Create a Netbeans project and add a Main class to it called “StreamEX1”.

Within this class, initialise a List with the following 10 Integer (objects) 23, 5, 67, 4, 3, 31, 90, 12, 45, 89

Then using streams carry out the following:

1. Print the numbers to the console.
2. Sort the numbers and again print them (in ascending order) to the screen.
3. Determine and print the largest number in the list.
4. Determine and print the smallest number in the list.
5. Determine and print the average of the numbers in the list.
6. Determine and print the sum of all the numbers in the list.
7. Print the numbers in the list that are less than 20.
8. Print the numbers in the list that are between 20 and 50 (inclusive).
9. Print the even numbers in the list.

Add another Main class to your project called “StreamEX2”.

Within this class, initialise a List with the months of the year (as Strings).

Then using streams carry out the following:

10. Print a count of the number of elements in this list (use a stream for this and not simply printing the size of the list).
11. Print all those month names which start with the letter “J”.
12. Print all those month names which have 5 letters or less.
13. Print all those month names which end with “ber” and contain 8 characters.
14. Using a stream, produce a filtered list which contains each of the twelve month names. However, in the filtered list, each occurrence of the letter ‘a’ should be replaced with ‘@’ in each of the month names. Print the contents of the filtered list to the console. For example:

```
J@nu@ry
Febru@ry
M@rch
April
M@y
June
July
August
September
October
November
December
BUILD SUCCES
```

You should end up with two lists here – the original and the filtered list (where every occurrence of the letter ‘a’ is replaced with ‘@’).

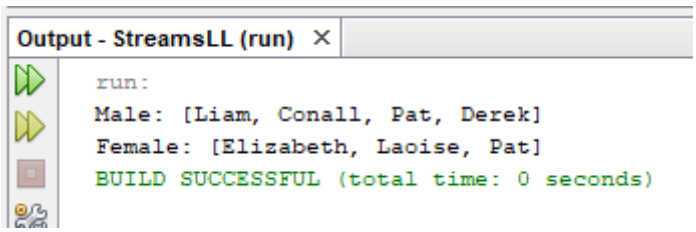
Add a class called 'Person' to your project. Ensure it matches the structure of the Person class from the lecture notes (slide 15).

Add another Main class to your project called "StreamEX3".

Create a list to store Person objects. Add a number of Person objects (at least 7) to this list (consider using the test data from slide 16 of the lecture notes).

Then using streams carry out the following:

15. Print to the console the details of all females who are aged between 40 and 50.
16. Determine and print to the console (as a double) the average age of all males in the list.
17. Using a stream, filter the male names into a separate list and then display them (you will end up with two lists here – the original and the new list containing just the names of all the males).
18. Display and group the names of the people in the list by gender – you will need to use the groupingBy method of the Collectors class to complete this exercise. Your output should look something like the following:



Consult the following code and examine the output it produces. Use the code as a basis for your solution to #18.

```
76          Map<Person.Gender, List<Person>> byGender
77                  = persons
78                          .stream()
79                          .collect(
80                                  Collectors.groupingBy(Person::getGender));
81
            for (Map.Entry<Person.Gender, List<Person>> entry : byGender.entrySet()) {
83              System.out.println(entry.getKey() + ": " + entry.getValue());
84          }
```