

## CS401 – Modern Programming Practice

### Final Review Points

**Final – Portion: Lesson – 7 to Lesson – 10**

Reference: Lecture slides, Demo codes & Homework Assignments

#### **Important Points for the MPP Exam**

1. The final examination will be held on **Week-4 Thursday Morning**.
2. The final will be timed. It will begin at 9.45 am and will end at 12:15 pm.
- 3. The final should be a closed book. You will get an API handout during the exam and need to return after the exam. You should not write anything in the handout.**
4. Bring a Pencil/Pen, Eraser and the necessary things. You are responsible for keeping your writing desk neat and clean. [ Use waste paper to keep the pencil sharpened dust].
5. Mobile should be turned off. You are not allowed to keep your mobile with you. So bring a backpack to keep your belongings. Keep the backpack in front of the dais.
6. Kindly avoid restroom breaks.

**The exam includes the following**

- a. True or False
- b. Multiple choice questions
- c. Finding output for the given code.
- d. Definitions/Concept Understanding to answer (a) to (c) part
- e. Converting Anonymous Implementation to Lambdas
- f. Converting Lambdas to Method reference
- g. Identify the suitable functional Interface for the given lambda expression.
- h. Making Lambda Library.
- i. Able to convert Imperative code to Declarative code
- j. Coding part from Lesson 8-10.
  - a. Functional coding style using streams, lambdas, and method references to write an answer for the given queries. You need to know how to use intermediate, terminal operations, functionally creating objects.
  - b. Sorting using a more functional style.
  - c. Write a Generic code for the given requirements.

## Practice the following :

### Lessons 7

1. Know about Java 8 interfaces and how to solve Evolving API problems
2. Know about how to add and use static and default methods in an interface?
3. Know about Java 8 Interface methods conflicts & how to solve those issues.
4. Applications of Java 8 interface with enum and for each(Consumer)
5. How do you create a singleton object using enum?
6. Importance of Overriding Object Methods.( toString(), equals() and hashCode())

Refer: package lesson7.lecture.hashcode and your homework Problem 1

### Lesson – 8

7. Functional style of programming principles/features. (Slide 5-7)
8. Benefits of Functional Style.(Slide-9)
9. Concepts of Functional Interface, Functor, Closure.
10. How do you create a Lambdas using a functional API interface?
11. What is a Free variable?
12. How do you create your own functional Interface and implement it using Lambdas?
13. Able to convert imperative style of coding into Functional style of coding.
14. Able to convert Anonymous implementation to Lambdas.
15. Able to convert Method Reference to Lambdas.
16. Practice with the four kinds of Method references.

### Lesson – 9

17. What is Stream?
18. How to Work with Streams? – Creation Stems 3 – Step Template
19. Practice with Intermediate operations(distinct, filter, map, sorted,etc.,)
20. Practice with Terminal operations(forEach, reduce, count, sum etc.,)
21. Practice with Comparator.Comparing().thenComparing(), reversed() to sort your collections.
22. You should know the ways to collect results.
23. How to work with Optional class?( *max, min, findFirst, findAny,get,orElse,isPresent ifpresent*)
24. Create a Lambda Library by replacing it with a suitable functional Interface.
25. Method reference for constructors(Classname::new)

**Refer:** peson.java, for the same class, we applied all the stream operations. Refer to the specific packages for each topic. If you need to solve the query part, refer to homework problems like 8 and 9.

**Skip Parts:** Summary Statistics, Slides: 54 – 62, Collecting result as a Map or Set, Primitive Streams and Optional Primitives.

## **Lesson 10**

- 26. Uses Generics.
- 27. What is Type Erasure?
- 28. Know about the restrictions on Generics. [ Rules 1 – 3 ]
- 29. How do you create a Generic class?
- 30. How to create a Generic method?
- 31. How to use a Wild card? ( Skip Slide 57-60 )
- 32. Know the applications and issues with upper(? extends superclsss) and lower bound(? super subclass).