

**Department of Computer Science
University of Delhi**

Skill Enhancement Courses (SECs)

Under UGCF 2022

DRAFT

**Approved in
UG Committee meeting held on May17, 2022
Faculty of Mathematical Sciences meeting held on May 25, 2022**

1. [List of SECs](#)

Table of SECs

Semester	SEC -No.	Title	L	T	P	Total credits	Pre-requisites
I/III/V/V II	SEC01A	Front End Web Design and Development	1	0	1	2	Nil
	SEC01B	Office Automation Tools	1	0	1	2	Nil
	SEC01C	App Development for Android	1	0	1	2	Basic knowledge of programming
	SEC1D	Basics of Programming in Python	1	0	1	2	For non-CS students
II/IV/VI/ VIII	SEC02A	Backend Web Development	1	0	1	2	Knowledge of HTML, CSS
	SEC02B	Data Analysis using Spreadsheet	1	0	1	2	Nil
	SEC02C	App Development for iOS	1	0	1	2	Basic knowledge of programming
	SEC2D	Introduction to Object Oriented Programming using Python	1	0	1	2	Basic knowledge of Python

Note:

1. For a prerequisite of Python/HTML with CSS, a course at plus 2 level is acceptable.
2. Batch size for Practicals will be (8-10) and Tutorials will be (12-15).
3. More SECs may be added in due course of time.

SEC01A: Front End Web Design and Development**Course Objective**

The course aims at introducing the basic concepts and techniques of client side web programming. The student shall be able to develop simple, interactive and stylish websites using HTML, CSS and Javascript.

Course Learning Outcomes

On successful completion of this course, the student will be able to:

1. Build websites using the elements of HTML.
2. Build interactive and stylish websites using the client side programming techniques with CSS and Javascript.
3. Learn to validate client-side data.
4. Define the structure and contents of the website using different features of CSS.

Syllabus

Unit 1 Introduction: Introduction to internet and web design. Basic concepts of web architecture.

Unit 2 HTML: Introduction to hypertext mark-up language (html), creating web pages, lists, hyperlinks, tables, web forms, inserting images, frames.

Unit 3 Cascading style sheet (CSS): Concept of CSS, creating style sheet, Importing style sheets, CSS properties, CSS styling (background, text format, controlling fonts), CSS rules, Style Types, CSS Selectors, CSS cascade, working with block elements and objects, working with lists and tables, CSS id and class, box model (introduction, border properties, padding properties,

margin properties).

Unit 4 Basics of Javascript: Document object model, data types and variables, functions, methods and events, controlling program flow, built-in objects and operators, validations.

References

1. Nixon, R., *Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5*, O'Reilly, 2018.
2. Powell, T.A. *HTML & CSS: The Complete Reference*, 5th edition, Tata McGrawHill, 2017.
3. Duckett, J., *JavaScript and JQuery: Interactive Front-End Web Development*, Wiley, 2014.

Additional References

- (i) Boehm, A., & Ruvalcaba, Z., *Murach's HTML5 and CCS*, 4th edition, Mike Murach & Associates, 2018.
- (ii) Ivan Bayross, *Web Enabled Commercial Application Development Using Html, Dhtml, Javascript, Perl CGI*, BPB Publications, 2010.

Suggested Practical List

HTML

1. Create an HTML document with following formatting – Bold, Italics, Underline, Colors, Headings, Title, Font and Font Width, Background, Paragraph, Line Brakes, Horizontal Line, Blinking text as well as marquee text.
2. Create an HTML document with Ordered and Unordered lists, Inserting Images, Internal and External linking
3. Create an HTML document for displaying the current semester's time table.
4. Create a website with horizontal and vertical frames. Top horizontal frame needs to show your college's name and logo. Bottom horizontal frame is to split into two vertical frames. The left frame has hyperlinks to pages related to faculty, courses, student activities, etc. The right frame shows the corresponding webpage based on the link

clicked on the left frame.

5. Create a student registration form using HTML which has the following controls and make interactive content presentation using CSS.:
 - I. Text Box
 - II. Dropdown box
 - III. Option/radio buttons
 - IV. Check boxes
 - V. Reset and Submit button
6. Create a webpage for your department with a drop down navigation menu for faculty, courses, activities, etc.. Implement the webpage using styles, rules, selectors etc. learnt in CSS
7. Write event driven programs in JavaScript for the following:
 1. Enter a number and on click of a button print its multiplication table.
 2. Print the largest of three numbers entered by the user.
 3. Find the factorial of a number entered by the user.
 4. Enter a list of positive numbers using the prompt terminated by a zero. Find the sum and average of these numbers.
8. Create a student registration form using text, radio button, check box, drop down box, text field and all other required HTML elements. Customize the CSS and javascript to input and validate all data. Create functions to perform validation of each element, example:
 - a. Roll number is a 7-digit numeric value
 - b. Name should be an alphabetical value(String)
 - c. Non-empty and valid fields like DOB

SEC01B: Office Automation Tools

Course Objective

The course introduces the students to document processing, presentation software and data handling. The basic features and skills of creating, editing, inserting tables, graphics as well as presentation tools along with spreadsheet data handling are covered.

Course Learning Outcomes

On successful completion of this course, a student will be able to:

1. create and refine documents using text formatting, tables and graphics.
2. use mail merge.
3. create macros and templates in documents.
4. protect documents.
5. create presentations containing transitions and animations. learn advanced presentation features like custom slide show, call outs and action buttons.
6. use referencing and functions for data handling.

Unit 1 Document Processing Basics: Creating, opening and saving a document, text formatting, header and footer, creating and editing of tables, importing graphics, insert picture, using word processor's drawing features, text in drawing.

Unit 2 Advanced Features: Creating macros, watermarks, templates, reviewing documents, comparing and combining documents, protection of documents-using passwords. Mail merge concept, main document, data sources, merging data source and main document.

Unit 3 Presentation Tools: Creating presentations, using blank presentation option, using design template option, adding slides, deleting a slide, importing images from the outside world, deleting a slide, numbering a slide, saving presentation transition and animations, adding notes to slides, customize slideshow.

Unit 4 Data Handling using spreadsheets: Relative, absolute and mixed referencing, mathematical and statistical functions, nested functions

References

1. Mali, L., *Libre office 5.1 Impress, Draw*. Base book- Vol 2, Notion Press, 2017..
2. Libreoffice Documentation Team, *Getting Started with LibreOffice 6.0*. Friends of OpenDocument, Inc., 2018.
3. Libreoffice Documentation Team, *LibreOffice 6.0 Writer Guide*. Friends of OpenDocument, Inc., 2018.

Suggested Practical List

To be done according to the above topics.

SEC01C: App Development for Android

Course Objective

This course is designed as the first course for developing simple mobile applications that can run on Android mobiles and tablets. The students will learn the fundamentals of Android OS. They will be able to write applications which can work with databases and store the data locally.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Install and configure Android app development tools.
2. Design user interfaces for android apps.
3. Handle mobile storage through apps.
4. Design apps to handle SQLite database

Syllabus

Unit 1 Android Systems: Introduction to Mobile devices and applications, Open Handset Alliance (OHA), Overview of Android OS and architecture, installing android studio. Introduction to Android application components, Intents, Android Manifest File and its common settings, Using Intent Filter, Permissions Activities and intents: understanding activity and its life cycle, Types of intents, intent filter, context, data sharing using intent

Unit 2 Android User Interface: Basic android UI, layouts, view and view attributes, button, controls. UI events and event listeners, animations, notifications, progress dialog, Action bar, toolbar, menus and pop ups, Tab based UI, Fragment, Types of Fragment, Fragment Lifecycle, communication between fragment and activity

Unit 3 Android Storage and APIs: Android storage: Using Android Data and Storage APIs, Managing data using SQLite, Sharing Data between Applications with Content Providers
Android APIs: Multimedia, Using Android Networking APIs, Using Android Web APIs, Using Android Telephony APIs, android location based services.

Note: Kotlin will be used for the implementation.

References

1. Meier Reto and Ian Lake, *Professional Android*, 4th edition, Wrox, 2018.
2. Rick Boyer, *Android 9 Development Cookbook*, Packt Publishing Limited, 2018.

SEC01D: Basics of Programming in Python

Course Objective

This course is designed as the first course that introduces programming concepts using Python. The course focuses on the development of Python programming to simple solve problems of different domains.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Understand the basics of programming language
2. Develop, document, and debug modular Python programs.
3. Apply suitable programming constructs and built-in data structures to solve a problem.

Syllabus

Unit 1 Introduction to Programming: Problem solving strategies; Structure of a Python program; Syntax and semantics; Executing simple programs in Python.

Unit 2 Creating Python Programs: Identifiers and keywords; Literals, numbers, and strings; Operators; Expressions; Input/output statements; Defining functions; Control structures (conditional statements, loop control statements, break, continue and pass, exit function), default arguments.

Unit 3 Built-in data structures: Mutable and immutable objects; Strings, built-in functions for string, string traversal, string operators and operations; Lists creation, traversal, slicing and splitting operations.

References

1. Taneja, S., Kumar, N. *Python Programming- A modular Approach*, Pearson Education India, 2018.
2. Balaguruswamy E., *Introduction to Computing and Problem Solving using Python*, 2nd edition, McGraw Hill Education, 2018.

Additional References

1. Brown, Martin C., *Python: The Complete Reference*, 2nd edition, McGraw Hill Education, 2018.
2. Guttag, J.V., *Introduction to computation and programming using Python*, 2nd edition, MIT Press, 2016.

Suggested Practical List

1. WAP to find the roots of a quadratic equation
2. WAP to accept a number 'n' and
 - a. Check if 'n' is prime
 - b. Generate all prime numbers till 'n'
 - c. Generate first 'n' prime numbers
 - d. This program may be done using functions
3. WAP to create a pyramid of the character '*' and a reverse pyramid

```
*
***
*****
*****
*****
*****
*****
*****
***
*
```

4. WAP that accepts a character and performs the following:
 - a. print whether the character is a letter or numeric digit or a special character
 - b. if the character is a letter, print whether the letter is uppercase or lowercase
 - c. if the character is a numeric digit, prints its name in text (e.g., if input is 9, output is NINE)
5. WAP to perform the following operations on a string

- a. Find the frequency of a character in a string.
 - b. Replace a character by another character in a string.
 - c. Remove the first occurrence of a character from a string.
 - d. Remove all occurrences of a character from a string.
6. WAP to swap the first n characters of two strings.
 7. Write a function that accepts two strings and returns the indices of all the occurrences of the second string in the first string as a list. If the second string is not present in the first string then it should return -1.
 8. WAP to create a list of the cubes of only the even integers appearing in the input list (may have elements of other types also) using the following:
 - a. 'for' loop
 - b. list comprehension

SEC02A: Backend Web Development

Course Objective

The course aims at introducing the basic concepts and techniques of server side web programming. The student shall be able to develop interactive and dynamic websites using HTML, PHP and MySQL.

Course Learning Outcomes

On successful completion of this course, the student will be able to:

1. build interactive and dynamic websites.
2. write the server side programming techniques with PHP for accessing the contents to/from the server
3. learn to validate server-side/backend data
4. use GET and POST methods for sending data within client and server.

Unit 1 Introduction to PHP: Basic Syntax, defining variables and constants, data types,

operators and expression, decision making statements, loop making constructs, mixing decisions and looping with HTML

Unit 2 String Handling: Creating a string and accessing its content, searching and replacing content of a string, formatting a string, string built-in-functions

Unit 3 Handling HTML Form with PHP: Creating a form, submitting data to the server at the backend using GET and POST methods, GET vs POST methods, create a student' registration form

Unit 4 Database Connectivity with MySQL: Connectivity with database, database creation, creating tables, insertion and retrieval of the data from the database, data manipulation.

References

1. Nixon, R., *Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5*, O'Reilly, 2018.
2. Murach J, Murach's, *PHP and MySQL*, 2nd Edition, Mike Murach & Associates, 2014.
3. Holzner S., *PHP: The Complete Reference*, McGraw Hill, 2017.

Suggested Practical List

1. Write a PHP script to print the sum of odd digits of a number.
2. Create a web page having two radio buttons. One for checking whether the given year is a Leap Year or not and another button to check whether the given number is a Palindrome or not. Write a PHP script for the functionality of each button.
3. Write a script in PHP to display a Multiplication Table using nested for loop.
4. Write a script in PHP to display simple Pyramid pattern:

```
*
* *
* * *
* * * *
* * * * *
```
5. Write a PHP script to perform following string operations using in-built functions and built an interactive web page having buttons for each of the following operation:
 - a. Find the length of a string
 - b. Find a substring from a string
 - c. Replace text within a string
 - d. Remove whitespace and other predefined characters from both sides of a string.

- e. Check if a value is a string
 - f. Reverse a given string
 - g. Convert the first character of each word in a string into uppercase.
6. Design a Login form and validate that form using PHP code (Server side validation). Display message box when data is empty otherwise redirect to next page and display “Welcome username!”.
7. Design a student registration form, using appropriate input fields consisting of following:
- a. First Name
 - b. Last Name
 - c. Gender
 - d. Roll Number
 - e. Department
 - f. Societies
- Submit and retrieve the form data using \$_POST, \$_GET variable.
8. Write PHP Code to make connection to MySql database, create database and tables and access their contents.
9. Write PHP code to insert, delete, and select the data from the database. Create proper forms for performing the above operations. Display the messages such as “The record is added in the database!” when data is inserted into the database, “A record is deleted from the database” when data is deleted from the database. Use appropriate button names such as Add Data, Delete Data, and Display Data.

SEC02B: Data Analysis using Spreadsheet

Course Objective

The course will help students to learn how to analyze data with spreadsheets. They will learn about referencing, charts, functions and various utilities.

Course Learning Outcomes

On successful completion of this course, a student will be able to:

1. perform data analysis and manipulation in a spreadsheet.
2. use built-in mathematical functions in a spreadsheet.
3. perform what-if analysis using Goal seek, ASAP utility add-ins in spreadsheets.
4. sort and filter data.
5. protect a spreadsheet

Unit 1 Introduction to Spreadsheets: workbook and worksheets; relative, absolute and mixed referencing; mathematical, statistical and database functions, nested functions, regular expressions in functions; pivot table.

Unit 2 Charts: Data visualization using built-in charts.

Unit 3 Utilities: What-if scenarios, goal-seek, solver, data validation, creating a dropdown list from a range of cells, data filtering and sorting, calculations using linked sheets, detective tools.

Unit 4 Protection: passwords and digital signatures in Spreadsheets.

References

1. Mali, L., *Libre Office 5.1 Writer, Calc: Math Formula Book* - Vol 1. Notion Press, 2017.
2. Libre Office 7.0 Calc Guide.
3. Libre Office 4.1 Calc Guide The Document Foundation.

Suggested Practical List

To be done according to the above topics.

SEC02C: App Development for iOS

Course Objective

This course is designed to enable a student to learn basics about iOS app development for iPhones/ iPads/ iPods. The course provides an overview of Swift programming language, which is used to design iOS apps. The students will be able to design apps with simple user interface. They will also be able to handle sensors like camera, microphones and GPS through these apps.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Learn about the Xcode app architecture

2. Define key programming terms relevant to Swift programming language
3. Develop iOS apps for user interfaces
4. Explain iOS API features to including location, sensors and gestures

Syllabus

Unit 1 iOS Technology Stack: Introduction to iOS technology stack: iOS architecture, StoryBoard, features of Xcode, components of iOS SDK. Introduction to swift: data types, variables, control flow and operators, Collections and functions in swift, classes and structures, inheritance, closure and. enumerations

Unit 2 User interactions: Controls, gesture organizers, touching views, Core Location and Mapkit, using Google Maps in iOS. Sensors in iOS. Data persistence: Core Data framework for storing persistent data, CRUD operations.

References

1. Craig Grummitt, *iOS Development with Swift*, Manning publications.

SEC02D: Introduction to Object Oriented Programming using Python

Course Objective

This course is designed for advanced programming concepts using Python. The course focuses on the development of Python programming using built-in data structures like lists, sets, tuples and dictionaries to solve problems of different domains. It also introduces the concept of object-oriented programming and file handling.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Develop, document, and debug modular Python programs.
2. Apply suitable programming constructs and built-in data structures to solve a problem.
3. Use and apply various data objects in Python.

4. Use classes and objects in application programs and handle data files.

Syllabus

Unit 1 Built-in data structures: List operations, tuples, sets, dictionaries and their operations, passing data structures to a user-defined function.

Unit 2 Object Oriented Programming: Introduction to classes, objects and methods; Standard libraries.

Unit 3 File and exception handling: File handling through libraries; Errors and exception handling.

References

1. Taneja, S., Kumar, N. *Python Programming- A modular Approach*. Pearson Education India, 2018
2. Balaguruswamy E. *Introduction to Computing and Problem Solving using Python*, 2nd edition, McGraw Hill Education, 2018

Additional References

1. Brown, Martin C. *Python: The Complete Reference*, 2nd edition, McGraw Hill Education, 2018.
2. Guttag, J.V. *Introduction to computation and programming using Python*, 2nd edition. MIT Press, 2016.

Suggested Practical List

1. Write a function that accepts two lists and returns the merged list in sorted order
2. WAP to read a file and
 - a. Print the total number of characters, words and lines in the file.
 - b. Calculate the frequency of each character in the file. Use a variable of dictionary type to maintain the count.
 - c. Print the words in reverse order.

- d. Copy even lines of the file to a file named 'File1' and odd lines to another file named 'File2'.
3. WAP to define a class Point with coordinates x and y as attributes. Create relevant methods and print the objects. Also define a method distance to calculate the distance between any two point objects.
4. Write a function that prints a dictionary where the keys are numbers between 1 and 5 and the values are cubes of the keys.
5. Consider a tuple t1=(1, 2, 5, 7, 9, 2, 4, 6, 8, 10). WAP to perform following operations:
 - a. Print half the values of the tuple in one line and the other half in the next line.
 - b. Print another tuple whose values are even numbers in the given tuple.
 - c. Concatenate a tuple t2=(11,13,15) with t1.
 - d. Return maximum and minimum value from this tuple
6. WAP to accept a name from a user. Raise and handle appropriate exception(s) if the text entered by the user contains digits and/or special characters.