

# UNIVERSITY OF MUMBAI



Revised syllabus (Rev- 2016) from Academic Year 2016 -17

Under

**FACULTY OF TECHNOLOGY**

## **Information Technology**

**Second Year** with Effect from AY 2017-18

**Third Year** with Effect from AY 2018-19

**Final Year** with Effect from AY 2019-20

**As per Choice Based Credit and Grading System**

with effect from the AY 2016–17

**S. E. Information Technology (Semester-III)**

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Pract	Tut	Theory	TW/Pract	Tut	Total
ITC301	Applied Mathematics III	4+1@	-	-	5	-	-	5
ITC302	Logic Design	4	-	-	4	-	-	4
ITC303	Data Structures & Analysis	4	-	-	4	-	-	4
ITC304	Database Management System	4	-	-	4	-	-	4
ITC305	Principle of Communications	3+1\$	-	-	4	-	-	4
ITL301	Digital Design Lab	-	2	-	-	1	-	1
ITL302	Data Structures Lab	-	2	-	-	1	-	1
IT303	SQL Lab	-	2	-	-	1	-	1
ITL304	Java Programming Lab	-	2+2*	-	-	2	-	2
<b>Total</b>		<b>21</b>	<b>10</b>	<b>-</b>	<b>21</b>	<b>5</b>	<b>-</b>	<b>26</b>

Course Code	Course Name	Examination Scheme								
		Theory					TW	Oral	Oral & Pract	Total
		Internal Assessment			End Sem. Exam	Exam Duration (in Hrs)				
		Test 1	Test 2	Avg.						
ITC301	Applied Mathematics III	20	20	20	80	3	-	-	-	100
ITC302	Logic Design	20	20	20	80	3	-	-	-	100
ITC303	Data Structures & Analysis	20	20	20	80	3	-	-	-	100
ITC304	Database Management System	20	20	20	80	3	-	-	-	100
ITC305	Principle of Communications	20	20	20	80	3	--	-	-	100
ITL301	Digital Design Lab	-	-	-	-	-	25	--	25	50
ITL302	Data Structures Lab	-	-	-	-	-	25	--	25	50
IT303	SQL Lab	-	-	-	-	-	25	-	25	50
ITL304	Java Programming Lab	-	-	-	-	-	50	--	50	100
<b>Total</b>		<b>100</b>	<b>100</b>	<b>100</b>	<b>400</b>	<b>-</b>	<b>125</b>	<b>--</b>	<b>125</b>	<b>750</b>

@ 4 hours shown as theory to be taken class wise and 1 hour to be taken tutorial as class wise

\$ 3 hours shown as theory to be taken class wise and 1 hour to be taken tutorial as batch wise

\* 2 hours shown as practical's to be taken class wise lecture and another 2 hours to be taken as batch wise practices in the lab.

Course Code	Course Name	Theory	Practical	Tutorial	Theory	TW/Practical	Tutorial	Total
ITL304	Java Programming Lab	--	2+2*	--	--	2	--	2

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Oral & Practical	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg. of two Tests				
ITL304	Java Programming Lab	--	--	--	--	50	50	100

\* 2 hours shown as practical's to be taken class wise lecture and other 2 hours to be taken as batch wise practical's in Lab.

**Lab Objectives:** Students will try:

1. To understand how to design, implement, test, debug, and document programs that use basic data types and computation, simple I/O, conditional and control structures, string handling and functions.
2. To understand the importance of Classes & objects along with constructors, Arrays and Vectors.
3. Discuss the principles of inheritance, interface and packages and demonstrate through problem analysis assignments how they relate to the design of methods, abstract classes and interfaces and packages.
4. To understand importance of Multi-threading & different exception handling mechanisms.
5. To learn experience of designing, implementing, testing, and debugging graphical user interfaces in Java using applet and AWT that respond to different user events.
6. To understand Java Swings for designing GUI applications based on MVC architecture.

**Lab Outcomes:** Upon Completion of the course the learner should be able to:

1. Implement Object Oriented programming concept using basic syntaxes of control Structures, strings and function for developing skills of logic building activity.
2. Identify classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem
3. Demonstrates how to achieve reusability using inheritance, interfaces and packages and describes faster application development can be achieved.
4. Demonstrate understanding and use of different exception handling mechanisms and

concept of multithreading for robust faster and efficient application development.

5. Identify and describe common abstract user interface components to design GUI in Java using Applet & AWT along with response to events
6. Identify, Design & develop complex Graphical user interfaces using principal JavaSwing classes based on MVC architecture

Hardware Requirements	Software Requirements	Other Requirements
PC With Following Configuration 1. Intel PIV Processor 2. 2 GB RAM 3. 500 GB Harddisk 4. Network interface card	1. Windows or Linux Desktop OS 2. JDK 1.8 or higher 3. Notepad ++ 4. JAVA IDEs like Netbeans or Eclipse	1. Internet Connection for installing additional packages if required

### Detailed Syllabus:

Sr. No.	Module	Detailed Contents	Hours	LO Mapping
1)	Fundamental of Java Programming	<p><b>Theory</b></p> <p>Overview of procedure and object oriented Programming, Java Designing Goals, Features of Java Language.</p> <p>Introduction to the principles of object-oriented programming: Classes, Objects, Abstraction, Encapsulation, Inheritance, Polymorphism,</p> <p>Keywords, Data types, Variables, Operators, Expressions, Types of variables and methods.</p> <p>Control Statements: If Statement, If-else, Nested if, switch Statement, break, continue. Iteration Statements: for loop, while loop, and do-while loop.</p> <p><b>Experiment 1:</b></p> <p>(Perform any three programs that covers Classes, Methods, Control structures and Looping statements)</p> <p>i) Write a Java program to understand how to accept input using Scanner or</p>	12	LO 1  LO 2

		<p>BufferedReader and print output using System.out.println statement.</p> <p>ii) Write a Java program to display the default value of all primitive data types in Java.</p> <p>iii) Write a Java program that prints all real solutions to the quadratic equation <math>ax^2+bx+c = 0</math>. Read in a, b, c and use the quadratic formula. If the discriminant <math>b^2-4ac</math> is negative, display a message stating that there are no real solutions.</p> <p>iv) Write a java program to test whether string is palindrome or not</p> <p>v) Write a java program to count number of alphabets, digits, special symbols, blank spaces and words from the given sentence.</p> <p>vi) Write a java program to count number of vowels and consonants from the given strings.</p> <p>vii) Write a Menu driven program in java to implement simple banking application. Application should read the customer name, account number, initial balance, rate of interest, contact number and address field etc. Application should have following methods.</p> <ol style="list-style-type: none"><li>1. createAccount()</li><li>2. deposit()</li><li>3. withdraw()</li><li>4. computeInterest()</li><li>5. displayBalance()</li></ol> <p>viii) Write a menu driven Java program which will</p>		
--	--	--	--	--

		<p>read a number and should implement the following methods</p> <ol style="list-style-type: none"> <li>1. factorial()</li> <li>2. reverse()</li> <li>3. testArmstrong()</li> <li>4. testPalindrome()</li> <li>5. testPrime()</li> <li>6. fibonacciSeries()</li> </ol> <p>ix) Write a Java program to demonstrate Method overloading</p>		
2)	<p>Classes, Objects, Arrays and Recursion</p>	<p><b>Theory</b></p> <p>Classes &amp; Objects: Class Fundamentals: Assigning Object Reference Variables, Passing parameters to Methods and Returning parameters from the methods, Nested and Inner Classes.</p> <p>Constructors: Parameterized Constructors, finalize( ) Method, Method overloading, Constructors overloading, Recursion, Command-Line Arguments.</p> <p>Wrapper classes, Java.util.Scanner, Java.io.BufferedReader, Java.io.DataInputStream, Java.io.DataOutputStream and String Buffer classes and String functions.</p> <p>Arrays &amp; Vectors: One Dimensional arrays, Two Dimensional array, Irregular arrays, dynamic arrays, Array List and Array of Object.</p> <p><b>Experiment 2</b></p> <p>(Perform any Five programs that covers Classes &amp; objects, Constructors, Command Line Arguments,</p>	12	<p>LO 1</p> <p>LO 2</p>

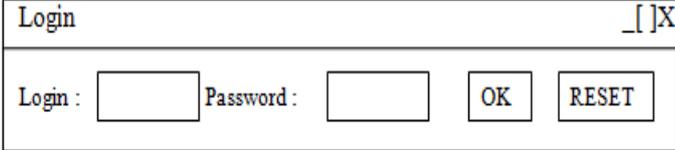
Arrays/Vectors & recursions)

- i) Write a java program to demonstrate Constructors, Parameterized Constructors and Constructor Overloading
- ii) Write a java program to demonstrate Command Line Arguments
- iii) Write a java program to demonstrate String Functions
- iv) Write a java program to demonstrate Array and Vectors operations
- v) Write a java programs to add n strings in a vector array. Input new string and check whether it is present in the vector. If it is present delete it otherwise add it to the vector.
- vi) Write a java programs to test whether the given element is present in the vector array.
- vii) Write a java programs to find frequency of a element in the given Vector array.
- viii) Write a java programs to add n strings in a vector array. Input new string and check whether it is present in the vector. If it is present delete it otherwise add it to the vector.
- ix) Write menu driven program to implement recursive functions for following tasks.
  - a) To find GCD and LCM
  - b) To find  $X^Y$
  - c) To print n Fibonacci numbers

		<p>d) To find reverse of number</p> <p>e) To <math>1+2+3+4+ \dots+(n-1)+n</math></p> <p>x) Write the Menu driven program to perform</p> <p>a) Addition of two matrices of order <math>m*n</math> and <math>p*q</math></p> <p>b) Multiplication of two matrices of order <math>m*n</math> and <math>p*q</math></p> <p>c) Transpose of matrix of order <math>m*n</math></p> <p>d) addition of diagonal and non-diagonal elements</p>		
3)	Inheritance, Interface and Packages	<p><b>Theory</b></p> <p>Inheritance Basics, , Types of Inheritance in Java, Concept of Super and sub class, inheriting Data members and Methods, Role of Constructors in inheritance, Making methods and classes final , Method overriding, Dynamic Method Dispatch, Abstract classes and methods</p> <p>Defining an interface, extending interfaces , implementing interfaces, accessing implementations through interface references, Interfaces vs. Abstract classes.</p> <p>Packages – Steps for defining, creating and accessing a Package, importing packages, Making JAR Files for Library Packages, java.util.Vector</p> <p><b>Experiment 3</b></p> <p>(Perform any Two programs that covers Inheritance, interfaces and packages)</p> <p>i) Write a java programs to demonstrate hierarchical inheritance</p> <p>ii) Write a java program to demonstrate extending &amp; implementing Interfaces</p>	08	LO 3

		<p>iii) Write a java program to demonstrate Modules and packages</p> <p>iv) Write a java program to create user defined packages</p>		
4)	Exception Handling and Multithreading	<p><b>Theory:</b></p> <p>Exception handling Mechanism: try, catch, throw, throws and finally.</p> <p>Multithreading: Need of Multithreading , Java thread Model, thread Life-Cycle, thread class Methods, Implementing Runnable, Extending thread, Synchronizing threads, synchronized Statement, Critical Factor in Thread –Deadlock.</p> <p><b>Experiment 4</b></p> <p>(Perform any Two programs that covers Exception Handling &amp; Multithreading)</p> <p>i) Write java programs to demonstrate Exception handling using try, catch, throw, throws and finally statements.</p> <p>ii) Write a Java Program to input the data through command Line and Find out total valid and in-valid integers. (Hint: use exception handling).</p> <p>iii) Write a Java Program to calculate the Result. Result should consist of name, seatno, date, center number and marks of semester three exam. Create a User Defined Exception class MarksOutOfBoundsException, If Entered marks of any subject is greater than 100 or less than 0, and then program should create a user defined Exception of type MarksOutOfBoundsException and must have a provision to handle it.</p>	06	LO3 LO 4

		<p>iv) Write java program to create a user defined Exception class known as PayOutOfBoundsException. Organization does not offer basic salary less than 8000. If entered salary is less than 8000 then program should create an Exception of Type PayOutOfBoundsException. Program should calculate gross salary by considering salary parameters such as DA, HRA, CA, TA, Professional tax, TDS, PF.. etc</p> <p>v) Write java programs to create user defined threads by extending thread class and by implementing runnable.</p> <p>vi) Write java program to print Table of Five, Seven and Thirteen using Multithreading (Use Thread class for the implementation) .</p> <p>vii) Write a java program to print first 20 prime numbers and 15 Fibonacci numbers by creating two child threads and also print the total time taken by each thread for the execution.</p> <p>viii) Write a java program to implement use of nested try-catch concept using appropriate example.</p> <p>ix) Write java program to create the child thread. Comment on the execution of main and Child Thread.</p> <p>x) Write java program to implement the concept of Thread Synchronization</p> <p>xi) Write a Java program to identify whether inputted data is byte/short/int/long/float/double/String/char type. (Use Exception Handling)</p>		
5)	Applet Programming, GUI	5.1 Applet: Applet fundamentals, Applet lifecycle, Creating applet, paint method Applet tag, Applet class methods.	10	LO3 LO4

development using AWT and Event handling	<p>Designing Graphical User Interfaces in Java, Components and Containers, Basics of Components, Using Containers, Layout Managers, AWT Components, Adding a Menu to Window, Extending GUI Features</p> <p>Event-Driven Programming in Java, Event-Handling Process, Event- Handling Mechanism, Delegation Model of Event Handling, Event Classes, Event Sources, Event Listeners, Adapter Classes as Helper Classes in Event Handling.</p> <p><b>Experiment 5</b></p> <p>(Perform any Three programs that covers Applet Programming, GUI development using AWT and Event handling)</p> <p>i) Write java program to draw the house on an applet.</p> <p>ii) On Applet: Take a Login and Password from the user and display it on the third Text Field which appears only on clicking OK button and clear both the Text Fields on clicking RESET button Perform same using AWT and Swings as well.</p>  <p>iii) Write java program to create an advertisement banner on an applet using multithreading</p> <p>iv) Write java program to create a registration form using AWT.</p> <p>v) Write a Java program to demonstrate the use of AWT components namely buttons, labels, text boxes, lists/combo boxes, menus with event handling.</p>		LO 5
--	--	--	------

		vi) Write a java program to store personal telephone directory in such a way that when user hits a character, the names which starts with the character and telephone numbers should appear.		
6)	Java Swings	<p><b>Theory</b></p> <p>6.1 Introducing Swing: AWT vs Swings, Components and Containers, Swing Packages, A Simple Swing Application, Painting in Swing, Designing Swing GUI Application using Buttons, JLabels, Checkboxes, Radio Buttons, JScrollPane, JList, JComboBox, Trees, Tables Scroll pane Menus and Toolbars</p> <p><b>Experiment 6</b></p> <p>(Perform any one programs that covers concept of Swings)</p> <p>i) Write a Java program to implement Swing components namely Buttons, JLabels, Checkboxes, Radio Buttons, JScrollPane, JList, JComboBox, Trees, Tables Scroll pane Menus and Toolbars to design interactive GUI.</p> <p>ii) Write a program to create a window with four text fields for the name, street, city and pincode with suitable labels. Also windows contains a button MyInfo. When the user types the name, his street, city and pincode and then clicks the button, the types details must appear in Arial Font with Size 32, Italics.</p>	06	LO4 LO 6

**Textbook Books:**

1. Herbert Schildt, "Java-The Complete Reference", Seventh Edition, Tata McGraw Hill Publication
2. E. Balguruswamy, "Programming with java A primer", Fifth edition, Tata McGraw Hill Publication

**Reference Books:**

1. D.T. Editorial Services, "Java 8 Programming Black Book", Dreamtech Press  
University of Mumbai, B. E. (Information Technology), Rev 2016

2. H. M. Deitel, P. J. Deitel, S. E. Santry, "Advanced Java 2 Platform How to Program" Prentice Hall
3. Learn to Master JAVA, from Star EDU solutions, by ScriptDemics

**Term Work:**

The term Work shall consist of at least 12 to 15 practical's based on the above list. The also Term work Journal must include at least 2 assignments.

**Term Work Marks:** 50 Marks (Total marks) = 40 Marks (Experiment) + 5 Marks (Assignments) + 5 Marks (Attendance)

**Oral & Practical Exam:** An Oral & Practical exam will be held based on the above syllabus.