Write a lambda or method reference for each of the following tasks:

a) Write a lambda that can be used in place of the following anonymous inner class:

new IntConsumer()

{

public void accept(int value)

{

System.out.printf("%d ", value);

}

}

ANS: value -> System.out.printf("%d ", value)

b) Write a method reference that can be used in place of the following lambda:

(String s) -> {return s.toUpperCase();}

ANS: String::toUpperCase

c) Write a no-argument lambda that implicitly returns the String "Welcome to lambdas!".

ANS: () -> "Welcome to lambdas!"

d) Create a one-parameter lambda that returns the cube of its argument.

Operation<Integer> print = value -> System.out.printf("%d\n", value);

print.display(20);

}

private interface Operation<T> {

void display(T t);

}

3..Assuming that list is a List<Employee> in Slide 60, explain in detail the stream pipeline:

listEmployees

.stream()

.filter(e-> e.getSalary() >=800 && e.getSalary() <1200)

.map(e-> new Employee

(e.getId(), e.getName(), e.getDepartment(), getSalary()\*1.02))

.collect(Collectors.

groupingBy(Employee::getDepartment,

Collectors.counting()))

.forEach((d, c) -> System.out.println("Dept: "+ d + ", Count: " + c));

answre:

The call to stream begins the pipeline with a Stream<employee>. The filter: produces a new stream that contains elements of the original stream that pass a given test

and its condition (e-> e.getSalary() >=800 && e.getSalary() <1200) so its produce list contians employees

that their salary greater than or equal 800 and less than 1200.. the map<empoloyee> produces a new stream after applying a function to each element of the original stream

apply this: e-> new Employee(e.getId(), e.getName(), e.getDepartment(), getSalary()\*1.02))

add to salary 1.02 of salarycollect(): repackaging elements to some data structures and applying some additional logic

in this example we get all employees work in specfic department by using (groupingBy)

and their count. forEach(): loops over the stream elements, we print every department and count of employees in this department.