S. Y. B. Voc. (Software Developmen t)

B. Voc. Software Development Syllabus for Second Year

Structure for Semester-I									
Course Code	**Course Name**	**Teaching Scheme (Hours/Week)**		**Examination Scheme and Marks**			**Credits**		
		Theory	**Practical**	**ISE**	**ESE**	**Total**	**TH**	**PR**	**Total**
BVSD-211	Introduction to Object Oriented Concepts using Core Java	03	---	50	50	100	03	---	03
BVSD-212	Introduction to Dot Net Framework and ASP.NET	03	---	50	50	100	03	---	03
BVSD-213	JavaScript using jQuery	03	---	50	50	100	03	---	03
BVSD-214	CSS Programming	03	---	50	50	100	03	---	03
BVSD-215	Lab Course on Core Java	---	04	50	50	100	---	04	04
BVSD-216	Lab Course on ASP.NET	---	04	50	50	100	---	04	04
BVSD-217	Lab Course on JavaScript using jQuery and CSS Programming	---	04	50	50	100	---	04	04
BVSD-218	Mini Project / On Job Training *	---	06	75	75	150	---	06	06
TOTAL		**12**	**18**	**425**	**425**	**850**	**12**	**18**	**30**

Structure for Semester-II									
Course Code	**Course Name**	**Teaching Scheme (Hours/Week)**		**Examination Scheme and Marks**			**Credits**		
		Theory	**Practical**	**ISE**	**ESE**	**Total**	**TH**	**PR**	**Total**
BVSD-221	Programming in Advance Java	03	---	50	50	100	03	---	03
BVSD-222	Introduction to MVC Framework	03	---	50	50	100	03	---	03
BVSD-223	Introduction to Python Programming	03	---	50	50	100	03	---	03
BVSD-224	Artificial Intelligence	03	---	50	50	100	03	---	03
BVSD-225	Lab Course on Advance Java	---	04	50	50	100	---	04	04
BVSD-226	Lab Course on MVC Framework	---	04	50	50	100	---	04	04
BVSD-227	Lab Course on Python Programming	---	04	50	50	100	---	04	04
BVSD-228	Mini Project / On Job Training *	---	06	75	75	150	---	06	06
TOTAL		**12**	**18**	**425**	**425**	**850**	**12**	**18**	**30**

***On Job Training should be carried out in any one subject per semester as per NBVSDC Guide lines for following Skill Sets:**

1. **Software Developer (SSC/Q0501)**

2. **Engineer Trainee (SSC/Q0507)**
3. **User Experience Designer (SSC/Q8404)**
4. **AI - Data Quality Analyst (SSC/Q8101)**
5. **Database Administrator (SSC/Q8109)**

Semester I Syllabus

Subject Name: Introduction to Object Oriented Concepts using Core Java	
Course Code : BVSD-211	**Semester: I**
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**
Credit :03	

Course Objectives :

1. To understand the basic concepts and fundamentals of platform independent object oriented language.
2. To demonstrate skills in writing programs using exception handling techniques and multi-threading.
3. To understand streams and efficient user interface design techniques.

Course Outcomes :

1. Student will be able to use the syntax and semantics of java programming language and basic concepts of OOP..
2. Students will be able to develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages..
3. Students will be to apply the concepts of Multithreading and Exception handling to develop efficient and error free codes.
4. Students will be to design event driven GUI and web related applications which mimic the real word scenarios.

	Contents	**Hours**
1	**Object oriented concepts** 1.1 Object oriented methodology 1.2 Features, advantages and Applications of OOPS	**2**
2	**Object oriented concepts** 2.1 A Short History of Java 2.2 Features of Java 2.3 Comparison of Java and C++ 2.4 Java Tools And Editors(Appletviewer,Jar,Jdb) 2.5 Java Environment	**8**
3	**Object and Classes** 3.1 Defining Your Own Classes and Use of 'this' Keyword. 3.2 Using Predefined Classes 3.3 Object the cosmic class 3.4 Constructor and Overloading Constructors 3.5 Method Parameters 3.6 Static Fields and Methods 3.7 Access Specifiers (public, protected, private, friendly (defualt)) 3.8 Creating Accesses and using Packages 3.9 Wrapper Classes 3.10 Garbage Collection(finalize() Method)	**8**
4	**Inheritance** 4.1 Inheritance Basics (extends Keyword) and Types of Inheritance Superclass, and Subclass and use of Super Keyword	**5**

	4.2 Method Overriding and Use of final keyword related to method and class 4.3 Use of Abstract class	
5	**Interfaces and Inner Classes** 5.1 Defining and Implementing Interfaces 5.2 Object Cloning 5.3 Inner Classes	**4**
6	**Exception Handling** 6.1 Dealing Errors 6.2 Catching exception and exception handling 6.3 Creating user defined exception. 6.4 Using assertion	**6**
7	**Strings, Streams and Files** 7.1 String class and StringBuffer Class 7.2 Stream classes 7.2.1. Byte Stream classes 7.2.2. Character Stream Classes 7.3 Using the File class 7.4 Creation of files 7.5 Reading/Writing characters and bytes 7.6 Handling primitive data type	**8**
8	**User Interface Components with AWT and Swing** 8.1 What AWT ? What is Swing? Difference between AWT and Swing. 8.2 The MVC Architecture and Swing 8.3 Layout Manager and Layouts, The JComponent class 8.4 Components - Buttons and Labels (JButton, JLabel), Checkboxes and Radio Buttons (JCheckBox and JRadioButton), Lists and Combo Boxes (JList and JCombo) along with the JScrollPane Class, Menus – Jmenu and the JPopupMenu Class, JMenuItem and JCheckBoxMenuItem, Scrollbars and Sliders(JScrollBar and JSlider), Dialogs (Message, confirmation, input (like file selection) and options(like color chooser)) 8.5 Event Handling: Event sources, Listeners, Adapters, Anonymous class	**10**
9	**Applet Programming** 9.1 Applet Life Cycle. 9.2 Applet HTML Tags. 9.3 Passing parameters to Applet 9.4 Repaint() and Update() method	**3**

Reference Books:

Herbert Schildt	Complete reference Java

Steven Horlzner	Java 2 programming black books
E. Balagurusamy	Programming with Java , A primer

Subject Name: Introduction to Dot Net Framework and ASP.NET	
Course Code : BVSD-212	**Semester: I**
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**
Credit :03	

Course Objectives :

1. To introduce the Microsoft framework architecture.
2. To develop of console application.
3. To build windows application.
4. To create a web application using .net
5. To develop the website & application

Course Outcomes :

1. Student will be able to create dynamic Web pages by using ASP.NET.
2. Students will be able to create a user interface on an ASP.NET page by using standard Web server controls.
3. Students will be able to create a user control and a custom server control and add them to an ASP.NET page
4. Students will be able to create your own Website, enhanced by using Master pages and Themes
5. Students will be able to find and eliminate bugs in an ASP.NET application
6. Students will be able to display dynamic data from a data source by using ADO.Net and data binding.
7. Students will be able to use Web service to enhance a Web application
8. Students will be able to deploy an ASP.NET application to a production Web server

	Contents	**Hours**
1	**Introduction to Dot Net Framework** 1.1 The Evolution of Web Development 1.2 HTML and HTML Forms 1.3 Server-Side Programming and Client-Side Programming 1.4 C#, VB, and the .NET Languages 1.5 The Common Language Runtime 1.6 The .NET Class Library 1.7 Visual Studio	**3**
2	**Introduction to ASP.NET** 2.1 Introduction of different Web Technology 2.2 What is Asp.Net 2.3 How Asp.Net Works 2.4 Use of visual studio 2.5 Different Languages used in Asp.Net.	**5**
3	**Setting up and Installing ASP.NET** 3.1 Installing Internet Information Server 3.2 Installation of Asp.Net 3.3 virtual directory 3.4 Application Setting in IIS.	**4**

4	**Coding Standards** 4.1 Overview of coding standards follows during programming	**3**
5	**Asp.Net Standard Controls** 5.1 Displaying information Label Controls Literal Controls Bulleted List 5.2 Accepting User Input Textbox controls RadioButton and RadioButtonList Controls CheckBox and CheckBoxList Controls Button controls LinkButton Control ImageButton Control Using Hyperlink Control DropDownList ListBox 5.3 Displaying Images Image Control Image Map Control Using Panel Control Using Hyperlink Control	**8**
6	**Asp.Net Validation Controls** 6.1 Required Field Validator Control 6.2 Regular Expression Validator Control 6.3 Compare Field Validator Control 6.4 Range Validator Control 6.5 Validation Summary Control 6.6 Custom Validator Control	**7**
7	**Designing Websites with master pages** 7.1 Creating master pages 7.2 Creating default contents 7.3 Nesting master pages	**7**
8	**Using the Rich Controls** 8.1 Accepting File Uploads 8.2 Saving files to file system 8.3 Calendar Control	**4**
9	**ADO.NET** 9.1 Introduction to ADO.NET 9.2 Connection string	**4**

Reference Books:

Stephen Walther	ASP.NET 3.5 Unleashed

Imar Spaanjaars	Beginning ASP.NET 3.5: In C# and VB
Dino Esposito	Microsoft ASP.NET and AJAX: Architecting Web Applications

Subject Name: JavaScript using jQuery	
Course Code : BVSD-213	**Semester: I**
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**
Credit :03	

Course Objectives :

1. To understand the JavaScript language & the Document Object Model.
2. To identify the capabilities of jQuery and their role in web design
3. To alter, show, hide and move objects on a web page
4. To check information inputted into a form.
5. To understand the how events work and create interactive content
6. To implementing validation with XHTML forms
7. To design and build rich interactive web applications
8. To install and setup a web page to use jQuery

Course Outcomes :

1. Students will be able to write and troubleshoot JavaScript statements, commands, variables, operators, conditionals, loops, arrays, and functions.
2. Students will be able to implement validation with XHTML forms
3. Students will be able to install and setup a web page to use jQuery
4. Students will be able to reference and manipulate web page content
5. Students will be able to respond to user events using JavaScript and jQuery, creating interactivity.
6. Students will be able to select and modify page elements and create special visual effects and animation using events and jQuery functions.
7. Student will be able to write modern, reactive Websites
8. Students will be able to update web basic website project

	Contents	**Hours**
1	**Introduction to JavaScript** 1.1 HTML and scripting languages 1.2 Where to insert JavaScript in HTML 1.3 The <script> tag 1.4 JavaScript variables and data types 1.5 Numeric and String operators 1.6 Comparison and Logical operators	**10**
2	**JavaScript Comments** 2.1 Single line comments 2.2 Single line comments at the end of a line 2.3 Multiple lines comments 2.4 Single line comment to prevent execution **2.5** Multiple lines comment to prevent execution	**6**
3	**JavaScript Arithmetic and JavaScript Assignment** 3.1 addition (+) operator,subtraction (-) operator ,multiplication (*) operator, division (/) operator, modulus (%) operator, increment (++) operator, decrement (--) operator	**5**

	3.2 Assignment Operators: = assignment operator, += assignment operator, -= assignment operator, *= assignment operator. /= assignment operator, %= assignment operator	
4	**Array** 4.1 Join two arrays - concat() 4.2 Join three arrays - concat() 4.3 Join all elements of an array into a string - join() 4.4 Remove the last element of an array - pop() 4.5 Add new elements to the end of an array - push() 4.6 Reverse the order of the elements in an array - reverse() 4.7 Remove the first element of an array - shift() 4.8 Select elements from an array - slice() 4.9 Sort an array (alphabetically and ascending) - sort() 4.10 Sort numbers (numerically and ascending) - sort() 4.11 Sort numbers (numerically and descending) - sort() 4.12 Add an element to position 2 in an array - splice()	**10**
5	**Basic JavaScript programming concepts** 5.1 The if statement 5.2 The else condition 5.3 The switch and case statements 5.4 The for and for..in loops 5.5 The while and do..while loops 5.6 The break and continue statements 5.7 The label identifier 5.8 Defining functions 5.9 Passing parameters and receiving data from functions 5.10 Variable scope and the var statement	**8**
6	**JavaScript Objects** 6.1 Creating a JavaScript variable 6.2 Creating a JavaScript object 6.3 Creating a JavaScript object (single line) 6.4 Creating a JavaScript object (multiple lines) 6.5 Creating a JavaScript object using new 6.6JavaScript objects are mutable	**8**
7	**jQuery - JavaScript Library** 7.1 Downloading and using jQuery library 7.2 Typical tasks you perform with jQuery 7.3 Selecting elements from the DOM 7.4 Changing element content 7.5 Adding/Changing/Removing attributes 7.6 Adding/Removing/Toggling CSS classes 7.7 Adding/Changing CSS properties 7.8 Adding/Replacing/Removing elements 7.9 jQuery built-in animation effects 7.10 Working with element Set 7.11 Working with arrays and objects	**12**

References:

1. Beginning JavaScript (4th Edition)-------- By Paul Wilton & Jeremy McPeak
2. JavaScript - The Definitive Guide (6th Edition) ------- By David Flanagan

Subject Name: CSS Programming	
Course Code : BVSD-214	**Semester: I**
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**
Credit :03	

Course Objectives :

1. To provide web developers with a standard way to define set of style characteristics
2. To understand CSS grid layout and flexbox
3. To provide web developers with a standard way to manage set of style characteristics
4. To provide capability through a technical model based on a hierarchical scope of effect.
5. To provide capability through a technical model based on the separation of a style from content, a well-defined set of published standards

Course Outcomes :

1. Students will be able to define CSS.
2. Students will be able to set up web pages with CSS
3. Students will be able to using CSS for styling text, font, and properties
4. Students will be able to styling page backgrounds
5. Students will be able to styling lists in CSS

	Contents	**Hours**
1	**Introduction to CSS** 1.1 What is CSS 1.2 CSS solved a big problem 1.3 CSS saves a lot of work 1.4 CSS syntax 1.5 CSS example 1.6 CSS comments	**2**
2	**CSS Selectors** 2.1 Element 2.2 ID 2.3 Class 2.4 Grouping	**1**
3	**Inserting CSS** 3.1 Inline CSS 3.2 Internal CSS 3.3 External CSS	**3**
4	**Properties of CSS** 4.1 Background 4.2 Text 4.3 Fonts 4.4 Links 4.5 lists 4.6 Position 4.7 Float	**5**

5	**CSS Tables** 5.1 Borders 5.2 Collapse borders 5.3 Width 5.4 Height 5.5 Color 5.6 Alignment 5.7 Padding	**2**
6	**CSS Box Model** 6.1 Margin 6.2 Border 6.3 Content 6.4Padding	**5**
7	**CSS Display** 7.1 Display Property 7.2 Block level elements 7.3 Inling elements 7.4 Display : none 7.5 Override default display value 7.6 Hide an element	**2**
8	**CSS Combinators**	**1**
9	**CSS Pseudo** 9.1 Pseudo Classes 9.2 Pseudo elements	**2**
10	**CSS Navigation bar**	**1**
11	**CSS Image gallery**	**1**
12	**Introduction to CSS3** 12.1 Introduction 12.2 Rounded corners 12.3 Gradients 12.4 Shadows 12.5 2D Transforms 12.6 3D Transforms 12.7 Transitions 12.8 Animations	**5**

References:

1. Functional CSS : Dynamic HTML without Javascript-------- By Dmintry Nesterkin
2. Web Design with HTML, CSS, JavaScript and jQuery Set------- By Jon Duckett
3. www.w3schools.com

Subject Name: Lab Course on Core Java	
Course Code : BVSD-215	**Semester: I**
Weekly Teaching Hours: PR: 01 Tut: 00	**Scheme of Marking TH:**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --** PR: 50, IA: 50, Total: 100
Credit :04	

No	List of Experiments
1	Assignment to demonstrate Java Basics using Java tools, javac, java, javap, javadoc, jdb.
2	Assignment to demonstrate creating objects, using new, static keyword, final. Setting the class path, constructors
3	Assignment to demonstrate Packages
4	Assignment to demonstrate Arrays of Objects Console I/O
5	Assignment to demonstrate Scanner.BufferedReader.Wrapper classes.
6	Assignment to demonstrate Inheritance.
7	Assignment to demonstrate Interfaces.
8	Assignment to demonstrate Exception Handling User define exceptions & use of keywords.
9	Assignment to demonstrate File Handling I.
10	Assignment to demonstrate File Handling II.
11	Assignment to demonstrate GUI Designing / AWT I.
12	Assignment to demonstrate GUI Designing / AWT II.
13	Assignment to demonstrate Event Handling I.
14	Assignment to demonstrate Event Handling II.
15	Assignments to demonstrate Applet Creation of an applet. Runtime parameter passing

Subject Name: Lab Course on ASP.NET	
Course Code : BVSD-216	**Semester: I**
Weekly Teaching Hours: PR: 01 Tut: 00	**Scheme of Marking TH:**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --** PR: 50, IA: 50, Total: 100
Credit :04	

No	List of Experiments
1	Assignment to demonstrate installation of .NET framework.
2	Assignment to demonstrate introduction to .NET framework.
3	Assignment to demonstrate application using controls label, textbox and button.
4	Assignment to demonstrate application using controls radio button and radio button list controls.
5	Assignment to demonstrate application using controls checkbox and checkboxlist.
6	Assignment to demonstrate application using controls linkbutton, imgbutton, and hyperlink.
7	Assignment to demonstrate application using controls dropdownlist and listbox.
8	Assignments to demonstrate application to display image using image control and image map control.
9	Assignments to demonstrate application to display image using panel control and hyperlink control.
10	Assignment to demonstrate application using validation control (Required field validator control ,regular expression validator control ,compare field validator control).
11	Assignment to demonstrate application using validation control (Range validator control , validation summary control).
12	Assignment to demonstrate application using master pages.
13	Assignment to demonstrate application using file upload.
14	Case study - I
15	Case study - II

Subject Name: Lab Course on JavaScript using jQuery	
Course Code : BVSD-218	**Semester: I**
Weekly Teaching Hours: PR: 01 Tut: 00	**Scheme of Marking TH:**
Exam Duration: 03 Hours	**Scheme of Marking PR: --** PR: 50, IA: 50, Total: 100
Credit :04	

No	List of Experiments
1	Assignment to demonstrate installation of .NET framework.
2	Assignment to demonstrate introduction to .NET framework.
3	Assignment to demonstrate application using controls label, textbox and button.
4	Assignment to demonstrate application using controls radio button and radio button list controls.
5	Assignment to demonstrate application using controls checkbox and checkboxlist.
6	Assignment to demonstrate application using controls linkbutton, imgbutton, and hyperlink.
7	Assignment to demonstrate application using controls dropdownlist and listbox.
8	Assignments to demonstrate application to display image using image control and image map control.
9	Assignments to demonstrate application to display image using panel control and hyperlink control.
10	Assignment to demonstrate application using validation control (Required field validator control ,regular expression validator control ,compare field validator control).
11	Assignment to demonstrate application using validation control (Range validator control , validation summary control).
12	Assignment to demonstrate application using master pages.
13	Assignment to demonstrate application using file upload.
14	Case study - I
15	Case study - II

Semester

II

Syllabus

Subject Name: Programming in Advance Java	
Course Code : BVSD-221	**Semester: I**
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**
Credit :03	

Course Objectives :

1. To provide the ability to design console based, GUI based and web based applications.
2. To understand integrated development environment to create, debug and run multi-tier and enterprise-level applications
3. To understand advanced technology in Java such as Internationalization, and Remote method Invocation
4. To understand how to work with JavaBeans.
5. To understand Java Servlet and Java Server Pages technology
6. To understand the knowledge of object-oriented paradigm in the Java programming language

Course Outcomes :

1. Students will be able to gain the knowledge of J2EE architecture
2. Students will be able to create dynamic web pages, using Servlets and JSP.
3. Students will be able to build Web Applications using Java Servlet API and Java Server Pages
4. Students will be able to apply event handling on AWT and Swing components
5. Students will be able to understand the multi-tier architecture of web-based enterprise applications using Enterprise JavaBeans (EJB).
6. Students will be able to develop network enabled application using Sockets
7. Students will be able to make a reusable software component, using Java Bean
8. Students will be able to identify Java code utilities in applets, Java packages, and classes

	Contents	**Hours**
1	**JDBC** 1.1 The design of JDBC 1.2 Basic JDBS program Concept 1.3 Drivers 1.3 Making the Connection, Statement, ResultSet 1.4 Executing SQL commands 1.5 Executing queries	**6**
2	**Multi-Threading** 2.1 Threading basics 2.2 Life cycle of thread 2.3 Creating Threads 2.4 Priorities and Synchronization	**6**
3	**Collection Framework** 3.1 Collection Interface List , sets 3.2 Sorted set	**8**

	3.3 Collection classes 3.4 Linked list 3.5 Array list 3.6 Vectors 3.7 Hash set 3.8 Tree set 3.9 Using Iterators and enumerations 3.10 Working with maps 3.11 Map interfaces 3.12 Map classes	
4	**Servlet** 4.1 Introduction 4.2 Life cycle of servlet 4.3 Types of servlet 4.4 Session Tracking 4.5 Cookie class 4.6 Servlet- Jdbc	**6**
5	**Remote Method Invocation** 5.1 Introduction to remote object 5.2 RMI architecture 5.3 Stubs and skeleton 5.4 Registry 5.5 Setting up RMI 5.6 Using RMI with applet	**7**
6	**Introduction to JSP** 6.1 Components of JSP – Directives, Tags, Scripting Elements 6.2 Building a simple application using JSP	**5**
7	**Java Beans** 7.1 What is bean 7.2 Advantages 7.3 Using Bean Development kit(BDK) 7.4 Introduction to jar and manifest files 7.5 The java beans API	**7**

Recommended Books

1. Java the Complete Reference, ninth edition by Herbert Schild, Publisher: McGraw Hills
2. Head First EJB 3.0 by Kathy Sierra, Bert Bates, Publisher: O'Reilly Media
3. Head First Servlets and JSP by Bryan Basham, Kathy Sierra & Bert Bates, Publisher: O'Reilly Media
4. Just Hibernate, A Lightweight Introduction to the Hibernate Framework by Madhusudhan Konda, Publisher: O'Reilly Media
5. Programming Jakarta Struts, 2nd Edition by Chuck Cavaness, Publisher: O'Reilly Media

BVSD-222 - Introduction to MVC Framework

Credits: 04
Objectives -:

- Do web development perfectly based on the ASP.NET Framework
- Build open source scalable Web applications
- Create highly professional and dynamic web pages and websites

Subject Name: Introduction to MVC Framework	
Course Code : BVSD-222	**Semester: I**
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**
Credit :03	

Course Objectives :
1. To understand architecture of ASP.NET MVC web applications.
2. To understand controllers containing action methods to process HTTP requests
3. To understand the role of Model, View and Controller in integrating them to develop a complete web application and Request life cycle.
4. To understand Layout View, Sections and Partial Views.
5. To understand validation framework for both client and server validations
6. To understand how to integrate ASP.NET Web Forms and ASP.NET MVC within one web application
Course Outcomes :
1. Students will be able to understand the benefits of MVC design over traditional ASP.NET Web Forms
2. Students will be able to applying validation framework for both client and server validations
3. Students will be able to explain role of Model, View and Controller
4. Students will be able to develop a complete web application
5. Students will be able to implementing ASP.NET Identities in ASP.NET MVC applications
6. Students will be able to use view models for validations
7. Students will be able to break a project into multiple modules using ASP.NET MVC Areas
8. Students will be able to build and deploy ASP.NET MVC application to the production server

	Contents	**Hours**
1	**Introduction to ASP.NET MVC** 1.1 The role of the Model, View, and Controller 1.2 Key benefits of ASP.NET MVC	**2**
2	**Getting Started with ASP.NET MVC** 2.1 ASP.NET MVC project templates 2.2 Understanding the structure of an ASP.NET MVC project	**3**

	2.3 Naming conventions 2.4 Creating views 2.5 Defining controllers 2.6 Defining a data model	
3	**Creating a Complete ASP.NET MVC 4 Application** 3.1 Creating strongly-typed views 3.2 Understanding URLs and action methods 3.3 Using HTML helpers 3.4 Handling form post-backs 3.5 Data validation	**3**
4	**Using the Razor View Engine** 4.1 Getting started with Razor 4.2 Razor design goals 4.3 Implementing a Razor view 4.4 Razor syntax 4.5 Accessing Model Data in Razor views	**3**
5	**Industrial-Strength ASP.NET MVC Applications** 5.1 ASP.NET application architecture best practices 5.2 Implementing a Repository and Entity Framework Data Model 5.3 Using Dependency Injection 5.4 Implementing a custom controller factory	**4**
6	**View Techniques** 6.1 Defining and using custom HTML Helpers 6.2 Defining a layout / MVC Master Page 6.3 Using Styles 6.4 Defining and using partial views 6.5 Razor Helper Method syntax	**4**
7	**Implementing Navigation in MVC web apps** 7.1 Defining view-model classes 7.2 Implementing Data Filtering in a Controller 7.3 Understanding the Routing mechanism 7.4 Adding custom entries to a route table 7.5 Defining defaults, parameters, and validation 7.6 Generating URLs and Hyperlinks 7.7 Custom Route constraints	**5**
8	**MVC State Management** 8.1 Using hidden fields 8.2 Session and Application State 8.3 Custom model bindings	**2**
9	**Additional Techniques** 9.1 View Scaffold Templates 9.2 Controller Scaffold Templates 9.3 Global Filters 9.4 Model binding 9.5 Asynchronous Controllers	**4**
10	**Using AJAX and jQuery with ASP.NET MVC**	**4**

	10.1 Overview of AJAX and ASP.NET MVC 10.2 Unobtrusive AJAX 10.3 Using AJAX Action Links 10.4 Overview of jQuery 10.5 jQuery Techniques 10.6 Using jQuery UI	
11	**ASP.NET MVC & LINQ - working with Data** 11.1 Language features used in LINQ 11.2 Creating simple LINQ queries 11.3 Using LINQ queries in a Web application; 11.4 Defining a data repository; 11.5 Using LINQ to define a data access component	**4**
12	**ASP.NET MVC 4 Techniques & Best Practices** 12.1 View scaffold templates 12.2 Controller scaffold templates 12.3 Dependency injection	**3**
13	**ASP.NET Web API with MVC** 13.1 Overview of the ASP.NET Web API 13.2 Building servers and clients 13.3 Content negotiation 13.4 Validation 13.5 Query able Interfaces 13.6 Dependency Injection	**4**

Recommended Books:

- Beginning ASP.NET 4.5: in C# and VB by Imar Spaanjaars
- Professional ASP.NET 4.5 in C# and VB by Jason N. Gaylord, Christian Wenz, Pranav Rastogi, Todd Miranda, Scott Hanselman
- Pro ASP.NET 4 in C# 2010 by Matthew MacDonald

Subject Name: Introduction to Python Programming		
Course Code : BVSD-223	**Semester: I**	
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**	
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**	
Credit :03		
Course Objectives :		
1. To understand basic principles of computers		
2. To study the programming basics (operations, control structures, data types, etc.)		
3. To apply various data types and control structure		
4. To understand class inheritance and polymorphism		
5. To understand the object-oriented program design and development		
Course Outcomes :		
1. Students will be to identify/characterize/define a problem		
2. Student will be able to use the Python programming language.		
3. Students will be able to use different control structure.		
4. Students will be to begin to implement code.		

	Contents	**Hours**
1	**Introduction to Python Language** 1.1 The Python Programming Language, History, features, Applications, Installing Python, Running Simple Python program 1.2 Basics of Python – 1.2.1. Variables 1.2.2. Constants 1.2.3. Python identifiers and reserved words 1.2.4. Lines and indentation, multi-line statements 1.2.5. Comments 1.2.6. Input/output with print and input functions, 1.2.7. Operations on Data such as assignment, arithmetic, relational, logical and bitwise operations 1.2.8. Command line arguments and processing command line arguments 1.3 Standard data types - basic, none, Boolean (true & False), numbers, 1.4 Strings 1.4.1. Concept, escape characters 1.4.2. String special operations 1.4.3. String formatting operator 1.4.4. Single quotes, Double quotes, Triple quotes 1.4.5. Raw String, Unicode strings, Built-in String methods.	**8**
2	**Language Component** 2.1 Introduction, 2.2 Control Flow and Syntax, 2.3 The if Statement,	**9**

	2.4 while Loop, break and continue, 2.5 for Loop, 2.6 Python Lists - concept, creating and accessing elements, updating & deleting lists, basic list operations, reverse 2.7 built-in List functions 2.8 Using Lists as stacks and Queues, List comprehensions 2.9 Functional programming tools - filter(), map(), and reduce() 2.10 Tuples 2.12.1 Creating & deleting tuples 2.12.2 Accessing values in a tuple 2.12.3 Updating tuples, delete tuple elements 2.12.4 Basic tuple operations 2.12.5 Indexing, slicing and Matrices, built- in tuple functions. 2.11 Sets - Concept, operations 2.12 Dictionaries 2.12.1 Creating and accessing values in a dictionary 2.12.2 Updating dictionary, delete dictionary elements 2.12.3 Properties of dictionary keys 2.12.4 built-in dictionary functions and methods	
3	**Functions and Modules** 3.1 Introduction 3.2 Defining Functions 3.3 Function Parameters 3.4 Scope of Variable 3.5 Function Documentation 3.6 Variable Number of Arguments 3.7 Keyword and Optional Parameters 3.8 Order of arguments (positional, extra & keyword) 3.9 Passing Collections to a Function 3.10 Mapping Functions in a Dictionary 3.11 Lambda 3.12 Modules 3.13 Standard Modules – sys 3.14 Standard Modules – math 3.15 Standard Modules – time 3.16 The dir Function	**8**
4	**Object and Classes** 4.1 Classes in Python 4.2 Principles of Object Orientation 4.3 Creating Classes 4.4 Instance Methods 4.5 File Organization	**8**
5	**I/O and Error Handling in Python** 5.1 Introduction 5.2 Data Streams	**12**

	5.3 Creating Your Own Data Streams 5.4 Access Modes 5.5 Writing Data to a File 5.6 Reading Data from a File 5.7 Additional File Methods 5.8 Using Pipes as Data Streams 5.9 Handling IO Exceptions 5.10 Working with Directories 5.11 Metadata 5.12 Errors 5.13 Run Time Errors 5.14 The Exception Model 5.15 Exception Hierarchy 5.16 Handling Multiple Exceptions	

Reference Books:

1. Introducing Python- Modern Computing in Simple Packages – Bill Lubanovic, O„Reilly Publication
2. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress
3. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries, et al., Pragmatic Bookshelf, 2/E 2014
4. Introduction to Computer Science Using Python- Charles Dierbach, Wiley Publication Learning with Python ", Green Tea Press, 2002

E-Books:

1. python_tutorial. pdf, python_book_01.pdf
2. Beginning Programming with Python for Dummies Paperback – 2015 by John Paul Mueller
3. A Beginner"s Python Tutorial: http://en.wikibooks.org/wiki/A Beginner%27s Python Tutorial.

Subject Name: Artificial Intelligence		
Course Code : BVSD-224	**Semester: I**	
Weekly Teaching Hours: TH: 03 Tut: 00	**Scheme of Marking TH: 50 IA: 50 Total: 100**	
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --**	
Credit :03		
Course Objectives :		
1. To learn various types of algorithms useful in Artificial Intelligence (AI).		
2. To convey the ideas in AI research related to emerging technology.		
3. To introduce ideas and techniques underlying the design of intelligent computer systems		
Course Outcomes :		
1. Students will be to apply the suitable algorithms to solve AI problems		
2. Student will be able to Identify and apply suitable Intelligent agents for various AI applications		
3. Students will be able to build smart system using different informed search / uninformed search or heuristic approaches		
4. Students will be to represent complex problems with expressive language of representation identify/characterize/define a problem.		
	Contents	**Hours**
1	**Introduction to Artificial Intelligence** 1.1 Introduction to AI 1.2 Comparison of AI, Machine Learning, Deep Learning 1.3 Applications of AI 1.4 AI Techniques 1.5 Intelligent Agents, Agents and Environments, Structure of Agents	**4**
2	**Problems, Problem Spaces and search** 2.1 Defining problem as a State Space Search 2.2 Production System 2.3 Problem Characteristics 2.4 Search & Control Strategies 2.5 Problems – Water Jug problem, Missionary Cannibal Problem, Block words Problem, Monkey & Banana problem	**5**
3	**Searching Algorithms** 3.1 Uninformed Search Algorithms/Blind Search Techniques 3.2.1 Breadth-first Search 3.2.2 Depth-first Search 3.2 Informed (Heuristic) search Techniques 3.2.1 Generate-and-test 3.2.2 Simple Hill Climbing 3.2.3 Best First Search 3.2.4 Constraint Satisfaction 3.2.5 Means End Analysis 3.2.6 A* and AO*	**6**
4	**Knowledge Representation**	**7**

	4.1 Definition of Knowledge 4.2 Types of knowledge (Procedural and Declarative knowledge) 4.3 Approaches to Knowledge Representation 4.4 Knowledge representation using Propositional and Predicate logic 4.5 Conversion to clause form 4.6 Resolution in Propositional logic 4.7 Resolution in Predicate logic	
5	**Slot and Filler Structures** 5.1 Weak structures (Semantic networks and Frame) 5.2 Strong structures (Conceptual dependencies and Script)	**4**
6	**Recent Trends in AI and Applications** 6.1 Introduction to Machine Learning, 6.2 Types of Learning, (Supervised, Unsupervised and Reinforcement Learning), 6.3 Predictive Analytics (Weather Forecasting) 6.4 AI-Powered Chatbots (SBI card chatbot (ILA))	**4**

Reference Books:

1. Artificial Intelligence, Tata McGraw Hill, Elaine Rich and Kevin Knight
2. Computational Intelligence, Eberhart, Elsevier, ISBN 9788131217832
3. Artificial Intelligence: A New Synthesis, Nilsson, Elsevier, ISBN 9788181471901
4. Introduction to Artificial Intelligence and Expert System, Dan Patterson, Prentice Hall of India
5. Pvt. Ltd., New Delhi, 1997
6. Artificial Intelligence: A Modern Approach, Russel &Norvig, Pearson Education
7. Introduction to Machine Learning, Ethem Alpaydin, PHI

E-References:

1. https://www.oracle.com/in/chatbots/what-is-a-chatbot/
2. https://www.dataversity.net/case-study-predictive-analytics-and-data-science-keep-an-eye-onthe-weather/
3. https://www.senseforth.ai/conversational-ai-case-studies/SBI-Cards/

Subject Name: Lab Course on Advanced Java	
Course Code : BVSD-225	**Semester: II**
Weekly Teaching Hours: PR: 01 Tut: 00	**Scheme of Marking TH:**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --** PR: 50, IA: 50, Total: 100
Credit :04	

No	List of Experiments
1	Graphics Programming using Swing
2	Multithreading
3	Database programming using JDBC
4	Implementation of list in Collection
5	Implementation of Set
6	Implementation of ArrayList
7	Servlets
8	Cookies
9	Java Server Pages
10	Networking
11	Implementation of RMI
12	Java Netbeans
13	Connection Oriented Transmission –Stream Socket Class
14	Case Study-I
15	Case Study -II

Subject Name: Lab Course on MVC Framework	
Course Code : BVSD-226	**Semester: II**
Weekly Teaching Hours: PR: 01 Tut: 00	**Scheme of Marking TH:**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --** PR: 50, IA: 50, Total: 100
Credit :04	

No	List of Experiments
1	Assignment on Understanding the structure of an ASP.NET MVC project
2	Assignment on Using the Razor View Engine
3	Assignment on Implementing a Repository and Entity Framework Data Model, custom controller factory
4	Assignment on View Techniques
5	Assignment on Implementing Navigation in MVC web apps
6	Assignment on Additional Techniques
7	Assignment on Using AJAX and jQuery
8	Assignment on working with Data
9	Assignment on Web API with MVC

Subject Name: Lab Course on Python Programming	
Course Code : BVSD-227	**Semester: II**
Weekly Teaching Hours: PR: 01 Tut: 00	**Scheme of Marking TH:**
TH Exam Duration: 03 Hours	**Scheme of Marking PR: --** PR: 50, IA: 50, Total: 100
Credit :04	

No	List of Experiments
1	Write python program to print Hello World
2	Write python program to Hello World using string variable
3	Write python program to store data in list and then try to print them.
4	Write python program to do basic trim and slice on string.
5	Write python program to print list of numbers using range and for loop
6	Write python program to store strings in list and then print them.
7	Write a program that returns a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.
8	Write python program to let user enter some data in string and then verify data and print welcome to user.
9	Write python program in which an function is defined and calling that function prints Hello World
10	Write python program in which an function(with single string parameter) is defined and calling that function prints the string parameters given to function.
11	Write python program in which an class is define, then create object of that class and call simple print function define in class.
12	Create a dictionary (in your file) of names and birthdays. When you run your program it should ask the user to enter a name, and return the birthday of that person back to them.
13	Write a Python class named Rectangle constructed by a length and width and a method which will compute the area and perimeter of a rectangle. –
14	Write a Python function that takes a list and returns a new list with unique elements of the first list.
15	Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle