Java 8

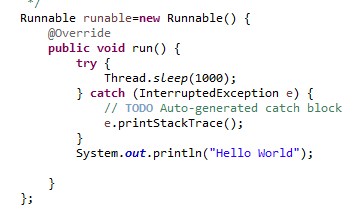
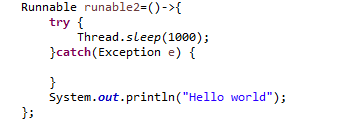
Lymda Expresstion :

Example runnable implemention :

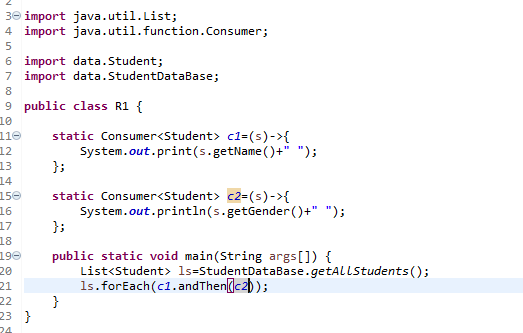
**new** Thread(runable).start();

Here we need to pass object of that class which implement of runnable interface

without lymda : with lymda:

**Working with consumer / forEach / chain**

****

**Output:**

Adam male

Jenny female

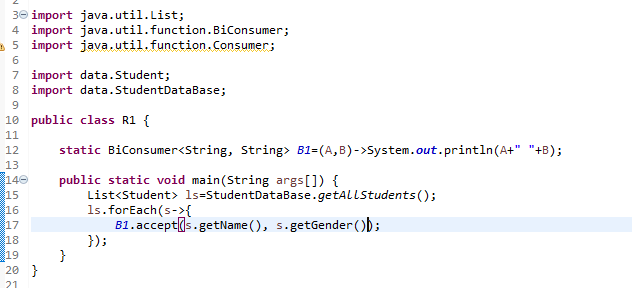
Emily female

Dave male

Sophia female

James male

**Working with Biconsumer / forEach / chain**

****

**Output:**

Adam male

Jenny female

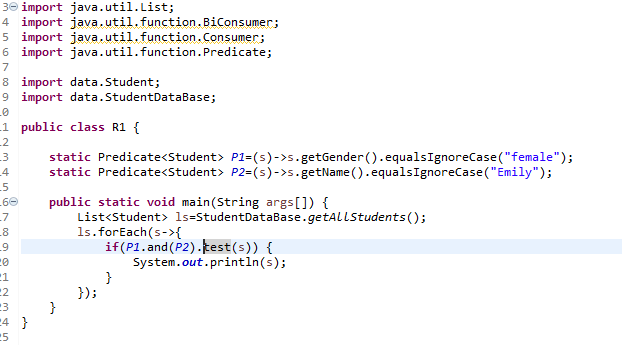
Emily female

Dave male

Sophia female

James male

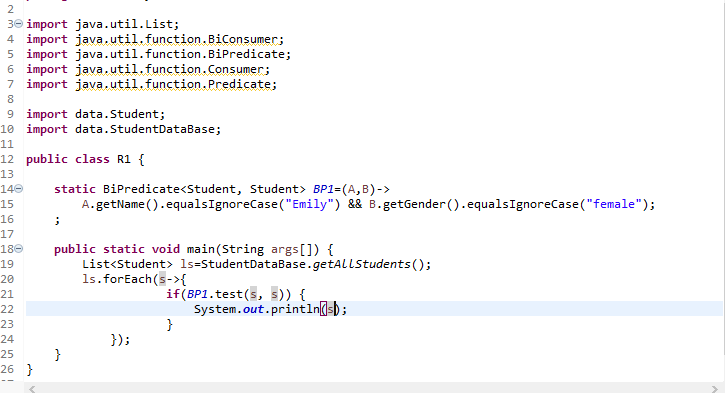
**Working with** Predicate and



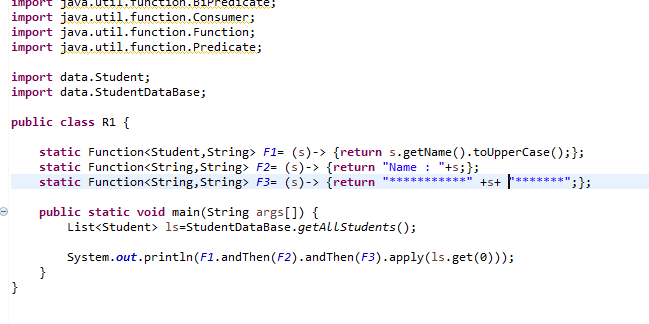
Output

Student{name='Emily', gradeLevel=3, gpa=4.0, gender='female', activities=[swimming, gymnastics, aerobics]}

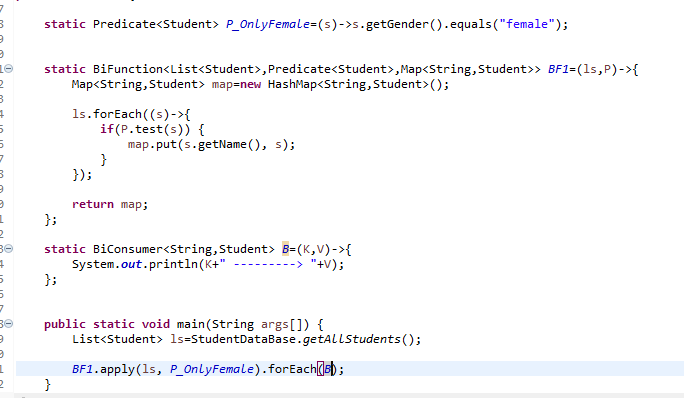
**Working with Bi**Predicate



Function

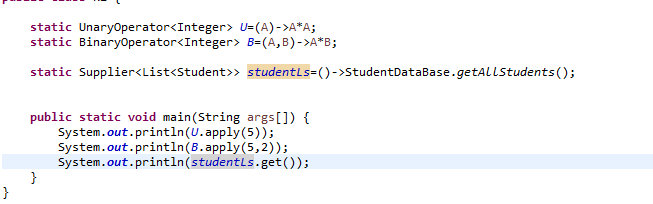


BiFunctional interface….



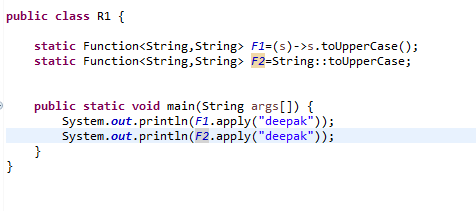


Uinary,Binary,supplier



Method reference ::

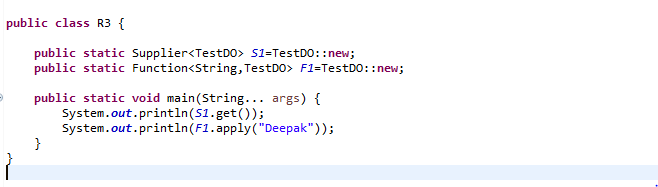
Class::method



Functional reference with consumer and unaryOperator

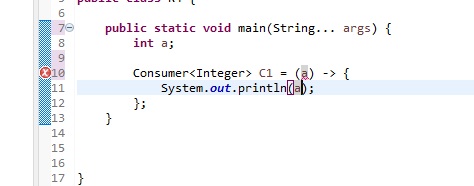


**Constructor Reference :**

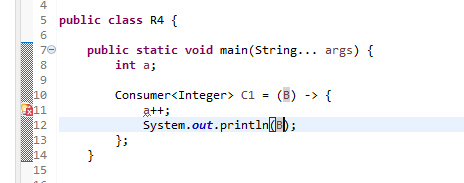


**Variable restriction**

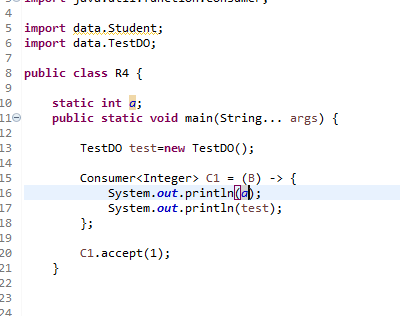
**Point 1. You can not give same name to local variable and parameter of lymda interface**

****

**Point 2 . can not change value inside lymda interface**

****

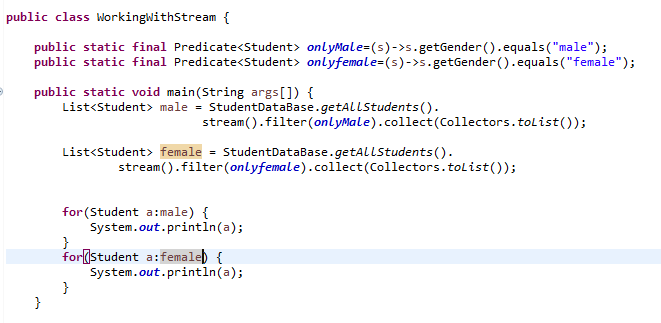
**Point 3 . You can use instance variable and class barible inside the lymda method**

****

**Java 8 stream API**

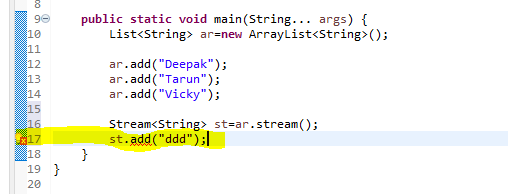
**1 filter(Predicate<T>) and collect (**

****

****

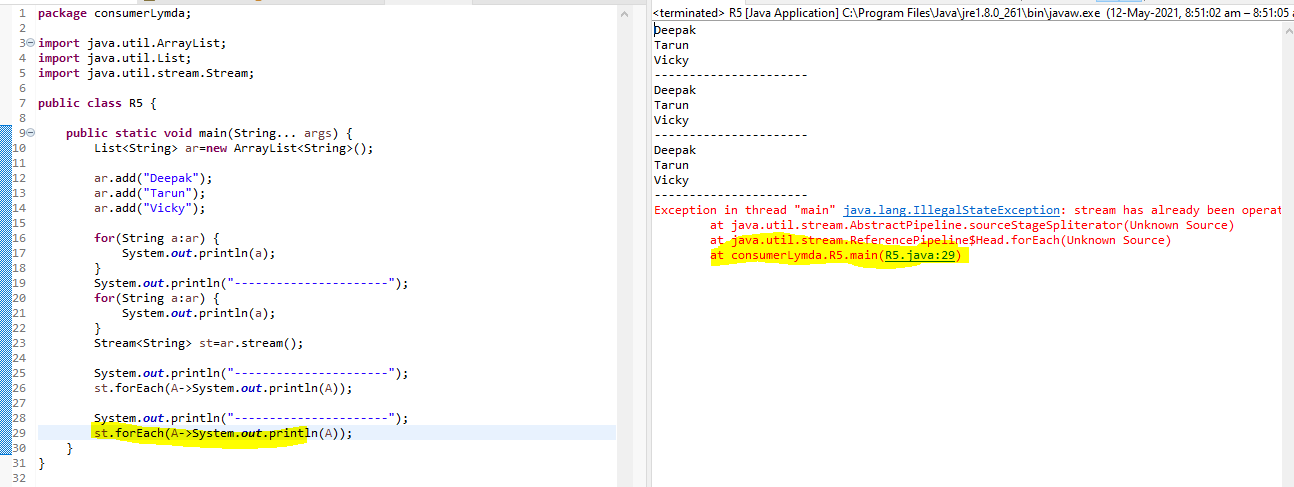
**Q difference between collection and stream.**

|  |  |
| --- | --- |
| **Collection** | **Stream** |
| 1. **Collection add and modify element** | **You cant modify elements** |

****

**Line no 17 showing that you can not enter new element to stream.**

|  |  |
| --- | --- |
| **Collection** | **Stream** |
| **Treverse multiple time** | **Only once** |

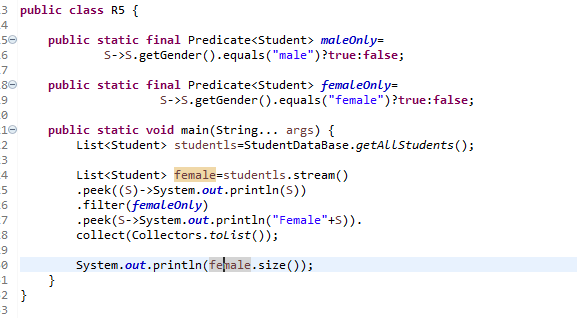
****

**You can see in code when I try to retervierse stream it show ERROR**

**Q. How can you debug stream API and what do mean by Peek method in stream API ?**

**Ans . Peek method can use as debug method of stream operation it take consumer as an argument .**

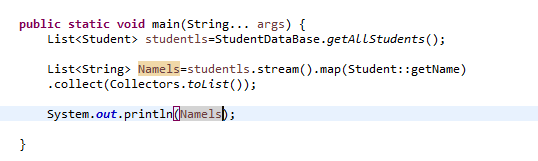
**Example.**

****

**Q . How can you convert one stream class to another.(What do you mean By “map” method is stream api?**

Ans:

Map method is use for converting one stream type to another stream type.

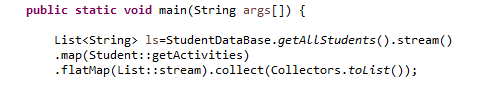


In this example we convert **student** stream to **string** stream.

**Q. Why we use Flat map;?**

Ans :

* when we have stream of Stream<List<String>> .

* Like in Student class we have method to get data Student::getName() but not List class.
* 

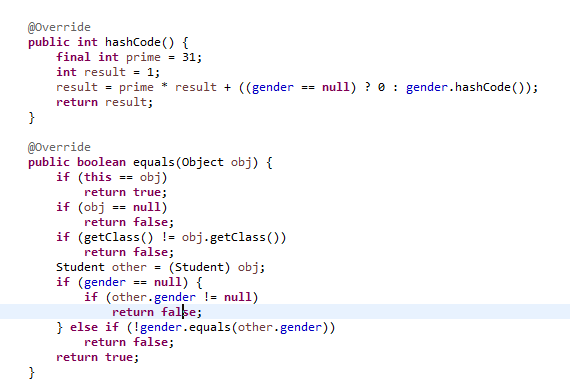
**Q. what do you understand by distinct , count , sort of Stream API ?**

Ans these are method in stream api. Perpouse of these method is clear by

**Q. Suppose you have to student object with same id ? how will you use distinct method of stream api**

Ans by overriding hashcode and equals method with id;

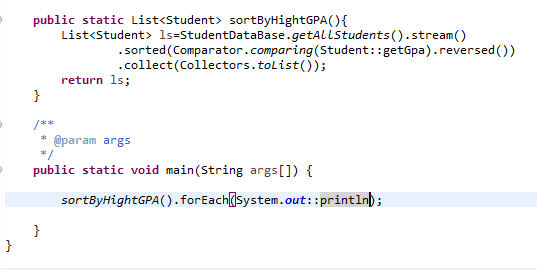
Eample (Note: In this example same gender treat as a same record);



Q.How to implement custom sort ? with revese order ?

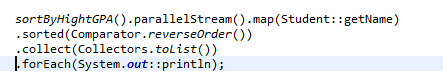
Ans .

**import** java.util.Comparator;



Q.How to you reverse List<String> using stream api ?

Ans .



Q. what does you mean by reduce function ?

Ans . It will reduce stream to a single value .

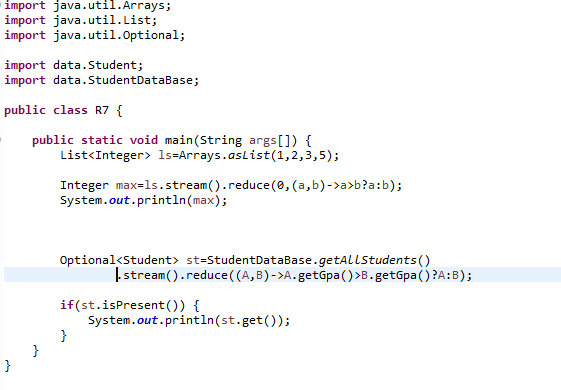
For example finding hieest number in list

There are 2 type of reduce method

One with default value and another is without default value.

Without default value return optional .





Find min and max Example?



**Q.What do understand by limit and skip method in stream ?**

**Ans .**

**Limit : For example you have list ->[1,2,3,4,5,6,7] and you use limit(3)**

Then this will create a sub stream of [1,2,3]

**Skip** : For same example above if you use skip(3) will create a list of

**[4,5,6,7]**

**Code example :**

****

**Q. difference between anyMatch,allMatch,noneMatch**

**Ans :**

**anyMatch : This method will return true if any element in stream match with Condition present in anyMatch(contion) method.**

**AllMatch :**

**This will return true if all element in stream match the condition**

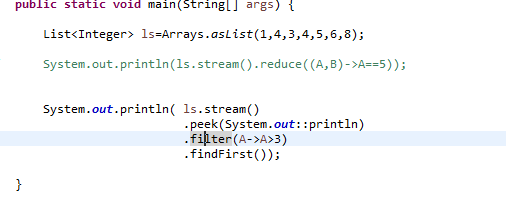
**NoneMatch : this will return true only if all element in stream does not match**

**The contion.**

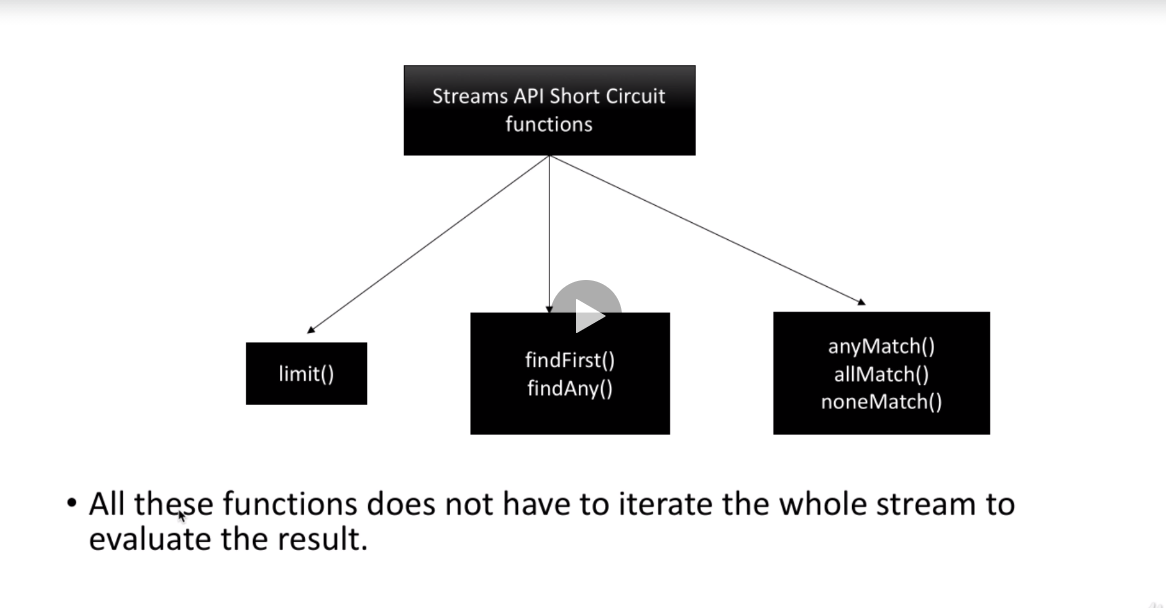


**Q.Find First and Find Any ?**

**Ans. Not complete answere until I reach pareler stream**

****

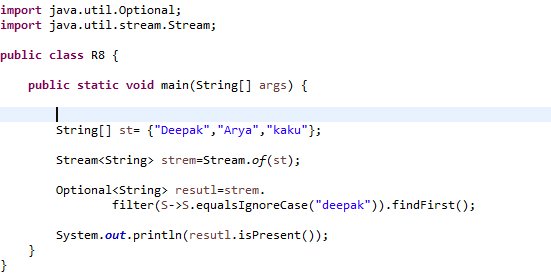
**As soon as filter pass any element to findFirst it will return the result won’t iterate all**

****

Q. What does you mean by “**of**” static method stream ?

Ans : You can use this to convert array to stream.

Example .



Q.Iterator vs for loop

Ans . for this time I did not found any difference.



Q . IntStream ,LongStream

Ans . range(1,5) - 1,2,3,4

RangeClosed(1,5) -> 1,2,3,4,5

Min,max,average

Example :



Q. IntSteam Method

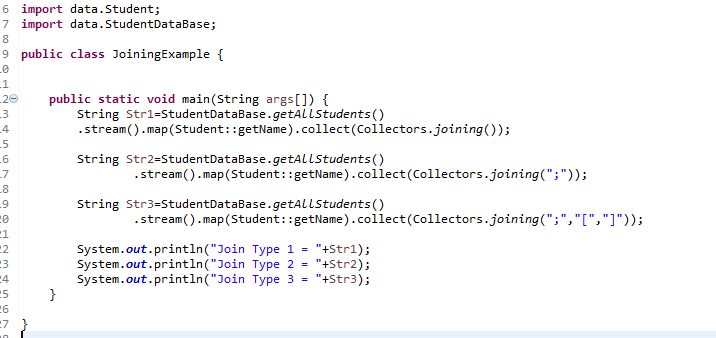
Boxing,unboxing,mapToObject,mapToLong,mapToDouble



Terminator Operators

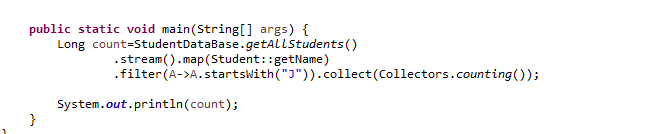
Q. Joining all type ?

Ans :



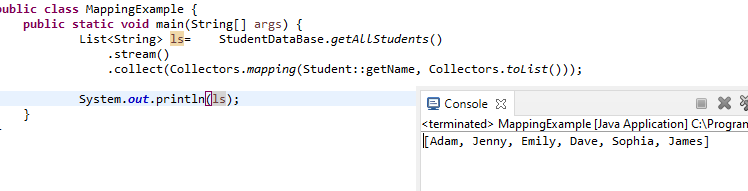


Q. Counting (collectors.counting)?

Ans . 



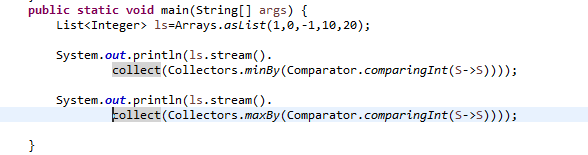
Q. Mapping





Q. MinBy and MaxBy

Ans . take comprator and return optional



Q. Summing and average terminal method



Q . Grouping example really \*



Q. have you every try with partition by? ( how to create 2 list one for odd number one for even number).

**public** **static** **void** main(String[] args) {

List<Integer> ls =IntStream.*range*(0, 10).boxed().toList();

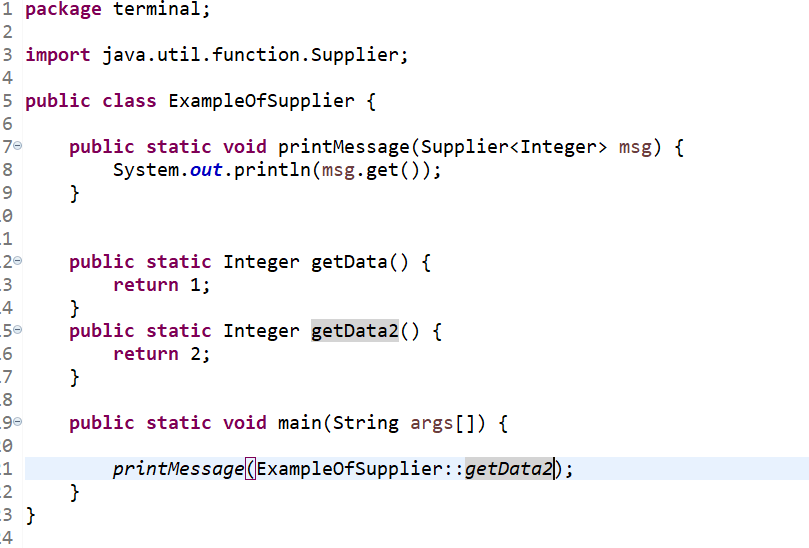
Map<Boolean,List<Integer>> mp=ls.stream().collect(Collectors.*partitioningBy*(S->S%2==0));

System.***out***.println("Even "+mp.get(**true**));

System.***out***.println("Odd "+mp.get(**false**));

}

Q working example of supplier



Q where should not use Pareller stream ?

Ans . while working with mutable objects.



Q.working with optional.

Ans . 

Q. difference between optional “of” and “ofnullable” ?

Ans  of -> always required valid value in case of null throw

Exception

Ofnullable -> can return value or empty both

Q. orElse vs orElseGet vs orElseThrow



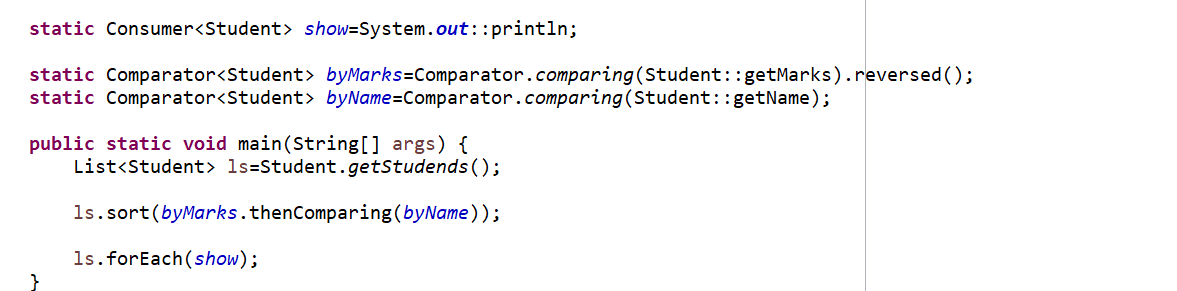
Q Difference between isPresent vs ifPresent in optional?

Ans isPreset -> return Boolean

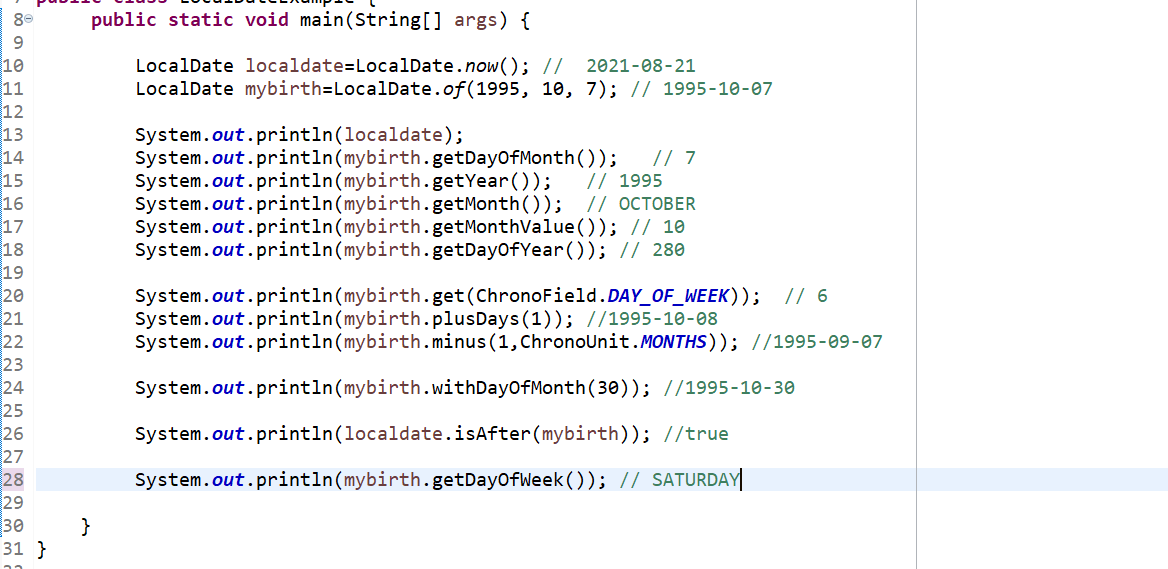
ifPresent -> take consumer like ifPresent(S->sysout(S);

Q. How to sort with multiple comparator.

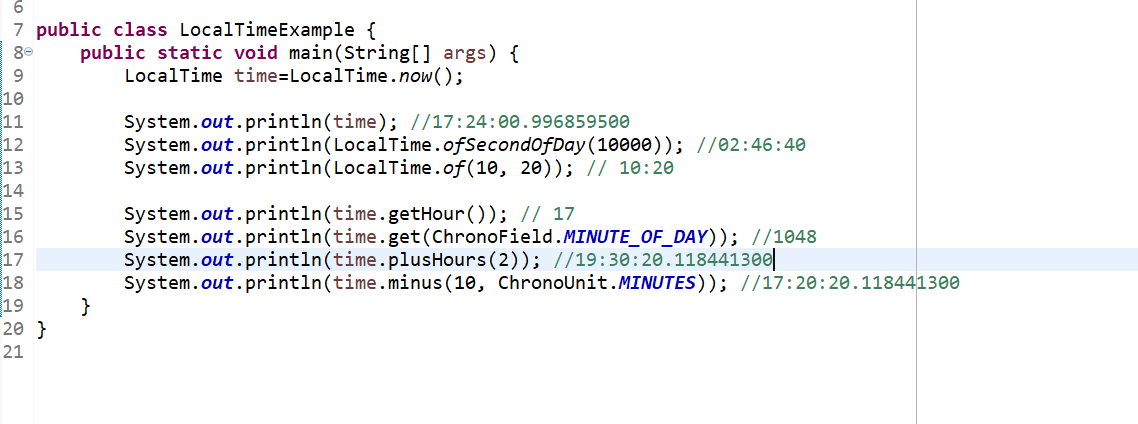




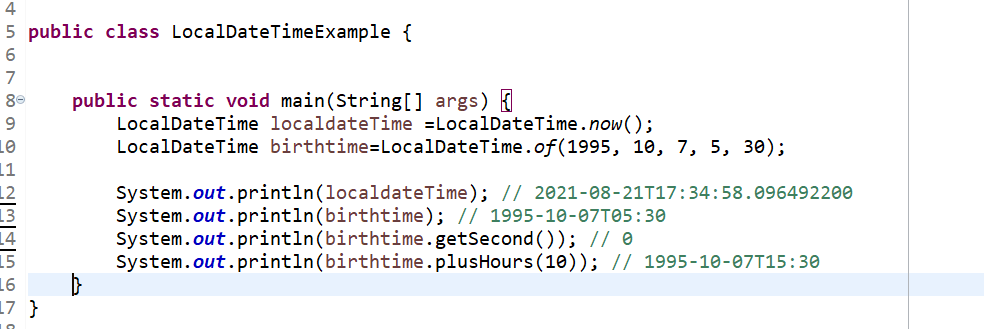
**LocalDate methods**



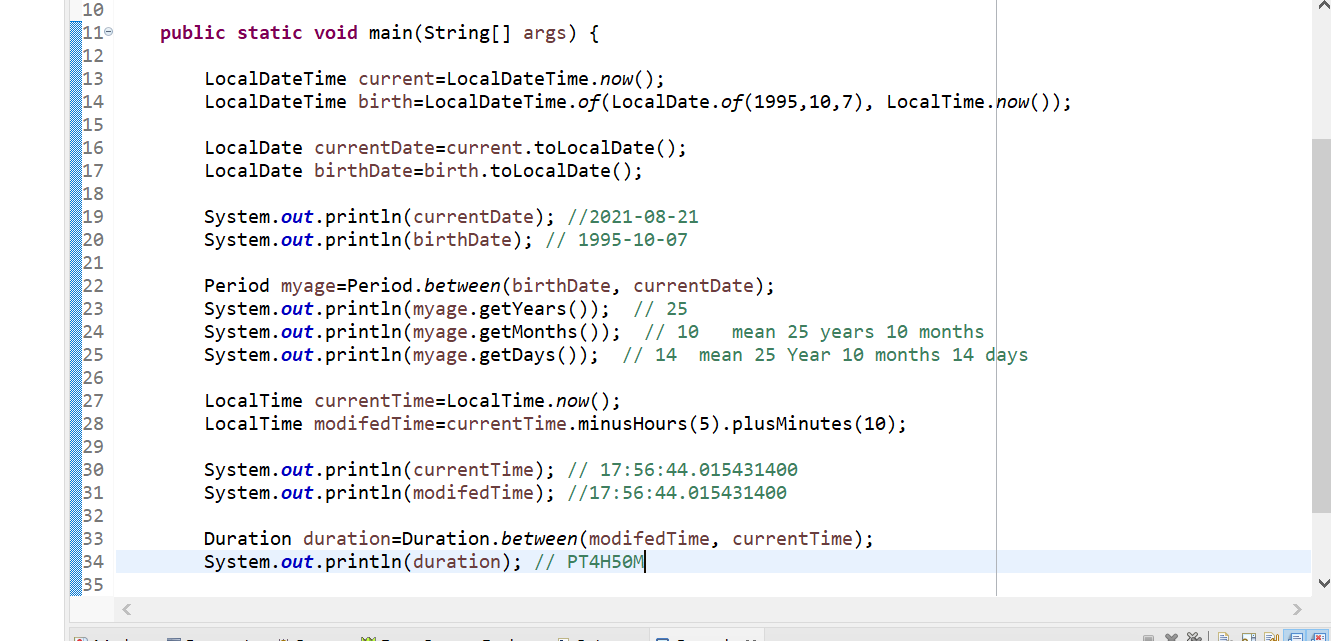
**LocalTime .**



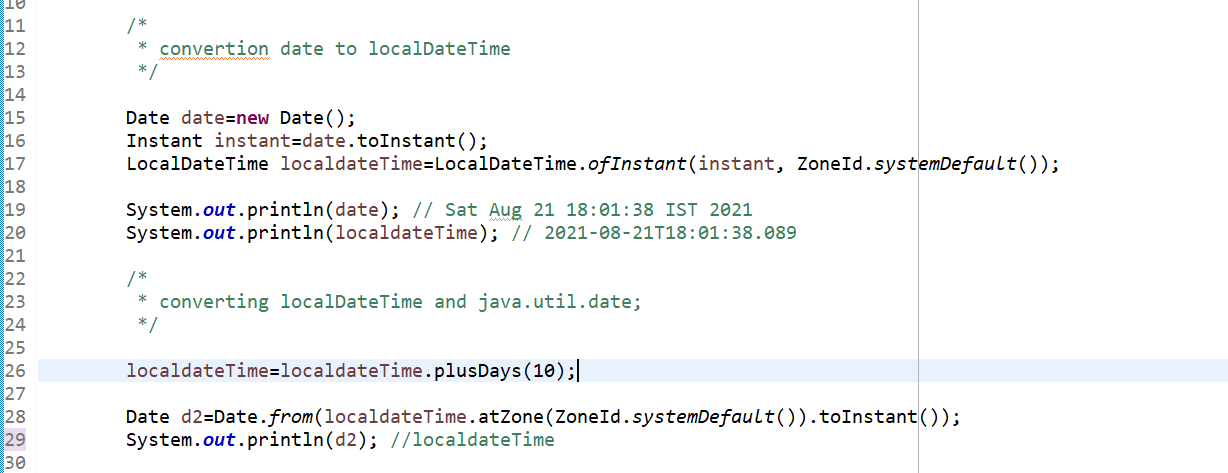
LocalDateandTime



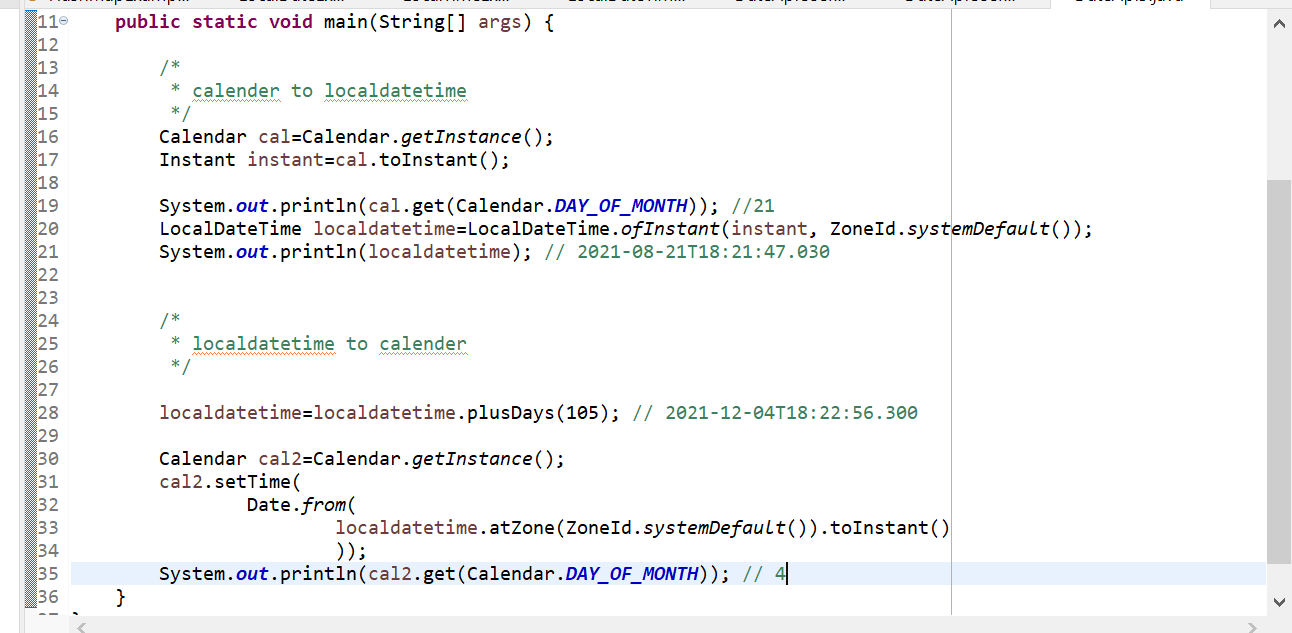
Some more date api



Converting java.util.date to LocalDateTime to java.util.date



**Converting calender to localDateTime ververse.**



Formating



