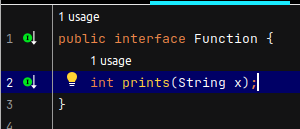
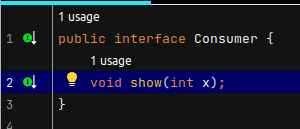
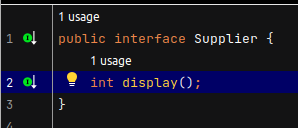
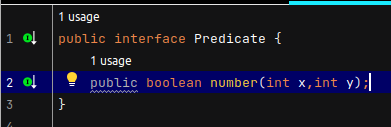
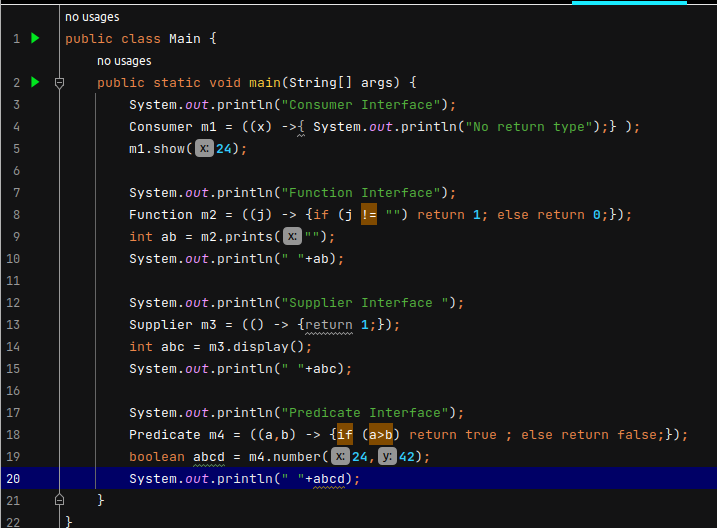
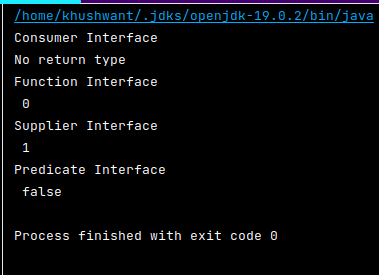
Q1. Implement following functional interfaces from java.util.function using lambdas:

* Consumer
* Supplier
* Predicate
* Function









Q2. Create and access default and static method of an interface.

public interface MyInterface {

default void myDefaultMethod() {

System.*out*.println("This is a default method.");

}

static void myStaticMethod() {

System.*out*.println("This is a static method.");

}

}

public class MyClass implements MyInterface {

public static void main(String[] args) {

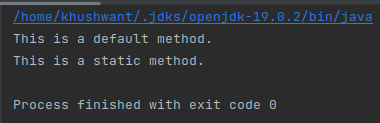
MyClass myClass = new MyClass();

myClass.myDefaultMethod();

MyInterface.*myStaticMethod*();

}

}



Q3. Sum all the numbers greater than 5 in the integer list using streams

import java.util.Arrays;

import java.util.List;

public class Example {

public static void main(String[] args) {

List<Integer> numbers = Arrays.*asList*(2, 7, 5, 9, 3, 8, 6);

int sum = numbers.stream()

.filter(n -> n > 5)

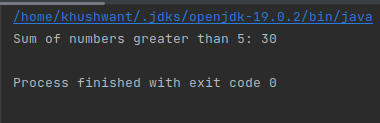
.mapToInt(Integer::intValue)

.sum();

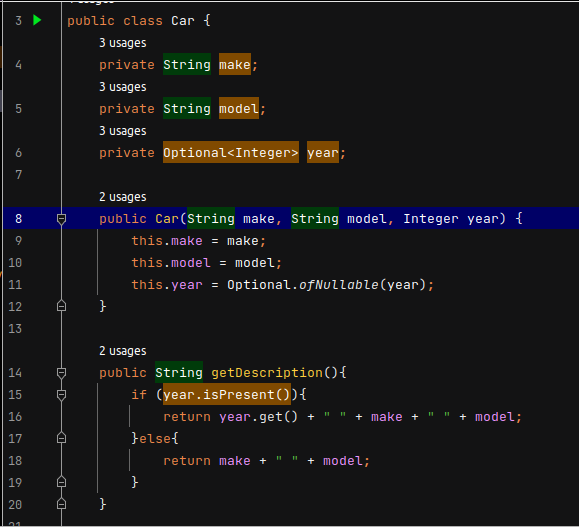
System.*out*.println("Sum of numbers greater than 5: " + sum);

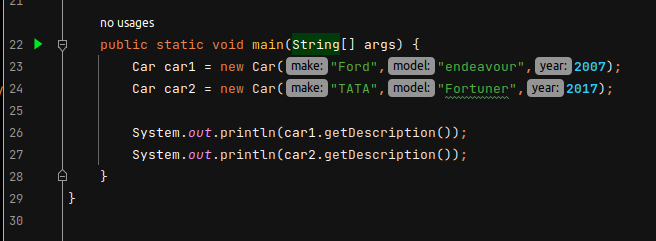
}

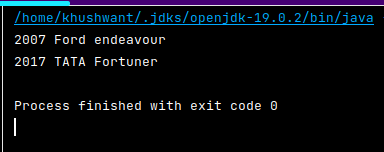
}



Q4. Write a program to showcase the use of optional class







Q5. Given a list of objects of following class:

class Employee{

String fullName;

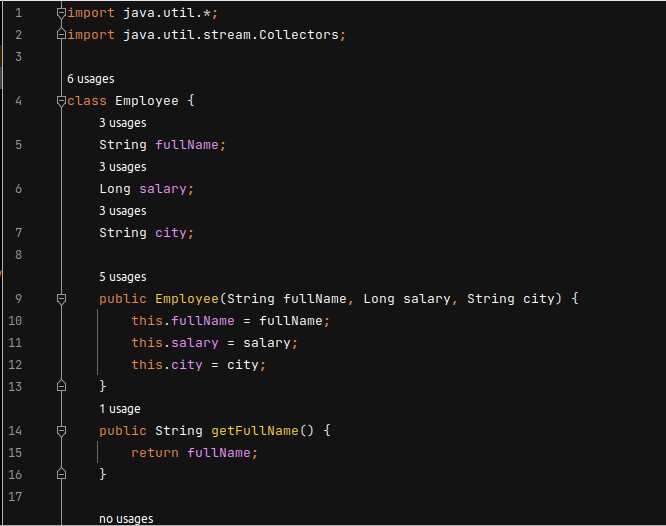
Long salary;

String city;

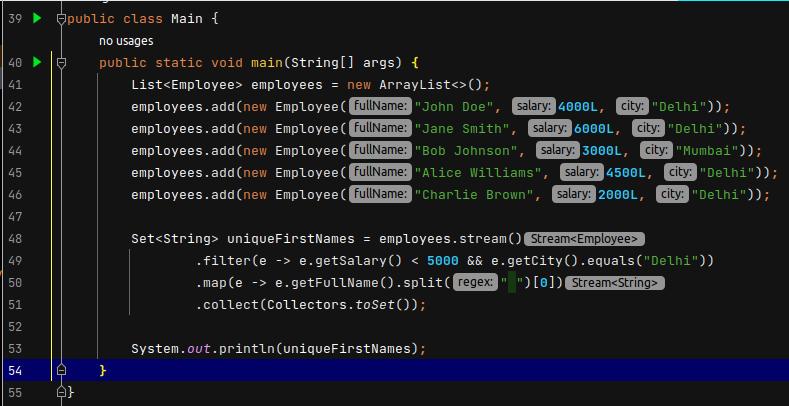
}

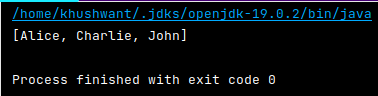
Get list of all unique firstNames of employees where their salary is less than 5000 and who live in delhi.

Note: Full name is concatenation of first name, middle name and last name with single space in between.





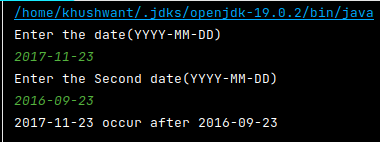




Q6. Using java 8 date/time api:

* WAP to get two dates from user and print if the first date occurs bfore or after the second date supplied by the user.





* WAP to print current date and time in 3 different time zones.

