



## Syllabus

Unit No.	Contents	Mapped CO
I	<b>Generics:</b> What are generics? A simple Generic Example, A generic class with two type parameters. The general form of a generic class, Bounded Types, Using wild card arguments, Creating generic method, Generic Interfaces, Some Generic Restrictions.	CO1, CO2
II	<b>Lambda Expressions:</b> Introducing Lambda Expressions, Block Lambda Expressions, Generic Functional Interfaces, Passing Lambda Expressions as Arguments, Lambda Expressions and Exceptions, Lambda Expressions and Variable Capture.	CO1,CO3
III	<b>Introduction To Collection Framework:</b> Collections Overview, <b>The Collection Interfaces:</b> The collection Interface, The List Interface, The Set Interface, The SortedSet Interface, The NavigableSet Interface, The Queue Interface, The Deque Interface. <b>The Collection Classes:</b> The ArrayList Class, The LinkedList class, The HashSet Class, The LinkedHashSet Class, The TreeSet Class, The PriorityQueue Class, The ArrayDeque Class, The EnumSet Class. Accessing a Collection Via an Iterator, Spliterators.	CO1,CO4
IV	<b>Developing Real World Applications Using Collections:</b> Storing user defined Classes in Collections, The Random Access Interface, Working with Maps, Comparators, The Collection Algorithms, and Arrays Class.	CO1,CO4
V	<b>More Utility Classes:</b> Date, Calendar, Gregorian Calendar, TimeZone, SimpleTimeZone, Locale, Random. <b>Regular Expressions:</b> Regular Expression Processing: Pattern, Matcher, Regular Expression Syntax, Demonstrating Pattern Matching, Two Pattern - Matching Options, Exploring Regular Expressions, Reflection.	CO1,CO5

Unit No.	Experiment Details	Mapped CO's
1.	Demonstrate generic methods with suitable examples.	<b>CO1,CO2,CO3,CO4,CO5</b>
2.	Develop a generic class with suitable examples.	<b>CO1,CO2,CO3,CO4,CO5</b>
3.	Develop a generic program for ordered collection of elements.	<b>CO1,CO2,CO3,CO4,CO5</b>
4.	Implement an efficient data structure for storing and processing non-duplicate elements.	<b>CO1,CO2,CO3,CO4,CO5</b>
5.	Implement a Map-data structure that provides a quick look up to retrieve a value using a key.	<b>CO1,CO2,CO3,CO4,CO5</b>
6.	Use a data structure to prevent data corruption, when two or more threads running concurrently.	<b>CO1,CO2,CO3,CO4,CO5</b>
7.	Take a Java file that contains number of methods. Replace them with lambda expressions. How many lines did it save? Was the code easier to read? Were you able to use method references?	<b>CO1,CO2,CO3,CO4,CO5</b>
8.	Implement various applications using regular expressions.	<b>CO1,CO2,CO3,CO4,CO5</b>

#### Learning Resources

<b>Text Books</b>	1. The Java Complete Reference, Herbert Scheldt, 10 <sup>th</sup> edition, TMH Publications, 2018.
<b>Reference Books</b>	1. Introduction to java programming by Daniel Liang, 10 <sup>th</sup> edition, Pearson. 2. Core Java: An Integrated Approach, New: Includes All Versions up-to Java 8, by R.Nageswara Rao, Dream-Tech Publishers. 3. Head First Java, Kathy Sierra, 2/e, Shroff Publishers, 2012.
<b>e- Resources &amp; other digital material</b>	1. <a href="https://www.geeksforgeeks.org/data-structures/">https://www.geeksforgeeks.org/data-structures/</a> 2. <a href="https://www.youtube.com/watch?v=OIAPZzGSbME">https://www.youtube.com/watch?v=OIAPZzGSbME</a> 3. <a href="https://www.cs.usfca.edu/~galles/visualization/Algorithms.html">https://www.cs.usfca.edu/~galles/visualization/Algorithms.html</a> 4. <a href="https://www.youtube.com/watch?v=S47aSEqm_0I&amp;list=PLgj_VZKxRKrxgFyOutPJpoLFBaQMOpK-">https://www.youtube.com/watch?v=S47aSEqm_0I&amp;list=PLgj_VZKxRKrxgFyOutPJpoLFBaQMOpK-</a> 5. <a href="https://www.geeksforgeeks.org/fundamentals-of-algorithms/">https://www.geeksforgeeks.org/fundamentals-of-algorithms/</a> 6. <a href="https://www.javatpoint.com/collections-in-java">https://www.javatpoint.com/collections-in-java</a>