Department of Higher Education

**National Education Policy - 2020
(NEP-2020)**

Model Syllabus for Bachelor of Science (B.Sc) in Computer Science and Bachelor of Computer Application (B.C.A)

(III & IV Semester)

Submitted to
Karnataka State Higher Education Council
Department of Higher Education
Government of Karnataka
Bengaluru.

Curriculum Design/Syllabus Framing Committee

Sl. No	Name	Designation
1.	Dr. Siddu P. Algur Vice Chancellor, VSK University, Ballari	Chairperson
2.	Dr. D.S. Guru Professor, University of Mysore, Mysuru	Member
3.	Dr. Shivashankar S Professor, Karnatak University, Dharwad.	Member
4.	Dr. B. L. Muralidhara Professor, Bangalore University, Bengaluru.	Member
5.	Dr. D.H. Manjaiah Professor, Mangalore University, Mangalore	Member
6.	Dr. Prabhakar C.J. Professor, Kuvempu University, Shankaraghatta	Member
7.	Dr. Aziz Makandar Professor, KSAW University, Vijayapura	Member
8.	Dr. Shivanand Gornale Professor, Rani Channamma University, Belagavi	Member
9.	Dr. Bhagyavana S Mudigoudra Assoc. Professor, Maharani Cluster University, Bengaluru	Member
10.	Dr. Ramesh Reddy Assoc. Professor, Nrupathunga University, Bengaluru.	Member
11.	Dr. Vijayakumar Kulkarni, Assoc. Professor, GFGC for Women, Jewargi Colony, Kalaburgi	Member
12.	Smt. Dr. Sharada Bhat Assoc. Professor, GFGC, Arikola	Member
13.	Shri. Ramchandra Adiga, Assoc Professor GFGC for Women, Ajjarakadu, Udupi Dist.	Member
14	Dr.Dayanand G.Savakar Professor, Dept. Computer Science, Rani Channamma University, Belagavi	Member
15	Dr. Santhosh Deshpande Professor, Dept. Computer Science, VTU, Belagavi	Member
16	Shri Shivanand S. Rumma Asst.Professor, Dept. of PG Studies and Research in Computer Science, Gulbarga University, Kalaburgi	Member
17	Shri. Anand V. Patil DMSM College of Computer Applications, Belagavi	Member
18	Smt. Deepa S. Garag Karnataka State Higher Education Academy. Dharwad.	Member
19	Dr. M. Jayappa Special Officer, KHSEC, Bengaluru	Member Convener

Index:

Model Curriculum Structure

Program: BSc (Basic and Honors)   **Subject: Computer Science**
                                                For

1. Computer Science as MAJOR with another Subject as MINOR (Table IIA of Model Curriculum)

2. Computer Science as MAJOR with another Subject also as MAJOR (Table IIIA of Model Curriculum)

3. Computer Science as MINOR with another Subject as MAJOR (As per Table IIA of Model Curriculum)

Sem	Discipline Specific Core Courses (DSC)	Hour / Week		DS Elective Courses	Hours/ Week
		Theory	Lab		
III	**DSC-3:** Object Oriented Programming Concepts and Programming in JAVA	4			
	DSC-3 Lab: JAVA Lab		4		
IV	**DSC-4:** Database Management Systems	4			
	DSC-4 Lab: DBMS Lab		4		

Course Title: Object Oriented Programming Concepts and Programming in Java	Course code: DSC3
Total Contact Hours: 52	Course Credits: 04
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Explain the object-oriented concepts and JAVA.
- Write JAVA programs using OOP concepts like Abstraction, Encapsulation, Inheritance and Polymorphism.
- Implement Classes and multithreading using JAVA.
- Demonstrate the basic principles of creating Java applications with GUI.

DSC3: Object Oriented Programming Concepts and Programming in Java

Unit	Description	Hours
1	**Introduction to Java:** Overview of Java, JVM, Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, Math class, Arrays in java.	09
2	**Objects and Classes:** Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, Inbuilt classes like String, Character, String Buffer, File, this reference.	10
3	**Inheritance and Polymorphism:** Inheritance in java, Super and sub class, Overriding, Object class, Polymorphism, Dynamic binding, Instance of operator, Abstract class, Interface in java, Package in java, LANG and UTIL package.	09
4	**Event and GUI programming:** Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Applet and its life cycle.	12
5	**I/O programming:** Text and Binary I/O, Binary I/O classes, Object I/O, Random Access Files. Multithreading in java: Thread life cycle and methods, Runnable interface, Thread synchronization, Exception handling with try catch-finally.	12

References:
1. Programming with Java, By E Balagurusamy – A Primer, 4th Edition, McGraw Hill Publication.
2. Core Java Volume I – Fundamentals, By Cay S. Horstmann, Prentice Hall.
3. Object Oriented Programming with Java: Somashekara M.T., Guru, D.S., Manjunatha K.S, 1st Edition, PHI Learning 2017.
4. Java 2 - The Complete Reference, Herbert Schildt, 5th Edition, McGraw Hill Publication, 2017.
5. Java - The Complete Reference, Herbert Schildt, 7th Edition, McGraw Hill Publication, 2017.

Course Title: JAVA LAB	Course code: DSC3LAB
Total Contact Hours: 52	Hours/week : 04
Formative Assessment Marks: 25	Course Credits: 02
Exam Marks: 25	Duration of Exam: 03 Hours

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Implement Object Oriented programming concept using basic syntaxes of control Structures
- Identify classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem
- Demonstrates how to achieve reusability using inheritance
- Demonstrate understanding and use of interfaces, packages, different exception handling mechanisms and concept of multithreading for robust faster and efficient application development.
- Identify and describe common user interface components to design GUI in Java using Applet & AWT along with response to events

Programming Lab

PART A: Java Fundamentals OOPs in Java

1. Program to assign two integer values to X and Y. Using the 'if' statement the output of the program should display a message whether X is greater than Y.
2. Program to list the factorial of the numbers 1 to 10. To calculate the factorial value, use while loop. (Hint Fact of 4 = 4*3*2*1)
3. Write a Java program to find the GCD of number.
4. Program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
5. Program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.
6. Program with class variable that is available for all instances of a class. Use static variable declaration. Observe the changes that occur in the object's member variable values.
7. a. To find the area and circumference of the circle by accepting the radius from the user.
   b. To accept a number and find whether the number is Prime or not

8. Program to create a student class with following attributes; Enrollment No: Name, Mark of sub1, Mark of sub2, mark of sub3, Total Marks. Total of the three marks must be calculated only when the student passes in all three subjects. The pass mark for each subject is 50. If a candidate fails in any one of the subjects his total mark must be declared as zero. Using this condition write a constructor for this class. Write separate functions for accepting and displaying student details. In the main method create an array of three student objects and display the details.

9. Program to define a class called employee with the name and date of appointment. Create ten employee objects as an array and sort them as per their date of appointment. ie, print them as per their seniority.

10. Create a package 'student. Fulltime. BCA 'in your current working directory
    a. Create a default class student in the above package with the following attributes: Name, age, sex.
    b. Have methods for storing as well as displaying

PART B: Exception Handling & GUI Programming

1. Program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values.

2. Program to handle Null Pointer Exception and use the "finally" method to display a message to the user.

3. Program which create and displays a message on the window

4. Program to draw several shapes in the created window

5. Write a java applet program to understand the properties of the font.

6. Program to create an applet and draw grid lines

7. Program which creates a frame with two buttons father and mother. When we click the father button the name of the father, his age and designation must appear. When we click mother similar details of mother also appear.

8. Create a frame which displays your personal details with respect to a button click

9. Program to move different shapes according to the arrow key pressed

10. Create a simple applet which reveals the personal information of yours.

11. Demonstrate the various mouse handling events using suitable example.

12. Program to create menu bar and pull-down menus.

Note: Student has to execute a minimum of 10 programs in each part to complete the Lab course

Course Title: Database Management System	Course code: DSC4
Total Contact Hours: 52	Course Credits: 04
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the various database concepts and the need for database systems.
- Identify and define database objects, enforce integrity constraints on a database using DBMS.
- Demonstrate a Data model and Schemas in RDBMS.
- Identify entities and relationships and draw ER diagram for a given real-world problem.
- Convert an ER diagram to a database schema and deduce it to the desired normal form.
- Formulate queries in Relational Algebra, Structured Query Language (SQL) for database manipulation.
- Explain the transaction processing and concurrency control techniques.

DSC7: Database Management System (DBMS)

Unit	Description	Hours
1	**Database Architecture:** Introduction to Database system applications. Characteristics and Purpose of database approach. People associated with Database system. Data models. Database schema. Database architecture. Data independence. Database languages, interfaces, and classification of DBMS.	10
2	**E-R Model:** Entity-Relationship modeling: E – R Model Concepts: Entity. Entity types, Entity sets, Attributes, Types of attributes, key attribute, and domain of an attribute. Relationships between the entities. Relationship types, roles and structural constraints, degree and cardinality ratio of a relationship. Weak entity types, E -R diagram.	10
3	**Relational Data Model:** Relational model concepts. Characteristics of relations. Relational model constraints: Domain constrains, key constraints. primary & foreign key constraints, integrity constraints and null values. Relational Algebra: Basic Relational Algebra operations. Set theoretical	15

	operations on relations. JOIN operations Aggregate Functions and Grouping. Nested Sub Queries-Views. Introduction to SQL (DDL,DML, DCL, TCL). Introduction to PL/SQL & programming of above operations in PL/SQL.	
4	**Data Normalization:** Anomalies in relational database design. Decomposition. Functional dependencies. Normalization. First normal form. Second normal form, Third normal form. Boyce-Codd normal form.	09
5	**Query Processing Transaction Management:** Introduction Transaction Processing. Single user & multiuser systems. Transactions: read & write operations. Need of concurrency control: The lost update problem, Dirty read problem. Types of failures. Transaction states. Desirable properties (ACID properties) of Transactions. Concurrency Control Techniques: Locks and Time stamp Ordering.	08

References:
1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7th Edition, Pearson, 2015
2. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
3. Introduction to Database System, C J Date, Pearson, 1999.
4. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010.
5. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2002

DBMS LAB

Course Title: DBMS LAB	Course code: DSC4LAB
Total Contact Hours: 52	Hours/week : 04
Formative Assessment Marks: 25	Course Credits: 02
Exam Marks: 25	Duration of Exam: 03 Hours

CO: Student would be able to create tables, execute queries and PL/SQL programs.

1. Execute DDL Commands
Consider the table:
STUDENT (regno number, name varchar2, dob date, marks number)
a) Create the above table with suitable constraints.
b) Remove the existing attribute marks from the table.
c) Change the data type of regno from number to varchar2.
d) Add a new attribute phno to the existing table.
e) Insert 5 tuples into the table.
f) Display the tuples in table

2. Execute DML Commands
Consider the table:
LIBRARY(bid number, title varchar2, author varchar2, publisher varchar2, yearof_pub number, price number)
a) create the above table.
b) Enter 5 tuples into the table.
c) Display all the tuples from the table.
d) Display different publishers from table.
e) Update price of all books with 5% GST amount.
f) Delete the details of book published by a specific Author.
e) Arrange the tuples in the alphabetical order of book title.
f) List the details of all books whose price range between 100 rs and 300 rs.

3. Execute DCL and group functions
Consider the table EMPLOYEE (EmpNo, EmpName, Dept, Salary, DOJ, Branch)
Perform the following operations:
a) Create the table
b) Insert 5 tuples into the table
c) Retrieve average salary of all employee
d) Retrieve number of employees
e) Retrieve distinct number of employee
f) Retrieve total salary of employee group by employee name and count similar names
g) Display details of employees whose salary is greater than 50000.
h) Perform Commit and Rollback operation.

4. Implement the Nested Queries.
An INVENTORY database has the following table.
ITEMS(itemcode number, name varchar2, price number)
PURCHASE(itemcode number, qty number)
a) Create the tables with the above attributes.
b) Enter 5 tuples into the tables.
c) List the items purchased.
d) List the items which are not purchased by anyone.

5. Implement Join operations in SQL
The COMPANY database consists of the tables:
EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)
DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)
Create tables, Insert 5 tuples each and perform the following
a) Give a 10 percent raise in salary for all employees working in the 'Research' Department.
b) Retrieve the name of each employee Controlled by department number 5 (use EXISTS operator).
c) Retrieve the name of each dept and number of employees working in each department which has at least 2 employees
d) Retrieve the name of employees and their department name (using NATURAL JOIN)
e) Perform EQUI join operation on the given tables.
f) Perform NON-EQUI join operation on the given tables.
g) Perform OUTER join operations on the given tables.

6. Create views for a particular table
The RAILWAY RESERVATION SYSTEM database consists of the tables:
TRAIN(TrainNo, TrainName, StartPlace, Destination)
AVAILABILITY(TrainNo, Class,StartPlace,Destination,No_of_seats)
a) Create view sleeper to display train no, start place, destination which have sleeper class and perform the following
  - insert new record
  - update destination='Manglore' where train no='RJD16'
  - delete a record which have train no='KKE55'
b) Create view details to display train no, train name, class
c) Create view total_seats to display train number, start place, use count function to no of seats , group by start place and perform the following
  - insert new record
  - update start place='Hubli' where train no='JNS8'
  - delete last row of the view
d) Rename view sleeper to class
e) Delete view details

7. Write PL/SQL procedure to compute factorial of a number using recursion
8. Given the table EMPLOYEE (EmpNo, Name, Salary, Designation, DeptID) write a cursor in PL/SQL to select the five highest paid employees from the table.
9. Given the table MOVIE(MID,MTitle, Language, Director,Year) write a function in PL/SQL to find the total number of Movies in the table.
10. Given the Table CUSTOMERS(CID,CName, Address) write a PL/SQL program which asks for customer ID, if the user enters invalid ID then the exception **invalid_id** is raised.

Model Curriculum for BCA

Sem	Core Courses	Hour / Week		DS Elective Courses	Hours/ Week
		Theory	Lab		
III	Database Management Systems	3			
	C# and DOT NET Framework	3			
	Computer Communication and Networks	3			
	LAB: DBMS		4		
	LAB: C# and DOT NET Framework		4		
IV	Python Programming	3			
	Computer Multimedia and Animation	3			
	Operating Systems Concepts	3			
	LAB: Multimedia and Animation		4		
	LAB: Python programming		4		

Course Title: Database Management System	Course code: CAC07
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the various database concepts and the need for database systems.
- Identify and define database objects, enforce integrity constraints on a database using DBMS.
- Demonstrate a Data model and Schemas in RDBMS.
- Identify entities and relationships and draw ER diagram for a given real-world problem.
- Convert an ER diagram to a database schema and deduce it to the desired normal form.
- Formulate queries in Relational Algebra, Structured Query Language (SQL) for database manipulation.
- Explain the transaction processing and concurrency control techniques.

DSC7: Database Management System (DBMS)

Unit	Description	Hours
1	**Database Architecture:** Introduction to Database system applications. Characteristics and Purpose of database approach. People associated with Database system. Data models. Database schema. Database architecture. Data independence. Database languages, interfaces, and classification of DBMS.	08
2	**E-R Model:** Entity-Relationship modeling: E – R Model Concepts: Entity. Entity types, Entity sets, Attributes, Types of attributes, key attribute, and domain of an attribute. Relationships between the entities. Relationship types, roles and structural constraints, degree and cardinality ratio of a relationship. Weak entity types, E -R diagram.	08
3	**Relational Data Model:** Relational model concepts. Characteristics of relations. Relational model constraints: Domain constrains, key constraints, primary & foreign key constraints, integrity constraints and null values.	12

	Relational Algebra: Basic Relational Algebra operations. Set theoretical operations on relations. JOIN operations Aggregate Functions and Grouping. Nested Sub Queries-Views. Introduction to SQL (DDL,DML, DCL, TCL) . Introduction to PL/SQL & programming of above operations in PL/SQL	
4	**Data Normalization:** Anomalies in relational database design. Decomposition. Functional dependencies. Normalization. First normal form, Second normal form, Third normal form. Boyce-Codd normal form.	06
5	**Query Processing Transaction Management:** Introduction Transaction Processing. Single user & multiuser systems. Transactions: read & write operations. Need of concurrency control: The lost update problem, Dirty read problem. Types of failures. Transaction states. Desirable properties (ACID properties) of Transactions. Concurrency Control Techniques: Locks and Time stamp Ordering.	08

References:
1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7th Edition, Pearson, 2015
2. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
3. Introduction to Database System, C J Date, Pearson, 1999.
4. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010.
5. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2002

Course Title: DBMS LAB	Course code: CAC07P
Total Contact Hours: 52	Hours/week : 04
Formative Assessment Marks: 25	Course Credits: 02
Exam Marks: 25	Duration of Exam: 03 Hours

CO: Student would be able to create tables, execute queries and PL/SQL programs.

1. Execute DDL Commands
Consider the table:
STUDENT (regno number, name varchar2, dob date, marks number)
a) Create the above table with suitable constraints.
b) Remove the existing attribute marks from the table.
c) Change the data type of regno from number to varchar2.
d) Add a new attribute phno to the existing table.
e) Insert 5 tuples into the table.
f) Display the tuples in table

2. Execute DML Commands
Consider the table:
LIBRARY(bid number, title varchar2, author varchar2, publisher varchar2, yearof_pub number, price number)
a) create the above table.
b) Enter 5 tuples into the table.
c) Display all the tuples from the table.
d) Display different publishers from table.
e) Update price of all books with 5% GST amount.
f) Delete the details of book published by a specific Author.
e) Arrange the tuples in the alphabetical order of book title.
f) List the details of all books whose price range between 100 rs and 300 rs.

3. Execute DCL and group functions
Consider the table EMPLOYEE (EmpNo, EmpName, Dept, Salary, DOJ, Branch)
Perform the following operations:
a) Create the table
b) Insert 5 tuples into the table
c) Retrieve average salary of all employee
d) Retrieve number of employees
e) Retrieve distinct number of employee
f) Retrieve total salary of employee group by employee name and count similar names
g) Display details of employees whose salary is greater than 50000.
h) Perform Commit and Rollback operation.

4. Implement the Nested Queries.

An INVENTORY database has the following table.

ITEMS(itemcode number, name varchar2, price number)

PURCHASE(itemcode number, qty number)

a) Create the tables with the above attributes.

b) Enter 5 tuples into the tables.

c) List the items purchased.

d) List the items which are not purchased by anyone.

5. Implement Join operations in SQL

The COMPANY database consists of the tables:

EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)

DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)

Create tables, Insert 5 tuples each and perform the following

a) Give a 10 percent raise in salary for all employees working in the 'Research' Department.

b) Retrieve the name of each employee Controlled by department number 5 (use EXISTS operator).

c) Retrieve the name of each dept and number of employees working in each department which has at least 2 employees

d) Retrieve the name of employees and their department name (using NATURAL JOIN)

e) Perform EQUI join operation on the given tables.

f) Perform NON-EQUI join operation on the given tables.

g) Perform OUTER join operations on the given tables.

6. Create views for a particular table

The RAILWAY RESERVATION SYSTEM database consists of the tables:

TRAIN(TrainNo, TrainName, StartPlace, Destination)

AVAILABILITY(TrainNo, Class,StartPlace,Destination,No_of_seats)

a) Create view sleeper to display train no, start place, destination which have sleeper class and perform the following

- insert new record
- update destination='Manglore' where train no='RJD16'
- delete a record which have train no='KKE55'

b) Create view details to display train no, train name, class

c) Create view total_seats to display train number, start place, use count function to no of seats , group by start place and perform the following

- insert new record
- update start place='Hubli' where train no='JNS8'
- delete last row of the view

d) Rename view sleeper to class

e) Delete view details

7. Write PL/SQL procedure to compute factorial of a number using recursion

8. Given the table EMPLOYEE (EmpNo, Name, Salary, Designation, DeptID) write a cursor in PL/SQL to select the five highest paid employees from the table.

9. Given the table MOVIE(MID,MTitle, Language,Director,Year) write a function in PL/SQL to find the total number of Movies in the table.

10. Given the Table CUSTOMERS(CID,CName, Address) write a PL/SQL program which asks for customer ID, if the user enters invalid ID then the exception **invalid_id** is raised.

Course Title: C# and Dot Net Framework	Course code: CAC08
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Describe Object Oriented Programming concepts like Inheritance and Polymorphism in C# programming language.
- Interpret and Develop Interfaces for real-time applications.
- Build custom collections and generics in C#.

DSC8: C# and Dot Net Framework

Unit	Description	Hours
1	**Introduction to .Net Technologies:** Introduction to Web Technologies. HTML Basics, Scripts. Sample Programs. Advantages and Disadvantages of Client-side and Server-side Scripts. Overview of Client-side Technologies and Server-side Technologies.	08
2	**Introduction to C#:** Overview of C#, Literals, Variables, Data Types. Operators, Expressions, Control Structures-Methods, Arrays, Strings. Structures, Enumerations. **OOPS with C#:** Classes, Objects, Inheritance, Polymorphism, Interfaces, Operator Overloading Delegates, Events, Errors and Exceptions.	08
3	**Introduction to VB.NET**: Introduction VB.NET -IDE – Creating a shortcut to start VB.NET. Maneuverings the Toolbar Auto-hide, Docking and Undocking, Placing and Resizing the Windows, Forms, Properties Window and Solution Explorer. Writing and Event Procedure. Execution Basic Keywords. Data Types. VB.NET statements. Conditional statements: If Else. Select Case, Switch and Choose Loops: Do, For Next, For Each Next, While loop. Arrays.	08
4	**Application Development on .NET:** C#.NET: Building Windows Applications, VB.NET: Windows Forms. Working with Controls, Timer. Picture-box, Group-box, Combo-box, Horizontal and Vertical Scrollbar. Numeric-up-down, Track-bar, and Progress-bar. Subroutines and Functions in VB.NET. Database applications	10
5	**ADO .NET Connectivity:** Introduction to ADO.NET, ADO vs ADO.NET. Architecture: Data reader, Data adopter, Accessing Data with ADO.NET. Programming Web Applications with Web Forms.	08

References:
1. "Programming in C#", E. Balagurusamy, 4th Edition, Tata McGraw-Hill, 2017.
2. "Visual Basic.NET", Shirish Chavan, 3rd Edition, Pearson Education, 2009.
3. "ASP.NET and VB.NET Web Programming", Matt J. Crouch, Edition 2012.
4. "Computing with C# and the .NET Framework", Arthur Gittleman, 2nd Edition, Jones & Bartlett Publishers, 2011

Course Title: C# and Dot Net Framework Lab	Course code: CAC08P
Total Contact Hours: 52	Hours/week : 04
Formative Assessment Marks: 25	Course Credits: 02
Exam Marks: 25	Duration of Exam: 03 Hours

1. Develop a C# .NET console application to demonstrate the conditional statements.
2. Develop a C# .NET console application to demonstrate the control statements.
3. Develop an application in C#.NET that demonstrates the windows controls
4. Demonstrate Multithreaded Programming in C#.NET
5. Demonstrate subroutines and functions in C#.NET
6. Develop an application for deploying various built-in functions in VB.NET
7. Demonstrate MDI application using VB.NET
8. Construct a console application to demonstrate the OOP Concepts
9. Develop a web application in VB.NET for dynamic Login Processing
10. Develop a Windows application with database connectivity for core-banking transactions

Course Title: Computer Communication and Networks	Course code: CAC09
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Explain the transmission technique of digital data between two or more computers and a computer network that allows computers to exchange data.
- Apply the basics of data communication and various types of computer networks in real world applications.
- Compare the different layers of protocols.
- Compare the key networking protocols and their hierarchical relationship in the conceptual model like TCP/IP and OSI.

Computer Communication and Networks

Unit	Description	Hours
1	**Introduction:** Computer Networks and its applications, Network structure, network architecture, Topologies, LAN, WAN, MAN, The OSI reference model, The TCP/IP reference model.	08
2	**The Physical Layer:** Transmission Media – Twisted pair, coaxial cable, optical fiber, radio transmission, microwaves and infrared transmission. Switching – message switching, Multiplexing.	07
3	**The Data Link Layer:** Data Link Layer design issues, Error detection – Single parity checking, Checksum, polynomial codes – CRC, Error correction- Hamming code, Elementary data link protocols, sliding window protocols	08
4	**The Network Layer:** Network layer design issues, Routing algorithms – Flooding, Distance vector routing, Hierarchical routing, Link state routing. Congestion, control algorithms – Leaky bucket, token bucket algorithm, admission control, Hop by Hop choke packets.	09
5	**The Transport Layer and Application Layer:** Elements of Transport service, Elements of Transport, protocols, Internet transport protocols (TCP & UDP), DNS, Electronic Mailing, and World Wide Web.	10

References:
1. Computer Networks, Andrew S. Tanenbaum, 5th Edition, Pearson Education, 2010.
2. Data Communication & Networking, Behrouza A Forouzan, 3rd Edition, Tata McGraw

Hill,2001.

3. Data and Computer Communications, William Stallings, 10th Edition, Pearson Education, 2017.

4. Data Communication and Computer Networks, Brijendra Singh, 3rd Edition, PHI, 2012.

5. Data Communication & Network, Dr. Prasad, Wiley Dreamtech.

6. http://highered.mheducation.com/sites/0072967757/index.htmls

Course Title: Python Programming	Course code: CAC10
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and creation of functions.
- Identify the methods to create and manipulate lists, tuples and dictionaries.
- Discover the commonly used operations involving file handling.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Develop the emerging applications of relevant fields using Python.

DSC10: Python Programming

Unit	Description	Hours
1	**Introduction** to Features and Applications of Python; Python Versions; Installation of Python; Python Command Line mode and Python IDEs; Simple Python Program. **Python Basics:** Identifiers; Keywords; Statements and Expressions; Variables; Operators; Precedence and Association; Data Types; Indentation; Comments; Built-in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples. Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range () and exit () functions.	08
2	**Exception Handling:** Types of Errors; Exceptions; Exception Handling using try, except and finally. **Python Functions:** Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement; Default Parameters; Command line Arguments; Key Word Arguments; Recursive Functions; Scope and Lifetime of Variables in Functions. **Strings:** Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods.	08
3	**Lists:** Creating Lists; Operations on Lists; Built-in Functions on Lists; Implementation of Stacks and Queues using Lists; Nested Lists. Dictionaries: Creating Dictionaries; Operations on Dictionaries; Built-in Functions on Dictionaries; Dictionary Methods; Populating and Traversing Dictionaries. **Tuples and Sets:** Creating Tuples; Operations on Tuples; Built-in Functions	08

	on Tuples; Tuple Methods; Creating Sets; Operations on Sets; Built-in Functions on Sets; Set Methods.	
4	**File Handling:** File Types; Operations on Files– Create, Open, Read, Write, Close Files; File Names and Paths; Format Operator. **Object Oriented Programming:** Classes and Objects; Creating Classes and Objects; Constructor Method; Classes with Multiple Objects; Objects as Arguments; Objects as Return Values; Inheritance- Single and Multiple Inheritance, Multilevel and Multipath Inheritance; Encapsulation- Definition, Private Instance Variables; Polymorphism- Definition, Operator Overloading.	08
5	**GU Interface:** The tkinter Module; Window and Widgets; Layout Management- pack, grid and place. **Python SQLite:** The SQLite3 module; SQLite Methods- connect, cursor, execute, close; Connect to Database; Create Table; Operations on Tables- Insert, Select, Update. Delete and Drop Records. **Data Analysis:** NumPy- Introduction to NumPy, Array Creation using NumPy, Operations on Arrays; Pandas- Introduction to Pandas, Series and DataFrames, Creating DataFrames from Excel Sheet and .csv file, Dictionary and Tuples. Operations on DataFrames. **Data Visualisation:** Introduction to Data Visualisation; Matplotlib Library; Different Types of Charts using Pyplot- Line chart, Bar chart and Histogram and Pie chart.	10

References:

1. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2n Edition, Green Tea Press. Freely available online @ https://www.greenteapress.com/thinkpython/thinkCSpy.pdf, 2015.
2. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
3. Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language, Fabio Nelli, Apress®, 2015
4. Advance Core Python Programming, MeenuKohli, BPB Publications, 2021.
5. Core PYTHON Applications Programming, Wesley J. Chun, 3rd Edition, Prentice Hall 2012.
6. Automate the Boring Stuff, Al Sweigart, No Starch Press, Inc, 2015.
7. Data Structures and Program Design Using Python, D Malhotra et al., Mercury Learning and Information LLC, 2021.
8. http://www.ibiblio.org/g2swap/byteofpython/read/
9. https://docs.python.org/3/tutorial/index.html

Programs for Practical Component:

Course Title: Python Programming Lab	Course code: CAC10P
Total Contact Hours: 52	Hours/week : 04
Formative Assessment Marks: 25	Course Credits: 02
Exam Marks: 25	Duration of Exam: 03 Hours

Part-A
1. Check if a number belongs to the Fibonacci Sequence
2. Solve Quadratic Equations
3. Find the sum of n natural numbers
4. Display Multiplication Tables
5. Check if a given number is a Prime Number or not
6. Implement a sequential search
7. Create a calculator program
8. Explore string functions

9. Implement Selection Sort
10. Implement Stack
11. Read and write into a file

Part-B
1. Demonstrate usage of basic regular expression
2. Demonstrate use of advanced regular expressions for data validation.
3. Demonstrate use of List
4. Demonstrate use of Dictionaries
5. Create SQLite Database and Perform Operations on Tables
6. Create a GUI using Tkinter module
7. Demonstrate Exceptions in Python
8. Drawing Line chart and Bar chart using Matplotlib
9. Drawing Histogram and Pie chart using Matplotlib
10. Create Array using NumPy and Perform Operations on Array
11. Create DataFramefrom Excel sheet using Pandas and Perform Operations on DataFrames

Note: A minimum of 10 Programs should be done in each Part.

Evaluation Scheme for Lab Examination:

Assessment Criteria	Marks
Writing 2 Programs	10
Execution of 1 Program	10
Viva and Record	05
Total	25

Course Title: Computer Multimedia & Animation	Course code: CAC11
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Write a well-designed, interactive Web site with respect to current standards and practices.
- Demonstrate in-depth knowledge of an industry-standard multimedia development tool and its associated scripting language.
- Determine the appropriate use of interactive versus standalone Web applications.

DSC11: Computer Multimedia & Animation

Unit	Description	Hours
1	Web Design: Origins and evolution of HTML, Basic syntax, Basic text markup, Images, Lists, Tables, Forms, Frame, Overview and features of HTML5. CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The <span> and <div> tags; Overview and features of CSS3. JavaScript: Object orientation and JavaScript; General syntactic characteristics; Primitives, operations, and expressions; Screen output and keyboard input.	12
2	Animation: What is an Animation? The Start and End States, Interpolation, Animations in HTML. All About CSS Animations, Creating a Simple Animation, Detailed Look at the CSS Animation Property, Keyframes, Declaring Multiple Animations, Wrap-up. All About CSS Transitions, Adding a Transition, Looking at Transitions in Detail, The Longhand Properties, Longhand Properties vs. Shorthand Properties, Working with Multiple Transitions.	09
3	HTML5 – SVG: Viewing SVG Files, Embedding SVG in HTML5, HTML5 − SVG Circle, HTML5 − SVG Rectangle, HTML5 − SVG Line, HTML5 − SVG Ellipse, HTML5 − SVG Polygon, HTML5 − SVG Polyline, HTML5 − SVG Gradients, HTML5 − SVG Star.	08
4	HTML5 – CANVAS: The Rendering Context, Browser Support, HTML5 Canvas Examples, Canvas - Drawing Rectangles, Canvas - Drawing Paths, Canvas - Drawing Lines, Canvas - Drawing Bezier Curves, Canvas - Drawing Quadratic Curves, Canvas - Using Images, Canvas - Create Gradients,	07
5	HTML5 - Styles and Colors, Canvas - Text and Fonts, Canvas - Pattern and Shadow, Canvas - Save and Restore States, Canvas - Translation, Canvas - Rotation, Canvas - Scaling, Canvas - Transforms, HTML5 Canvas - Composition, Canvas – Animations.	06

References:
1. The Complete Reference HTML and CSS, 5th Edition, Thomas A Powell, 2017.
2. Animation in HTML, CSS, and JavaScript, Kirupa Chinnathambi, 1st Edition, Createspace Independent Pub, 2013.
3. https://www.w3.org/Style/CSS/current-work#CSS3
4. http://bedford-computing.co.uk/learning/cascading-style-sheets-css/

Course Title: Multimedia Animation Lab	Course code: CAC11P
Total Contact Hours: 52	Course Credits: 02
Formative Assessment Marks: 25	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 25	

List of Lab programs

Part-A

1. Write an HTML program to display Multi-media data (text, images, audios, videos, gifs, etc) on a webpage.

2. Write an HTML program to create and display navigations menus using list tags and anchor tag

3. Write a HTML code to create your class time table.

4. Write an HTML code to create a frameset having header, navigation and content sections.

5. Write an HTML program to create student Registrations form on submitting the form check whether fields are empty or not using JavaScript. If any fields are empty display an error message

6. Write an HTML program to create bio-data(CV or Resume) and to change the following CSS properties
   - Font
   - Text
   - Background

7. Write an HTML program to create div and apply the following CSS properties on created div
   - Margin
   - Padding
   - Border
   - Box shadow

8. Write an HTML program to create a box and using CSS transform and transition properties move the box to the center of the web page on loading web-page

9. Write an HTML program to create a circle and create an animation of bouncing of the circle for 10 sec

10. Write an HTML program to create page loading animations

Part-B

1. Write an HTML program to draw line, polyline and rectangle and fill rectangle with red color using SVG tag.

2. Write an HTML program to draw star and multiple circle and with different color using SVG tag

3. Write an HTML program to create logo with linear gradient properties using SVG tag.

4. Write an HTML program to draw Square and Rectangle using canvas tag and JavaScript

5. Write an HTML program to draw bezier curve using canvas tag and JavaScript

6. Write an HTML Program to import an external image into a canvas and then to draw on that image

7. Write an HTML program to draw a rectangle box using canvas and to change background color to red, scale of the rectangle to 2 on move-over (hover) properties.

8. Write an html program to draw a circle using canvas and to apply the rotations animations on loading the page

Evaluation Scheme for Lab Examination:

Assessment Criteria	Marks
Writing 2 Programs	10
Execution of 1 Program	10
Viva and Record	05
Total	25

Course Title: Operating System Concepts	Course code: CAC12
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Explain the fundamentals of the operating system.
- Comprehend multithreaded programming, process management, process synchronization, memory management and storage management.
- Compare the performance of Scheduling Algorithms
- Identify the features of I/O and File handling methods.

DSC12: Operating System Concepts

Unit	Description	Hours
1	**Introduction to Operating System:** Definition, History and Examples of Operating System; Computer System organization; Types of Operating Systems; Functions of Operating System; Systems Calls; Operating System Structure. **Process Management:** Process Concept- Process Definition, Process State, Process Control Block, Threads; Process scheduling- Multiprogramming, Scheduling Queues, CPU Scheduling, Context Switch; Operations on Processes- Creation and Termination of Processes; Inter process communication (IPC)- Definition and Need for Inter process Communication; IPC Implementation Methods- Shared Memory and Message Passing;	08
2	**Multithreaded Programming**: Introduction to Threads; Types of Threads; Multithreading- Definition, Advantages; Multithreading Models; Thread Libraries; Threading Issues. **CPU Scheduling**: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling; Real-Time CPU Scheduling.	10
3	**Process Synchronization:** Introduction; Race Condition; Critical Section Problem and Peterson's Solution; Synchronization Hardware, Semaphores; Classic Problems of Synchronization- Readers and Writers Problem, Dining Philosophers Problem; Monitors. **Deadlocks:** System Model; Deadlocks Characterization; Methods for Handling Deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection; and Recovery from Deadlock.	10

4	**Memory Management:** Logical and Physical Address Space; Swapping; Contiguous Allocation; Paging; Segmentation; Segmentation with Paging. **Virtual Memory:** Introduction to Virtual Memory; Demand Paging; Page Replacement; Page Replacement Algorithms; Allocation of frames. Thrashing.	08
5	**File System:** File Concepts- Attributes, Operations and Types of Files; File System; File Access methods; Directory Structure; Protection; File System Implementation- File System Structure, Allocation Methods, Free Space Management	06

References:

1. Operating System Concepts, Silberschatz' et al., 10th Edition, Wiley, 2018.
2. Operating System Concepts - Engineering Handbook, Ghosh PK, 2019.
3. Understanding Operating Systems, McHoes A et al., 7th Edition, Cengage Learning, 2014.
4. Operating Systems - Internals and Design Principles, William Stallings, 9th Edition, Pearson.
5. Operating Systems – A Concept Based Approach, Dhamdhere, 3rd Edition, McGraw Hill Education India.
6. Modern Operating Systems, Andrew S Tanenbaum, 4th Edition, Pearson.

Course Title: **Artificial Intelligence**	Course Credits: 2
Total Contact Hours: 13 hours of theory and 26 hours of practical	Duration of ESA: 01 Hour
Formative Assessment Marks: **20 marks**	Summative Assessment Marks: 30 marks

Course Outcomes (COs):

At the end of the course, students will be able to:
- Appraise the theory of Artificial intelligence and list the significance of AI.
- Discuss the various components that are involved in solving an AI problem.
- Illustrate the working of AI Algorithms in the given contrast.
- Analyze the various knowledge representation schemes, Reasoning and Learning techniques of AI.
- Apply the AI concepts to build an expert system to solve the real-world problems.

Course Content (Artificial Intelligence)

	Details of topic	Duration

Course – 1 - Azure AI Fundamentals (AI-900)	AI-900 pathway consists of 5 courses and 2 reading material:
	  i. Introduction to AI on Azure
	  ii. Use visual tools to create machine learning models with Azure Machine Learning
	  iii. Explore computer vision in Microsoft Azure
	  iv. Explore natural language processing
	  v. Explore conversational AI
	  vi. Tune Model Hyperparameters - Azure Machine Learning (Reading)
	  vii. Neural Network Regression: Module Reference - Azure Machine Learning (Reading
		05 hours

Practical	1. Prepare the data 2. Model the data 3. Visualize the data 4. Analyse the data 5. Deploy and maintain deliverables	13 hours

Course – 2 - Data Analyst Associate (DA-100)	DA-100 pathway consists of 5 courses and 2 reading material: 1. Get started with Microsoft data analytics 2. Prepare data for analysis 3. Model data in Power BI 4. Visualize data in Power BI 5. Data analysis in Power BI 6. Manage workspaces and datasets in Power BI 7. Key Influencers Visualizations Tutorial - Power BI 8. Smart Narratives Tutorial - Power BI	Microsoft Docs	08 hours
Practical	1. Describe Artificial Intelligence workloads and considerations 2. Describe fundamental principles of machine learning on Azure 3. Describe features of computer vision workloads on Azure 4. Describe features of Natural Language Processing (NLP) workloads on Azure	13 hours

References to learning resources:
1. The learning resources made available for the course titled "Azure AI Fundamentals (AI-900) and Data Analyst Associate (DA-100)." on Future Skills Prime Platform of NASSCOM.

Pedagogy
Flipped classroom pedagogy is recommended for the delivery of this course.
For every class:
1. All the faculty who takes this class should go for a Faculty Development Program on these before starting the session.
2. Faculty needs to introduce this course to the students then students need to start learning from Future Skills PRIME platform.
3. Faculty also needs to explain the course outcomes and needs of the course and why it is needed for the students.
4. Then students need to start learning online after registering on the platform.
5. Classroom activities are designed around the topic of the session towards developing better understanding, clearing doubts and discussions of high order thinking skills like application, analysis, evaluation, and design.
6. Every theory class ends with announcement of exercise for practical activity of the week.

Exercises:

Practical Exercises	Weightage in marks
After each chapter students' needs to complete exercises based on the learning in Azure environment.	No Weightage (But students need to complete it to move to next chapter) .

Assessment:

Formative Assessment	
Assessment Occasion	**Weightage in Marks**
1. Summative Assessment: After completion of both the courses, the student can optionally give Assessment for each of the courses on Future Skills Prime platform. Students will have two attempts and those who score at least 50% marks per course will get certificate from NASSCOM-MeitY.	This assessment may be given 50% weight in computing the final grade of the students.

Course Title: **Open Source Tools**	Course Credits: 2 (1L+0T+2P)
Semester: III	Duration of SEE: 01 Hour
Total Contact Hours: 13 hours of theory and 26-28 hours of practicals	SEE: 30 Marks IA: 20 Marks

Course Outcomes (COs):

- Recognize the benefits and features of Open Source Technology and to interpret, contrast and compare open source products among themselves
- Use appropriate open source tools based on the nature of the problem
- Write code and compile different open-source software.

Course Content (Open Source Tools)

Module	Details of topic	Duration
Module 1: Open Source Softwares	i. Introduction to Open sources, Need of Open Sources, Open Source –Principles, Standard Requirements, Advantages of Open Sources – ii. Free Software – FOSS iii. Licenses – GPL, LGPL, Copyrights, Patents, Contracts & Licenses and Related Issues iv. Application of Open Sources. Open Source Operating Systems : FEDORA, UBUNTU	05 hours
Module 2: Programming Tools And Techniques	i. Usage of design Tools like Argo UML or equivalent ii. Version Control Systems like Git or equivalent iii. Bug Tracking Systems (Trac, BugZilla) iv. BootStrap	04 hours
Module 3: Case Studies	**i.** Apache **ii.** Berkeley Software Distribution **iii.** Mozilla (Firefox) **iv.** Wikipedia **v.** Joomla **vi.** GNU Compiler Collection **vii.** Libre Office	04 hours

Text Book:

1. KailashVadera, Bhavyesh Gandhi, "Open Source Technology", Laxmi Publications Pvt. Ltd 2012, 1st Edition.

Reference Book:

1. Fadi P. Deek and James A. M. McHugh, "Open Source: Technology and Policy", Cambridge Universities Press 2007.

Course Title: **Python Programming Concepts**	Course Credits: 3 (3L+0T+0P)
Semester: III/IV	Duration of SEE: 03 Hour
Total Contact Hours: 42	SEE: 60 Marks IA: 40 Marks

Course Outcomes (COs):

- Explain the fundamentals of Computers.
- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and the creation of functions.
- Identify the methods to create and store strings.

Unit I   Fundamentals of Computers   10 Hrs

Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and History of Computers, Types of Computers, Basic Organization of a Digital Computer; Number Systems – different types, conversion from one number system to another; Types of Software – System Software and Utility Software; Computer Languages - Machine Level, Assembly Level & High Level Languages, Translator Programs
– Assembler, Interpreter and Compiler; Planning a Computer Program - Algorithm, Flowchart and Pseudo code with Examples.

Unit II   Python Basics   10 Hrs

Introduction to Features and Applications of Python; Python Versions; Installation of Python; Python Command Line mode and Python IDEs; Simple Python Program. Identifiers; Keywords; Statements and Expressions; Variables; Operators; Data Types; Indentation; Comments; Built-in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples; Illustrative programs.

Unit III   08 Hrs

Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range() and exit () functions; Illustrative programs.

Unit IV   08 Hrs

Python Functions: Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement; Default Parameters; Command line Arguments; Key Word Arguments; Illustrative programs.

Unit V   **06 Hrs**

Strings: Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods; Illustrative programs.

References

1. Computer Fundamentals (BPB), P. K. Sinha & Priti Sinha
2. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2nd Edition, Green Tea Press. Freely available online 2015. @https://www.greenteapress.com/thinkpython/thinkCSpy.pdf
3. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
4. http://www.ibiblio.org/g2swap/byteofpython/read/
5. http://scipy-lectures.org/intro/language/python_language.html
6. https://docs.python.org/3/tutorial/index.html

Open Elective

ELECTRONIC COMMERCE

Course Title: **E-Commerce**	Course Credits: 3 (3L+0T+0P)
Semester: III/IV	Duration of SEE: 03 Hour
Total Contact Hours: 42	SEE: 60 Marks IA: 40 Marks

Course Outcomes (COs):
- Compare how internet and other information technologies support business processes.
- Demonstrate an overall perspective of the importance of application of internet technologies in business administration
- Explain the basic business management concepts.
- Demonstrate the basic technical concepts relating to E-Commerce.
- Identify the security issues, threats and challenges of E-Commerce.

UNIT I   Introduction to E-Commerce and Technology Infrastructure   09 Hrs

Working of Web - HTML Markup for Structure - Creating simple page - Marking up text - Adding Links - Adding Images - Table Markup - Forms - HTML5

UNIT II   Building an E-Commerce Website, Mobile Site and Apps   10 Hrs

Systematic approach to build an E-Commerce: Planning, System Analysis, System Design, Building the system, Testing the system, Implementation and Maintenance, Optimize Web Performance – Choosing hardware and software – Other E-Commerce Site tools – Developing a Mobile Website and Mobile App

UNIT III   E-Commerce Security and Payment Systems   09 Hrs

E-Commerce Security Environment – Security threats in E-Commerce – Technology Solutions: Encryption, Securing Channels of Communication, Protecting Networks, Protecting Servers and Clients – Management Policies, Business Procedure and Public Laws - Payment Systems

UNIT IV   Business Concepts in E-Commerce   09 Hrs

Digital Commerce Marketing and Advertising strategies and tools – Internet Marketing Technologies – Social Marketing – Mobile Marketing – Location based Marketing – Ethical, Social, Political Issues in E-Commerce

UNIT V   Project Case Study   05 Hrs

Case Study: Identify Key components, strategy, B2B, B2C Models of E-commerce Business model of any e-commerce website - Mini Project : Develop E-Commerce project in any one of Platforms like Woo-Commerce, Magento or Opencart

Text Book:
1. Kenneth C. Laudon, Carol Guercio Traver - E-Commerce, Pearson, 10th Edition, 2016

References:
1. http://docs.opencart.com/
2. http://devdocs.magento.com/
3. http://doc.prestashop.com/display/PS15/Developer+tutorials
4. Robbert Ravensbergen, ―Building E-Commerce Solutions with Woo Commerce‖, PACKT, 2nd Edition