

<b>Course code</b>	<b>JAVA Programming</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>J</b>	<b>C</b>
<b>CSE 1007</b>		<b>3</b>	<b>0</b>	<b>2</b>	<b>0</b>	<b>4</b>
<b>Pre-requisite</b>		<b>Syllabus version</b>				
		v.1.0				
<b>Course Objectives:</b>						
To understand the core language features of Java and its Application Programming Interfaces (API) for implementing threads, exceptions, database connections, file management and development of dynamic web applications using JavaFX GUI , Servlets and Java Server Pages.						
<b>Expected Course Outcome:</b>						
After successfully completing the course the student should be able to						
<ul style="list-style-type: none"> <li>Implement Java applications to solve real world problems</li> </ul>						
<ul style="list-style-type: none"> <li>Design and build multi-threaded Java applications</li> </ul>						
<ul style="list-style-type: none"> <li>Design, Develop and Deploy dynamic web applications using JavaFX, Servlets and Java Server Pages</li> </ul>						
<b>Student Learning Outcomes (SLO):</b>		<b>1,9,14</b>				
<b>Module:1</b>	<b>Java Basics</b>	<b>5 hours</b>	<b>SLO: 1</b>			
Java design goals - Features of Java Language - JVM - Bytecode - Java source file structure – Basic programming constructs: lexical issues - data types - variables – Java coding standards - operators - control and looping constructs - Arrays – one dimensional and multi-dimensional – enhanced for loop – String, StringBuffer & StringBuilder, Wrapper classes						
<b>Module:2</b>	<b>Object Oriented Programming</b>	<b>7 hours</b>	<b>SLO: 9</b>			
Class Fundamentals – Declaring objects and assigning object reference variables– array of objects – constructors –methods – overloading methods and constructors– “this” keyword – static block - nested class – inner class – garbage collection – finalize(). Inheritance – types - use of “super” - Polymorphism – abstract class – interfaces – packages and sub packages.						
<b>Module:3</b>	<b>Robustness and Concurrency</b>	<b>8 hours</b>	<b>SLO: 9</b>			
Exception Handling - Exceptions & Errors - Types of Exception - Control Flow in Exceptions - Use of try, catch, finally, throw, throws in Exception Handling - user defined exceptions - Multithreading – Thread creation - Life cycle of a Thread- Thread priorities- methods to prevent Thread execution - synchronization – interthread communication – deadlock.						
<b>Module:4</b>	<b>Files, Streams and Collection framework</b>	<b>6 hours</b>	<b>SLO: 1,9</b>			
Java I/O streams – Working with files – Serialization and deserialization of objects –Collection framework – List, Map, Set – Generic classes and methods.						
<b>Module:5</b>	<b>GUI Programming and Database Connectivity</b>	<b>7 hours</b>	<b>SLO: 9,14</b>			
GUI programming using JavaFX, Exploring events, controls and JavaFX menus – Working with JDBC.						

<b>Module:6</b>	<b>Servlet</b>	<b>6 hours</b>	<b>SLO: 9,14</b>
Introduction to servlet – Lifecycle of Servlet - Developing and Deploying Servlets - Exploring Deployment Descriptor (web.xml) - Handling HTTP Request and Response –Servlets with database communication - Session Tracking.			
<b>Module:7</b>	<b>JSP</b>	<b>4 hours</b>	<b>SLO: 1,14</b>
JSP Tags and Expressions - JSP Expression Language (EL) - Using Custom Tag.			
<b>Module:8</b>	<b>Contemporary issues:</b>	<b>2 hours</b>	<b>:</b>
Industry expert talk			
	<b>Total Lecture hours:</b>	<b>45 hours</b>	
<b>Text Book(s)</b>			
1.	Herbert Schildt , The Complete Reference -Java, Tata McGraw-Hill publisher, 10 <sup>th</sup> Edition, 2017.		
<b>Reference Books</b>			
1.	Y. Daniel Liang, “Introduction to Java programming” -comprehensive version-11th Edition, Pearson publisher, 2017		
2.	Cay Horstmann,”Big Java”, 4th edition, John Wiley & Sons publisher, 5 <sup>th</sup> edition, 2015		
3.	E.Balagurusamy, “Programming with Java”, Tata McGraw-Hill publishers, 5 <sup>rd</sup> edition, 2014.		
4	Paul J. Deitel, Harvey Deitel ,Java SE8 for Programmers (Deitel Developer Series) Prentice Hall publisher, 3rd Edition, 2014		
.			
Mode of Evaluation:			
<b>List of Challenging Experiments (Indicative)</b>		<b>SLO: 14</b>	
1.	Write a program to demonstrate the knowledge of students in multidimensional arrays and looping constructs.  Eg., If there are 4 batches in BTech - “CSE1007” course, read the count of the slow learners (who have scored <25) in each batch. Tutors should be assigned in the ratio of 1:4 (For every 4 slow learners, there should be one tutor). Determine the number of tutors for each batch. Create a 2-D jagged array with 4 rows to store the count of slow learners in the 4 batches. The number of columns in each row should be equal to the number of groups formed for that particular batch ( Eg., If there are 23 slow learners in a batch, then there should be 6 tutors and in the jagged array, the corresponding row should store 4, 4, 4, 4, 4,3). Use for-each loop to traverse the array and print the details. Also		

	print the number of batches in which all tutors have exactly 4 students.	
2.	<p>Write a program to demonstrate the knowledge of students in String handling.</p> <p>Eg., Write a program to read a chemical equation and find out the count of the reactants and the products. Also display the count of the number of molecules of each reactant and product.</p> <p>Eg., For the equation, <math>2\text{NaOH} + \text{H}_2\text{SO}_4 \rightarrow \text{Na}_2\text{SO}_4 + 2\text{H}_2\text{O}</math>, the O/P should be as follows.</p> <p>Reactants are 2 moles of NaOH, 1 mole of H<sub>2</sub>SO<sub>4</sub>.</p> <p>Products are 1 mole of Na<sub>2</sub>SO<sub>4</sub> and 2 moles of H<sub>2</sub>O.</p>	
3.	<p>Write a program to demonstrate the knowledge of students in Inheritance.</p> <p>Eg: Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should maintain a minimum balance and if the balance falls below this level, a service charge is imposed.</p> <p>Create a class account that stores customer name, account number and type of account. From this derive the classes cur_acct and sav_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks :</p> <ol style="list-style-type: none"> <li>Accept deposit from a customer and update the balance.</li> <li>Display the balance</li> <li>Compute and deposit interest.</li> <li>Permit withdrawal and update the balance.</li> <li>Check for the minimum balance, impose penalty, necessary, and update the balance.</li> </ol>	
4.	<p>Write a program to demonstrate the knowledge of students in working with user-defined packages and sub-packages.</p> <p>Eg., Within the package named 'primespackage', define a class Primes</p>	

	<p>which includes a method <code>checkForPrime()</code> for checking if the given number is prime or not. Define another class named <code>TwinPrimes</code> outside of this package which will display all the pairs of prime numbers whose difference is 2. (Eg, within the range 1 to 10, all possible twin prime numbers are (3,5), (5,7)). The <code>TwinPrimes</code> class should make use of the <code>checkForPrime()</code> method in the <code>Primes</code> class.</p>	
5.	<p>Write a program to demonstrate the knowledge of students in Java Exception handling.</p> <p>Eg., Read the Register Number and Mobile Number of a student. If the Register Number does not contain exactly 9 characters or if the Mobile Number does not contain exactly 10 characters, throw an <code>IllegalArgumentException</code>. If the Mobile Number contains any character other than a digit, raise a <code>NumberFormatException</code>. If the Register Number contains any character other than digits and alphabets, throw a <code>NoSuchElementException</code>. If they are valid, print the message 'valid' else 'invalid'</p>	
6	<p>Write a program to demonstrate the knowledge of students in multithreading.</p> <p>Eg., Three students A, B and C of B.Tech- II year contest for the PR election. With the total strength of 240 students in II year, simulate the vote casting by generating 240 random numbers (1 for student A, 2 for B and 3 for C) and store them in an array. Create four threads to equally share the task of counting the number of votes cast for all the three candidates. Use synchronized method or synchronized block to update the three count variables. The main thread should receive the final vote count for all three contestants and hence decide the PR based on the values received.</p>	
7	<p>Write a program to demonstrate the knowledge of students in File handling.</p> <p>Eg., Define a class 'Donor' to store the below mentioned details of a blood donor.</p> <p style="padding-left: 40px;">Name, age, Address, Contact number, blood group, date of last donation</p> <p>Create 'n' objects of this class for all the regular donors at Vellore. Write these objects to a file. Read these objects from the file and display only</p>	

	those donors' details whose blood group is 'A+ve' and had not donated for the recent six months.																			
8	<p>Write a program to demonstrate the knowledge of students in working with Java collection framework.</p> <p>Eg., Assume only a maximum of 3 courses can be registered by a student for week end semester classes. Create a hashmap 'h1' with 'n' key-value pairs where keys are the names of students and values are the courses registered by them. Create another hashmap 'h2' with 'm'key-value pairs where keys are the names of courses offered for B.Tech and values are the names of faculty handling the courses. Write appropriate code to</p> <ul style="list-style-type: none"><li>- Add or remove a student from h1</li><li>- Iterate over the maps and display the key-value pairs stored in them</li><li>- Given a student name, fetch the names of all those who teach him/her.</li></ul> <p>Eg:, if the elements of h1 are</p> <table><tr><td>Stud name</td><td>Courses registered</td></tr><tr><td>A</td><td>Python, maths, c</td></tr><tr><td>B</td><td>c, c++</td></tr><tr><td>C</td><td>C++, physics,chemistry</td></tr></table> <p>And if the elements of h2 are</p> <table><tr><td>Course name</td><td>Faculty</td></tr><tr><td>Python</td><td>111</td></tr><tr><td>Maths</td><td>222</td></tr><tr><td>C</td><td>333</td></tr><tr><td>C++</td><td>444</td></tr></table>	Stud name	Courses registered	A	Python, maths, c	B	c, c++	C	C++, physics,chemistry	Course name	Faculty	Python	111	Maths	222	C	333	C++	444	
Stud name	Courses registered																			
A	Python, maths, c																			
B	c, c++																			
C	C++, physics,chemistry																			
Course name	Faculty																			
Python	111																			
Maths	222																			
C	333																			
C++	444																			

	Physics`	555		
	Chemistry	666		
	Digital electronics	777		
	For the student “B”, faculty should be displayed as 333 and 444.			
9	Write a program to demonstrate the knowledge of students in GUI programming using JavaFX.			
10	Write a program to demonstrate the knowledge of students in JDBC. Eg: Create a student table with fields roll number, name, percentage. Insert values in the table. Display all the details of the student table in a tabular format on the screen.			
11	Write a program to demonstrate the knowledge of students in Servlet programming.  Eg., Write a servlet which counts how many times a user has visited a web page. If the user is visiting the page for the first time, display a welcome message. If the user is re-visiting the page, display the number of times visited. (Use cookies)			
12	Write a program to demonstrate the knowledge of students in handling HTTP Request and Response.  Eg: Write a program to create a shopping mall. User must be allowed to do purchase from two pages. Each page should have a page total. The third page should display a bill, which consists of a page total of whatever the purchase has been done and print the total. (Use HttpSession)			
13.	Write a program to demonstrate the knowledge of students in JSP.  Eg: Client sends user name and password to JSP on the Server. Server receives, validates and sends back the validation result to client as response.			
14.	Write a program to demonstrate the knowledge of students in JSP.			

--

	Eg: Create a JSP page for an online multiple choice test. The questions are randomly selected from a database and displayed on the screen. The choices are displayed using radio buttons. When the user clicks on next, the next question is displayed. When the user clicks on submit, display the total score on the screen.			
Mode of evaluation:				
Recommended by Board of Studies		DD-MM-YYYY		
Approved by Academic Council		No. xx	Date	DD-MM-YYYY

### Session-wise Plan

S.No	Topics Covered	Class Hours	Lab Hours	Levels of mastery	Text/Reference Book	Remarks
1	Java source file structure –basic programming constructs - comments – Lexical issues - data types – variables - operators - control and looping	2	2	Familiarity	1	Lab Component
2	Arrays – one dimensional multi-dimensional – enhanced for loop – String, StringBuffer & StringBuilder, Wrapper classes	3	2	Familiarity	1	Lab Component
3	Class Fundamentals - Object & Object reference – array of objects – constructors – methods – overloading – “this” reference – static block - nested class – inner class – garbage collection – finalize()	4	2	Usage	1	Lab Component
4	Inheritance – types - use of “super” - Polymorphism – abstract class – interfaces – packages and sub packages.	3	2	Usage	1	Lab Component
5	Robustness and Concurrency: Exception Handling	4	2	Usage	1	Lab Component



	- Exceptions & Errors - Types of Exception - Control Flow in Exceptions - Use of try, catch, finally, throw, throws in Exception Handling - user defined exceptions						
6	Multithreading – Thread creation – Life cycle of Thread- Thread priorities- - methods to prevent Thread execution - synchronization – interthread communication – deadlock.	4	2	Usage	1	Lab Component	
7	Files, Streams and Object serialization, Data structures: Java I/O streams – Working with files – Serialization and deserialization of objects	3	2	Usage	1	Lab Component	
8	Collection framework – List, Map, Set, Generics	3	2	Assessment	1	Lab Component	
9	GUI programming using JavaFX,	2	2	Usage	1		
10	Exploring events, controls and JavaFX menus	4	2	Usage	1	Lab Component	
11	Working with JDBC	1	2	Usage	1	Lab Component	
12	Introduction to servlet - Servlet life cycle - Developing and Deploying Servlets	3	2	Assessment	1		
13	Exploring Deployment Descriptor	3	2	Usage	1	Lab Component	

	(web.xml) - Handling Request and Response – Accessing Database-Session Tracking & Management.						
14	JSP Tags and Expressions - JSP Expression Language (EL) - Using Custom Tag	4	2	Usage	4	Lab Component	