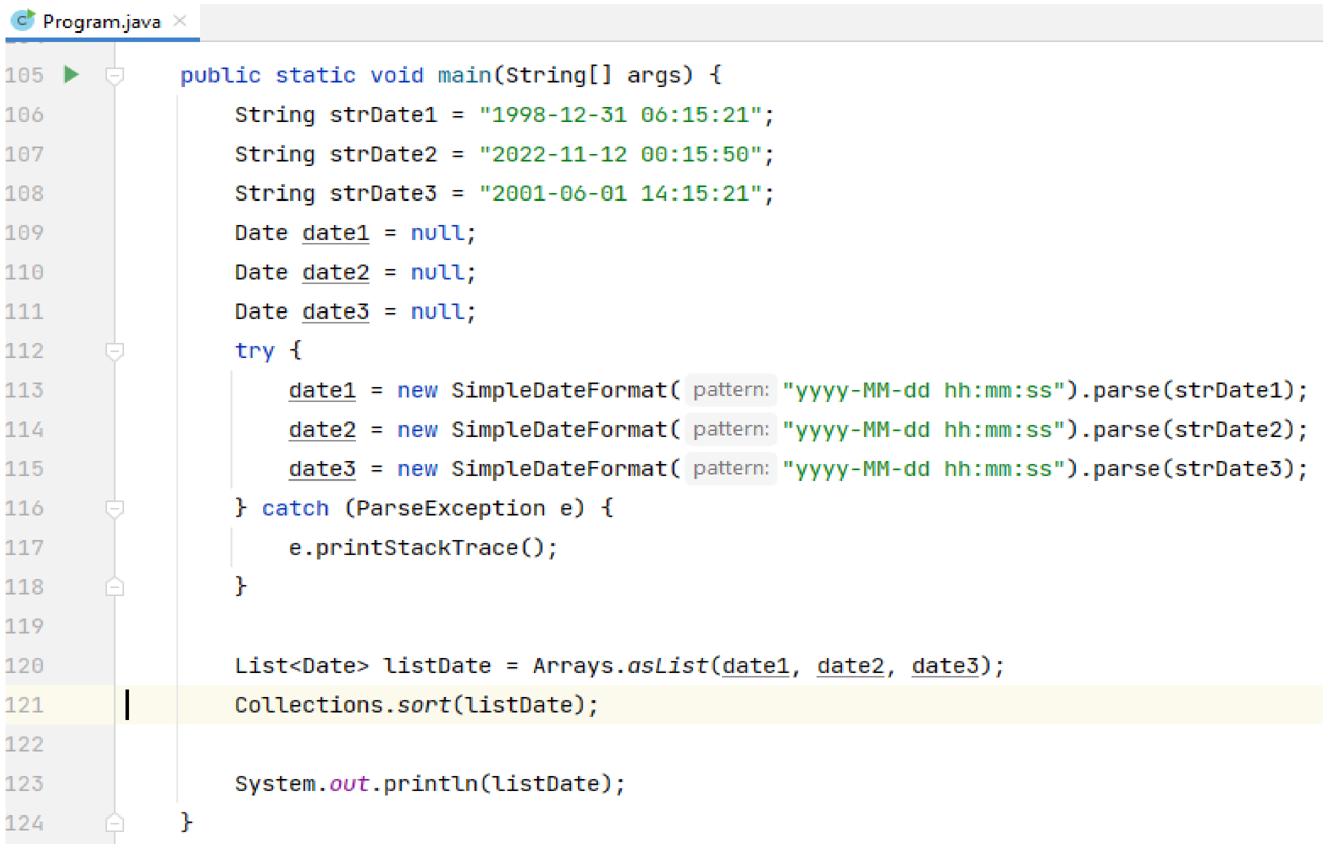
Program

```
C:\Users\Admin\.jdks\corretto-11.0.17\bin\java.exe "-javaagent:C:\Pr
Input: [James Mary, Robert Patricia, John Jennifer, Michael Linda]
Output: [James Mary, John Jennifer, Michael Linda, Robert Patricia]
```

2. Sort Integer, String, Date

Sort: Integer, String, Date

Date



Program.java

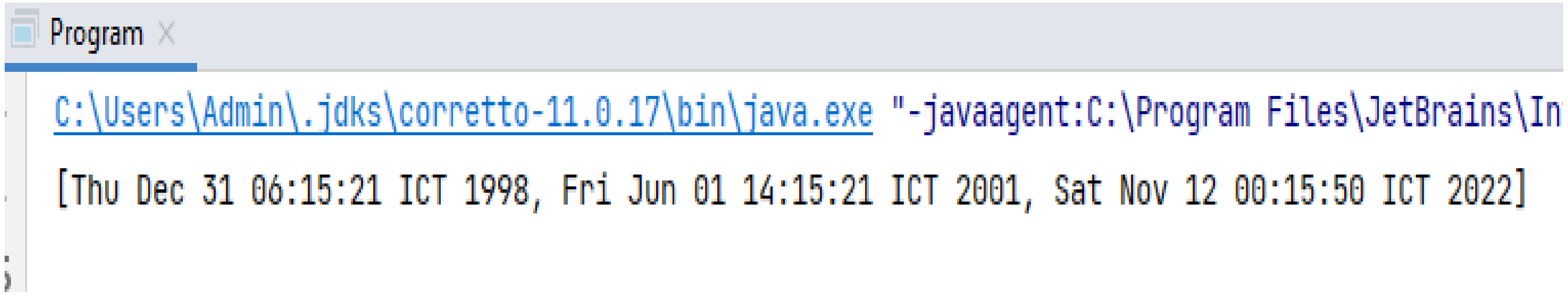
```java
    public static void main(String[] args) {
        String strDate1 = "1998-12-31 06:15:21";
        String strDate2 = "2022-11-12 00:15:50";
        String strDate3 = "2001-06-01 14:15:21";
        Date date1 = null;
        Date date2 = null;
        Date date3 = null;
        try {
            date1 = new SimpleDateFormat( pattern: "yyyy-MM-dd hh:mm:ss").parse(strDate1);
            date2 = new SimpleDateFormat( pattern: "yyyy-MM-dd hh:mm:ss").parse(strDate2);
            date3 = new SimpleDateFormat( pattern: "yyyy-MM-dd hh:mm:ss").parse(strDate3);
        } catch (ParseException e) {
            e.printStackTrace();
        }

        List<Date> listDate = Arrays.asList(date1, date2, date3);
        Collections.sort(listDate);

        System.out.println(listDate);
    }
```

2. Sort Integer, String, Date

Kết Quả

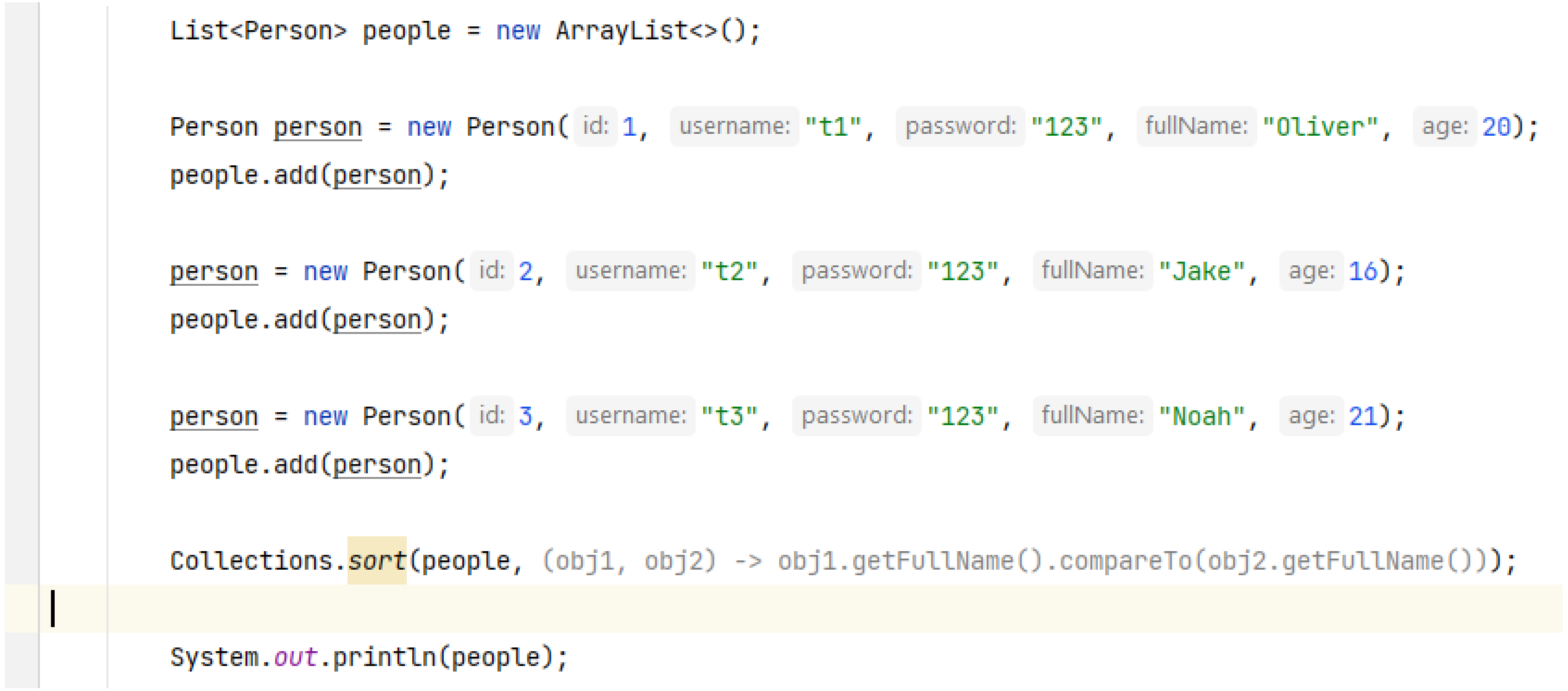


Program

```
C:\Users\Admin\.jdks\corretto-11.0.17\bin\java.exe "-javaagent:C:\Program Files\JetBrains\In
[Thu Dec 31 06:15:21 ICT 1998, Fri Jun 01 14:15:21 ICT 2001, Sat Nov 12 00:15:50 ICT 2022]
```

3. Sort Object By Property

Sort By fullName



```
List<Person> people = new ArrayList<>();

Person person = new Person( id: 1, username: "t1", password: "123", fullName: "Oliver", age: 20);
people.add(person);

person = new Person( id: 2, username: "t2", password: "123", fullName: "Jake", age: 16);
people.add(person);

person = new Person( id: 3, username: "t3", password: "123", fullName: "Noah", age: 21);
people.add(person);

Collections.sort(people, (obj1, obj2) -> obj1.getFullName().compareTo(obj2.getFullName()));

System.out.println(people);
```

3. Sort Object By Property

Kết Quả

```
[
Person{id=2, username='t2', password='123', fullName='Jake', age=16},
Person{id=3, username='t3', password='123', fullName='Noah', age=21},
Person{id=1, username='t1', password='123', fullName='Oliver', age=20}
]
```

3. Sort Object By Property

Decending

obj1.getFullName().compareTo(obj2.getFullName()) ->

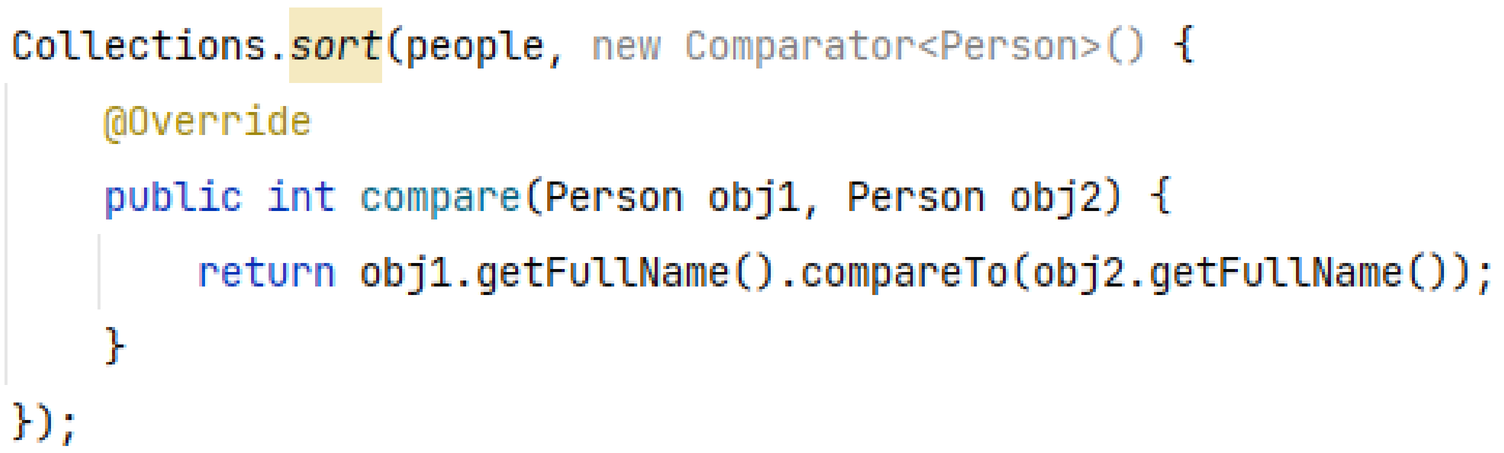
obj2.getFullName().compareTo(obj1.getFullName())

3. Sort Object By Property

Vd: Hãy sắp xếp theo age tang, giảm

3. Sort Object By Property

Có Thể Dùng Cách 2, Decending đổi obj1 với obj2



```java
Collections.sort(people, new Comparator<Person>() {
    @Override
    public int compare(Person obj1, Person obj2) {
        return obj1.getFullName().compareTo(obj2.getFullName());
    }
});
```