Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: **C & CPP**   **Credit Hours:**   **30**

1) First C Program

- -Header Files
  - example: #include <stdio.h>
- main()
- Curly braces { }
- printf()
- semicolon ;

- Compiling a C program
  - example: gcc filename.c -o output parameter
- Executing a C program
  - example: ./output parameter
- Errors

2) First C++ Program

- Header files
  - --example: #include <iostream>
- main()
- Curly braces { }
- -cout<<
- semicolon ;

- Compiling a C++ program
  - example: g++ filename.cpp -o output parameter
- Executing a C program
  - example: ./output parameter

3) Tokens in C and C++

- Data types, constants, identifiers

- Keywords
  - example: if, break, else
- Constants
- Data types
  - example: int, float, char, double
- Format specifiers
  - example: %d, %f, %c, %lf
- Range of data types
- Variables
- Identifier
- Errors

4) Functions in C and C++

- What is a function
- Syntax for declaration of a function
- Function with arguments
  - example: return-type function-name(parameter);
- Function without arguments
  - example: return-type function-name;
- Calling a function
- Errors

5) Scope of Variables in C and C++

- Introduction
- Syntax of declaring a variable
  - example: data-type var-name;
- Syntax for initializing a variable
  - example: data-type var-name = value;
- Scope of variables
- Global variable
- Local variable
- Error

6) If And Else If statement in C and C++

- What are Statements.
- Syntax for if and
- If-else Statement
- Errors

7) Nested if and switch statement in C and C++

- Nested if statement.
- Switch statement.
- Syntax for nested-if statement
- Syntax for switch statement
- break statement
- Comparison between nested if-else and switch statement
- Errors

8) Increment and Decrement Operators in C and C++

- Increment Operator
  - example: ++
- Postfix increment
  - example: a++
- Prefix increment
  - example: ++a
- Decrement Operator
  - example: --
- Postfix decrement
  - example: a--
- Prefix decrement
  - example: --a
- Typecasting
- Errors

9) Arithmetic Operators in C and C++

- Arithmetic Operators
- Addition Operator
  - example: a + b
- Subtraction Operator
  - example: a - b
- Multiplication Operator
  - example: a * b
- Division Operator
  - example: a \ b
- Modulus Operator
  - example: a % b

- Errors

10) **Relational Operators in C and C++**

- Double Equal to
  - example: a == b
- Not Equal to
  - example: a != b
- Greater Than
  - example: a > b
- Less Than
  - example: a < b
- Greater than Equal To
  - example: a >= b
- Less Than Equal To
  - example: a <= b
- Errors

11) **Logical Operators in C and C++**

- And &&
- Or ||
- Not !
- Errors

Intermediate level

12) **Loops in C and C++**

- Loops
- Syntax for while and do-while loop
- Comparison of while and do-while loop
- Syntax for
- for loop
- Errors

13) **Arrays in C and C++**

- What are arrays
- 1-D Arrays
- Syntax for Declaration of arrays
  - example: data type array_name [size];

- Syntax for Initialization of arrays
  - example: data type array_name [size]=value;
- Accepting values from the user
- Errors

14) **Working with 2-D Arrays in C and C++**

- What are 2-D Arrays.
- Range of arrays
- Syntax for Declaration of 2-D arrays
  - example: data type array_name[row][column];
- Syntax for initialization of 2-D arrays
  - example: data type array_name[row][column]=

{

{row-val},{col-val}

};

- Errors

15) **Strings in C and C++**

- What is a string
- Syntax for declaring a string
- Syntax for initializing a string
- To read a string from keyboard
- Errors

16) **String Library Functions in C and C++**

- What are string library functions.
- Types of string library functions
  - Strcpy
  - Strlen
  - Strcmp
  - Strcat
- Errors

Advanced level

17) **Working with Structures in C and C++**

- Introduction
- Syntax of structures

- Declaration and initialization
- Declaration of structure variable
- Accessing structure variables

18) **Understanding Pointers in C and C++**

- Introduction
- Syntax of Pointer
    - example: int *iptr;
- Declaration
    - example:

int a; (integer a) int *aptr; (pointer to an integer *aptr) aptr = &a; (aptr set to address of a) Address Pointer

- Errors

19) **Function call in C and C++**

- types of function calls
- function pass by value
- function pass by reference

20) **File Handling in C**

- File handling functions
- Opening a File closing a file
    - example: fopen, fclose
- Reading data from a File

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Akanksha Saini

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

भारतीय प्रौद्योगिकी संस्थान मुंबई
पवई, मुंबई-400 076, भारत

Indian Institute of Technology Bombay
Powai, Mumbai-400 076, India

दूरभाष/Phone : (+91-22) 2572 2545
फैक्स/Fax : (+91-22) 2572 3480
वेबसाईट/Website : www.iitb.ac.in

IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: **Front Accounting**   **Credit Hours:** **30**

Basic Level

1. **Overview of Front Accounting**

   - Introduction to Front Accounting

   - Prerequisites for learning Front Accounting

   - OS and FrontAccounting version used

   - Features of FrontAccounting

   - Who can use FrontAccounting?

   - Glimpse of Spoken Tutorials available on FrontAccounting

   - Future tutorials in FrontAccounting

   - About Spoken Tutorial project

   - About Spoken Tutorial workshops

   - Timed discussion forum of Spoken Tutorials

2. **Installation of FrontAccounting on Linux OS**

   - Installation FrontAccounting on Linux OS

   - System prerequisites for installation

   - Demonstration on how to install XAMPP on Linux

   - Start the XAMPP services

   - Download FrontAccounting software

   - Move the folder account into the web server's root directory

   - Create Users and Database for FrontAccounting

   - Steps to follow to do the installation

   - Login to the FrontAccounting interface

   - Additional Reading material for solutions if port error in XAMPP services

   - About Spoken tutorial project

3. **Installation of FrontAccounting on Windows OS**

   - Installation FrontAccounting on Windows OS

   - System prerequisites for installation

   - Demonstration on how to install XAMPP on Windows

   - Start the XAMPP services

- Download FrontAccounting software
- Move the folder account into the web server's root directory
- Create Users and Database for FrontAccounting
- Steps to follow to do the installation
- Login to the FrontAccounting interface
- Additional Reading material to resolve Port error in XAMPP services
- About Spoken tutorial project

4. **Setup in FrontAccounting**

- Various modules in setup
- Create our own Organization or Company
- How to add home currency?
- What is fiscal year?
- About fiscal year setup
- Update Company setup
- Increase the Login timeout option
- New user accounts Setup
- Setup access permission
- How to change the date format?

5. **Banking and General Ledger in FrontAccounting.**

- Various modules in Banking and General Ledger
- About Charts of Accounts
- Why we need GL Accounts, Groups and Class?
- How to create new GL Class?
- About Class ID and Class Name
- List of various Class types
- List of various Class names
- How to create new GL Groups?
- Demonstration of adding Unique ID
- How to create new GL Accounts?

6. **Journal Entry and Balance sheet in FrontAccounting**

- Various modules in Banking and General Ledger
- How to introduce capital in the business?
- Example for journal entry
- Process journal entry
- View the journal entry
- Enter a new journal entry
- Reference number for each journal entry
- How to add an attachment for the journal entry
- Balance sheet Drill down option

- How to void a transaction?

7. **Items and inventory in FrontAccounting**

- Various modules in Items and inventory
- What are items and inventory?
- How to set units of measure?
- How to set the Item Categories for items?
- Default Item categories in FrontAccounting
- Add a new Item category for Finished Goods
- Add a new Item for Finished Goods
- Assign sales prices to individual sales item
- Different Item types
- Different Sales types

8. **Taxes and Bank Accounts in FrontAccounting**

- About Taxes and Bank Accounts
- How to add GST and Sevice tax for a company?
- About Sale and Purchase GL Account for tax
- How to add new GL Account for service tax?
- Set up a Bank Account for the company
- Set Current account and Petty Cash account details
- Deposit amount in the company's Bank Account
- View the GL postings for the Deposit
- How to transfer petty cash
- Reconcile the Bank Account

9. **Setup for Sales in FrontAccounting**

- About Sales
- How to Setup Sales
- Different type of Sales Types
- How to add Sales Person and Sales Areas
- About Provision and Turnover Break point Level
- About Customers
- Add and Manage Customers
- To check whether default branch is added or not
- Customer Branch
- About Mailing address and Billing address for customer

10. **Place Sales Order in FrontAccounting**

- Enter the sales quotation entry for a customer
- Sales order inclusive of GST
- About shipping charges
- Place Quotation

- Make Sales Order against the Quotation
- Place Order
- Make delivery against the order
- Process dispatch
- Sales Order Inquiry
- Customer Payment

11. **Purchase and Reports in FrontAccounting**

- Various modules under Purchases
- Add new supplier
- Purchase Order Entry
- Place Order
- Receive Items on the Purchase Order
- Process Receive items
- Purchase invoice for the receival
- Supplier payment for the invoice
- Payment Process
- Various reports related to the transactions

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: <u>Java</u>   **Credit Hours:**   30

1. Getting started with Java – installation

Install jdk from Synaptic Package Manager, Choose openjdk-6-jdk from the list of packages available, Mark it for installation, The installation will take a few seconds

Verify the installation

At the command prompt type java -version, so the version number of the jdk will be displayed

Run a simple java program and see if it works

Type javac TestProgram.java for compiling the code and java TestProgram for executing the code

1. **Java - First program**
    1. write simple java program
    2. print "My First Java Program!" on Console
    3. save the file
    4. file name given to the java file
    5. compile the file
    6. run the file
    7. correct the errors
    8. naming conventions for class
    9. naming conventions for method
    10. naming conventions for variable

    Eclipse

2. Installing Eclipse
    1. Install Eclipse on Ubuntu on the Terminal
    2. Set up the proxy on the Terminal

3. Then fetch the list of all the available softwares

4. Type sudo apt-get update

5. Then install eclipse on the Terminal

6. Type sudo apt-get install eclipse

7. Verify if Eclipse is installed on the system

8. Installing Eclipse on Debian,Kubuntu,Xubuntu

9. Installing Eclipse on Redhat

10. Installing Eclipse on Fedora,centos and suse linux

3. Getting started with Eclipse

1. Eclipse is an Integrated Development Environment

2. It is a tool on which one can write, debug and run java programs easily

3. Open Dash Home and type Eclipse in the search box.

4. We get Workspace Launcher

5. On clicking on Workbench we get the Eclipse IDE

6. Go to File->New->Project and select Java Project

7. Create a project named EclipseDemo and create a class inside DemoClass

8. Learn about Package Explorer and Editor portlet

4. Hello World Program

1. Open Eclipse

2. Create a Java Project named DemoProject

3. Create a class named DemoClass

4. Class name and file name will be the same

5. Eclipse suggests various possibilities as we type a command

6. Eclipse also completes the parentheses by automatically adding the closing parentheses

7. Include the statement that we want to print

8. Eclipse also completes the quotes by adding the closing quote

9. Compile and execute the program

10. Change the code to print

5. Errors and Debugging

1. When writing a Java Program, here is a list of typical errors:

2. Missing semicolon(;)

3. Missing double quotes(".")

4. Mis-match of filename and classname

5. Typing the print statement n lower case

6. The line which has the error will be indicated with a red cross mark on the left margin

7. The list of errors is displayed by hovering the mouse over the cross mark

8. Create a class ErrorFree with Errors, debug the code and run it

9. Eclipse also offers intelligent fixes

6. Programming features of Eclipse

Auto completion, Sets the corresponding closing brace when we open the brace, Provides a drop-down list of methods when you start typing the code. Syntax highlighting, Classname is highlighted in pink color and method in blue color.Keyboard shortcuts, F11 to debug a program and Ctrl plus H to search a specific file. Error highlighting Cross symbol in the program denotes errors, Remove semicolon and error details are displayed when mouse is hovered over cross symbol.

Fundamental Programming Structures in Java

7. Numerical datatypes

Define datatypes and numerical datatypes

Int, float, byte ,short, long, double,range & size of each numerical datatypes, declaration and initialization of numerical datatypes, valid and invalid declaration

8. Arithmetic Operations

Define an operator, Define arithmetic operators, addition, subtraction, multiplication, division, modulo, simple program to demonstrate arithmetic operators, appropriate datatypes for appropriate values, save, compile and run the program

9. Strings

char datatype, letter, digit, punctuation marks, **tab**, or a space are all characters.Program explaining the variable and the character data.Introduction to strings, Creating string by Direct Initialization, Creating string by using new operator, Method of string class String length(), String concat(), String toUpperCase(), String toLowerCase()

10. Primitive type conversions
    1. define type conversion or type casting
    2. higher order integer to lower order integer- Explicit type casting
    3. program to show explicit type casting
    4. common mistake in explicit type casting.
    5. program to show common mistake in explicit type casting
    6. lower order integer to higher order integer – Implicit type casting
    7. program to show implicit type casting
    8. char to integer
    9. integer to char
    10. program to show char to int type casting.

Control flow

1. define switch case statement
2. compare switch and nested if
3. switch case syntax
4. working of a switch case statement
5. use of keyword switch
6. valid and invalid use of keyword **case**
7. use of keyword default
8. use of keyword break
9. program to demonstrate switch case statement
10. save, compile and run the program to check the output

16. while loop
    1. Loop control statement
    2. types of loop control statements
    3. Introduction to while loop
    4. syntax of while loop
    5. Program using while loop
    6. Check the output.
    7. Introduction to infinite loop
    8. loop variable modification
    9. Check the output
    10. How to terminate the infinite loop

17. for loop
    1. syntax, introduction to for loop
    2. for loop syntax
    3. loop variable
    4. loop condition
    5. loop variable increment or decrement
    6. loop block
    7. flow of loop
    8. advantage of using loop

18. do while loop
    1. define do while
    2. do while syntax
    3. working of do while loop
    4. example of do while loop
    5. explain the do while programming
    6. save, compile and run the program to check the output
    7. how different is it from the while loop
    8. program to demonstrate the differences

Arrays

19. Introduction to Arrays
    1. About Arrays
    2. Declare an Array
    3. Initialize an array
    4. Intilalization using for loop
    5. Index of an array elements
    6. change values of an array
    7. print the value of an array
    8. Advantage of an array.

20. Array operations
    1. import java.util.Arrays
    2. use methods from class Arrays
    3. toString() method
    4. sort() method
    5. fill() method
    6. copyof() method
    7. copyofRange() method
    8. about parameters for each method.

Classes & Objects

21. Creating class
    1. Whatever we can see in this world are all objects
    2. Objects can be categorized into groups known as class
    3. This is class in real world
    4. Human Being is an example of class in real world
    5. Class in java is the blue print from which individual objects are created
    6. Class consists defines a set of properties called variables and a set of behaviors called methods
    7. Syntax for creating class
    8. Create a simple class Student using Eclipse
    9. The Student class can contain properties

22. Creating Object
    1. An object is an instance of a class
    2. Each object consist of state and behavior
    3. Object stores it state in fields or variables
    4. It exposes its behavior through methods
    5. Reference variables

6. Create a class named TestStudent

7. Create an object of the Student class

8. Use new operator

9. Check what the reference variable contains

10. Create one more object of the Student class and check what the reference variable contains

23. Instance fields

1. Also known as non-static fields

2. Open the TestStudent class which we have created

3. Access the fields roll_number and name using dot operator

4. See the output

5. Initialize the field and see the output

6. Change the modifier of the fields to private

7. Debug the error that you get

8. Change the modifier to protected

9. Each object of a class will have unique values

10. Create two objects of the Student class

24. Methods

1. method definition

2. write simple method

3. method returning value

4. call a method in another method

5. flow of the program

6. call a static method

7. call a method from another class

8. method signature

9. method body

Constructors

25. Default constructor

1. what is a constructor?

2. what is a default constructor?

3. when is it called?

4. define a constructor

5. initialize the variables

6. call the constructor

7. difference between constructor and method

26. Parameterized constructors

1. What is a parameterized constructor?

2. create constructor without parameter

3. create a constructor with parameter

4. assign values to the variables in the constructor

5. pass arguments during the constructor call

6. working of parameterized constructor

7. show common errors

8. resolve the errors

9. create another parameterized constructor

10. why to use constructor?

27. Using this keyword

1. this is a reference to the current object

2. helps to avoid name conflicts

3. we can use this keyword inside a constructor to call another one

4. the constructors must be in the same class

5. explicit constructor invocation

6. Explain it using the parameterized constructor code

7. Make this statement the last one in the constructor

8. You will get an error

9. this statement should be the first one inside a constructor

28. Non-static block

1. Non-static block

2. Any code written between two curly brackets

3. Executed for each object that is created

4. Executes before constructor's execution

5. can initialize instance member variables of the class

6. create a class named NonStaticTest

7. Create a non-static block and a constructor inside it

8. Check the output

9. Include multiple non-static blocks

10. they will be executed in the sequence in which they appear in the class

11. Check the output

12. Non-static block is not a substitute for constructor

29. Constructor Overloading
    1. define multiple constructor
    2. what is constructor overloading?
    3. constructor with different number of parameters.
    4. parameters with different datatypes.
    5. how is constructor overloaded?
    6. flow of overloading process.
    7. advantage of constructor overloading.
30. Method Overloading
    1. define multiple methods.
    2. methods with same name.
    3. methods with different number of parameters.
    4. methods with different datatypes of parameter.
    5. what is method overloading?
    6. example for overloadin method
    7. how to overload method?
    8. advantage of method overloading.
    9. error in method overloading.
31. Taking user input in Java
    1. What is BufferedReader?
    2. Importing three classes from Java.io package
    3. How to take the input from the user?
    4. Syntax to implement BufferedReader
    5. What is InputStreamReader?
    6. Create object of InputStreamReader
    7. Create object of BufferedReader
    8. About IOException
    9. About throws keyword
    10. Typecasting

Intermediate Level

1. Subclassing and Method Overriding
    - Definition of subclassing
    - Demo of subclassing using an Employee and Manager class
    - Single inheritance
    - Use of extends keyword
    - Private members in a super class
    - Definition of method overriding

- Annotation
- @Override Annotation

2. Calling methods of the superclass
   - super keyword
   - Call methods of the super class
   - Constructor of the super class
   - Demo of super keyword using an Employee and Manager class
   - Single inheritance
   - Use of extends keyword
   - Private members in a super class

3. Using final keyboard
   - final keyword
   - What is final keyword and its application?
   - Where final keyword can be declared?
   - final variable
   - final static variables
   - static block
   - final variable as parameter
   - final method
   - private final method
   - final class

4. Polymorphism
   - Polymorphism in Java
   - Run-time polymorphism
   - Virtual Method Invocation
   - Compile-time polymorphism
   - Role of JVM
   - IS-A test
   - Static binding
   - Dynamic binding

5. Abstract Classes
   - Abstract Classes in Java
   - What are Abstract Methods
   - What are Concrete Methods
   - Properties of Abstract Methods and Abstract Classes
   - How to use Abstract Methods

6. Java Interfaces
   - Java Interfaces
   - Implementing Interface
   - Implementation Classes
   - Interfaces Vs Abstract classes
   - Implementing Multiple Interfaces
   - Usage of Interfaces with an example

7. Static Variables
   - What is Static Variable in Java?
   - Usage of Static Variables with Example
   - Static Variables Vs Instance Variables
   - Final Static Constants

8. Static Methods
   - What is static method in Java?
   - Static methods Vs Instance methods
   - Usage of static method with example
   - Passing object variables in a static method

9. Static Blocks
   - What is a static block
   - Declaring and defining a static block
   - How static blocks are invoked and executed

10. Exception Handling
    - What is an Exception
    - Types of Exceptions
      - Checked Exceptions
      - Unchecked Exceptions
    - Explaining *ArrayIndexOutOfBoundsException*
    - Demonstrating Checked Exceptions with example
    - Handling Exceptions using try-catch blocks

- Explaining *ArithmeticException*

- Demonstrating Unchecked Exceptions with example

- Explaining *FileNotFoundException*

- Usage of finally block

- Explaining *NullPointerException*

1. Custom Exceptions

  - What is a Custom Exception

  - Demonstration of custom exception

  - Custom exception example - "InvalidMarkException"

  - Usage of "throw" keyword

  - How to resolve errors in custom exceptions

  - Resolve error using "Add throws declaration" option

  - Usage of "throws" keyword

  - Example for "FileNotFoundException"

  - How to handle multiple exceptions

  - How to use "surround with try/catch" option

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

भारतीय प्रौद्योगिकी संस्थान मुंबई
पवई, मुंबई-400 076, भारत

Indian Institute of Technology Bombay
Powai, Mumbai-400 076, India

दूरभाष/Phone   : (+91-22) 2572 2545
फैक्स/Fax   : (+91-22) 2572 3480
वेबसाइट/Website : www.iitb.ac.in

IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: <u>Libre Office base</u>   **Credit Hours:**   30

Basic Level

1. Introduction
   - What is LibreOffice Base?
   - What can you do with Base?
   - Prerequisites for using Base
   - Relational Database basics.
   - Create a new database
   - Create a table

2. Tables and Relationships
   - Adding data to a table.
   - Define and create relationships.

3. Create a simple form
   - What is a form?
   - Create a form using the Wizard.

4. Modify a simple form
   - How to enter data in a form
   - How to Modify a form

5. Build a complex form with form controls
   - Ungroup form elements.
   - Rename labels.
   - Change fonts and sizes.
   - Change location of the form elements.

6. Add List Box form control to a form
   - How to modify a form using LibreOffice Base.

7. Add Push Button to a form
   - How to Add a Push Button to a form

8. Enter and update data in a form
   - How to enter data and update data using a form.

9. Create queries using Query Wizard

- Create simple queries using,the Query wizard Select fields

- Set the sorting order of the fields,

- And provide search criteria or conditions for a query.

- What a query is.

10. Create queries using Design View

- Create a query by using the Design View.

- Add tables to the Query Design window Select fields

- Set up aliases,

- Set up sorting order And provide search criteria for a query.

11. Create reports

- Create a Report

- Select, Label and Sort the report fields

- Select report layout

- choose report type: static or dynamic

12. Modify a Report

- Modify a report by customizing the layout

- Look and feel of the report.

Intermediate Level

1. Create tables

- Create view.

- Create a table by copying an existing table.

2. Create Subforms

- How to Create a Subform

3. Create simple queries in SQL View

- Create Simple Queries in SQL View,

- Write simple SQL

- Use SELECT, FROM , and WHERE clauses

- Choose upper, lower, or mixed cases for naming fields and tables

4. Create simple queries in SQL View II

- Write Queries in SQL View

- Use ORDER BY clause

- Use JOINS

- Use Aggregate functions

- Use the GROUP BY clause

- Use built in Functions

Advanced Level

1. Access data sources
   - Access a data source
   - Register *.odb databases.
   - View data sources.
   - Use data sources in Writer

2. Database Maintenance
   - Modify data structures.
   - De-fragment database.
   - Backups.

3. Indexes, Table Filter, SQL Command window
   - Indexes
   - Table Filter
   - And the SQL Command window

4. Database Design, Purpose, OrganizeTables
   - Determining the purpose of our database
   - Finding and organizing the information required
   - Dividing the information into tables.
   - What is Database Design

5. Database Design, Primary Key and Relationships
   - Turn information items into columns
   - Specify primary keys
   - Set up the table relationships

6. Design Refine Database Design and Normalization Rules
   - Refine the database design
   - Apply the normalization rules
   - Test the database design

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Akanksha Saini

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: **Libre Office Calc**   **Credit Hours:** **30**

Basic Level

1. Introduction to LibreOffice Calc

   - What is Calc,Who should use Calc,What can be done using Calc.

   - About spreadsheets,sheets and cells.

   - Basic features – parts of main Calc window - Title & Menu Bar, Standard Toolbar, Formatting Toolbar, Formula Bar, Status Bar

   - What is a cell, row, column, Create new document, enter data in one cell, save in Calc(CSV and other formats),opening/closing document.

   - Save as ods, csv, xls, xlsx and export to PDF

2. Working with Cells

   - How to enter numbers,text,numbers as text,date and time in Calc.

   - How to use the Format Cells dialog box.

   - How to Navigate between cells and in between sheets.

   - How to select items in rows,columns and sheets.

3. Working with Sheets

   - Inserting and Deleting rows and columns Calc.

   - Inserting and Deleting Sheets in Calc.

   - Renaming Sheets.

4. Formatting Data

   - Borders, Color, Formatting Text, Increasing Cell Size

   - Formatting multiple lines of text,numbers,fonts,cell borders,cell background

   - Automatic Wrapping and Using manual breaks

   - Merging cells, rows and columns, issues with using merge cells.

   - Shrinking text to fit the cell

5. Basic Data Manipulation

   - Paste and paste special (values, transpose), pasting a spreadsheet into writer as a table

   - Introduction to Formulas – Sum, Average, basic formula (=a*b+c)

- Subtotal
- Different kinds of functions available in spreadsheet
- Sorting by Columns
- Filtering Data - Basics

6. Working with data
   - Speed up using Fill tools and Selection lists.
   - Sharing content between sheets
   - Remove data,Replace data,Change part of a data

Intermediate Level

1. Using Charts and graphs in Calc
   - Creating,Editing and Formatting Charts
   - Types of charts
   - Resizing and moving of charts

2. Formulas and Functions
   - Creating formulas,operator types and referencing
   - Basic arithmetic and statistic functions - relative and fixed ($) referencing in a function
   - Rounding off numbers
   - If..else
   - Important string functions (left, right, middle, concatenate)

3. Linking Calc Data
   - Referencing other sheets and documents
   - Working with Hyperlinks

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Akanksha Saini

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

भारतीय प्रौद्योगिकी संस्थान मुंबई
पवई, मुंबई-400 076, भारत

Indian Institute of Technology Bombay
Powai, Mumbai-400 076, India

दूरभाष/Phone : (+91-22) 2572 2545
फैक्स/Fax   : (+91-22) 2572 3480
वेबसाईट/Website : www.iitb.ac.in

IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: **Libre Office Impress** **Credit Hours:** **30**

Basic Level

1. **Introduction to LibreOffice Impress**
   - Basic Features
   - Toolbars
   - Save in MS Office and other formats
   - PDF Export open MS Office Documents

2. **Creating a presentation document and basic formatting**
   - Inserting Slides
   - Copying Slides
   - Using Tabs
   - Font size and Color, Bold

3. **Viewing a Presentation Document**
   - Views options – Normal, Outline, Master Layout
   - Uses of different view options

4. **Inserting Pictures and Other Objects in a Document**
   - Insert Pictures/Hyperlinks (Within/Across Documents, from Web)
   - Creating Tables
   - Format the Pictures and Objects

5. **Printing a Presentation Document**
   - Printing slides and Handouts Different Options, when used

Intermediate Level

1. Slide Master, Slide Design
   - Creating Background for Slides
   - Different Slide Templates (Layouts)
2. Custom Animation
   - Manipulate Appearance of Lines in a Slide
3. Slide Creation
   - Slide Show
   - Slide Transition
   - Automatic Show (with Preset Timings)

Advanced Level

1. Presentation Notes
   - Use of Notes
   - Printing of Notes

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Akanksha Saini

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: **LibreOffice Writer**   **Credit Hours:**   **30**

Basic Level

1. **Introduction to LibreOffice Writer**
   - Basic Features
   - Toolbars
   - Save in MS Office, PDF and other formats
   - Open MS Office Documents

2. **Typing text and basic formatting**
   - Bullet points
   - Bold/Underline/Italics
   - Font name/Font size/Font color in Writer
   - Other important and popularly used formatting features

3. **Inserting pictures and other objects in a document**
   - Insert pictures
   - Hyperlinks (within, across documents, from web)
   - Creating tables

4. **Viewing and printing a text document**
   - Views options – print layout, web layout, full screen
   - Print preview, formatting for printing, printing all, some pages, printer settings

Intermediate Level

1. Using search replace to make document wide changes, auto correct feature to create text shortcuts
   - Search replace for select text
   - Auto-correct feature

Advanced Level

1. Using track changes as a peer review / collaborative constructivist tool, accepting and rejecting changes
   - How to use record changes to peer review documents, accept/reject these
2. Headers, Footers and notes
   - Page format – header footer, how these can change within the same document (first page without header footers)
   - Useful footer information (page number, title)
   - Footnotes and endnotes
3. Creating newsletter with multiple columns and advanced formatting
   - Advanced use as a desktop tool to create a note with multiple columns use features like word count,Spell check

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay

Syllabus IIT Bombay Certification under the Spoken Tutorial Project

Course Name: <u>Python 3.4.3</u>   **Credit Hours:** 30

Module 1: Basic Plotting

Getting started with ipython

1. Use Python 3.5.2
2. Use Ipython version 5.1.0
3. IPython is an enhanced interactive Python interpreter.
4. Invoke the IPython interpreter
5. Quit the IPython interpreter
6. Navigate the IPython session history
7. Use tab-completion to work faster.
8. See the documentation of functions using question mark.
9. Interrupt commands using Ctrl + C when we make an error.
10. round command

Using the plot command interactively

1. Use Python 3.4.3
2. Use Ipython version 5.1.0
3. Start IPython with pylab.
4. ImportError if matplotlib is not installed
5. clf() function
6. Use the linspace function to create equally spaced points in a region.
7. Find the length of sequences using len function.
8. Plot mathematical functions using plot.
9. Clear drawing area using clf.
10. Usage of buttons in the UI of the plot window such as - save, zoom, move axis, back and forward and Home

Embellishing a plot

1. Use Python 3.4.3
2. Use Ipython version 5.1.0
3. Modify the attributes of a plot
4. Pass additional keyword arguments to plot command
5. Add title to a plot using 'title' command.
6. Incorporate LaTeX style formatting by adding a $ sign before and after the string.
7. Label x and y axes using xlabel() and ylabel() commands.
8. Add annotations to a plot using annotate() command.
9. Get the limits of axes using xlim() and ylim() commands.
10. Set the limits of axes using xlim() and ylim() commands.

Saving plots

1. Use Python 3.4.3
2. Use Ipython version 5.1.0
3. Save plots using the savefig() function.
4. Save the plots in different formats like
   - pdf
   - ps
   - png
   - svg
   - eps

Multiple plots

1. Use Python 3.4.3
2. Use IPython 5.1.0
3. Draw multiple plots which are overlaid.
4. Operations on individual plots.
5. Use the figure command.
6. Distinguish between multiple overlaid plots.
7. Use the legend command.
8. Serial number of the plot to select corresponding plot.
9. Switch between the plots
10. Saving individual plots.

Subplots

1. Use Python 3.4.3
2. Use IPython 5.1.0
3. Creating subplots
4. Switching between subplots.
5. Subplot command

6. Passing arguments to subplot command.
7. First argument is the number of rows of subplots.
8. Second argument is the number of columns of subplots
9. Third argument specifies the serial number for subplot.

Additional features of IPython

1. Use Python 3.4.3
2. Use IPython 5.1.0
3. Retrieve the history using %history command.
4. View only a part of history by passing argument to %history command.
5. Pass arguments to %history to get particular lines of code
6. Save the required lines of code in required order using '%save' command.
7. Use '%run -i' command to run a saved script.

Module 2: Plotting Experimental Data

loading data from files

1. load data from file
2. single column
3. multiple columns separated by delimiter
4. cat command
5. loadtxt()
6. columns separated by spaces
7. columns separated by semi-colon
8. unpack argument
9. delimiter argument
10. three columns of data

Plotting the data

1. plotting data
2. list
3. list element-wise squaring
4. plot data points
5. clear plots
6. errorbar function
7. dots or filled circles in plot
8. plot with red pluses
9. explore documentation in ipython
10. plot with errorbars
11. using format argument

Other types of plots

1. scatter plot
2. scatter function
3. scatter plot with various arguments
4. logarithmic plot
5. loglog function
6. cat command
7. loadtxt function
8. unpack parameter of loadtxt
9. linspace
10. scatter versus plot

Plotting charts

1. Use Python 3.4.3
2. Use IPython 5.1.0
3. To produce scatter plot
4. Plot a pie chart using pie() function
5. Plot a bar chart using bar() function
6. Access the matplotlib online help
7. Charts with line hatching

Module 4: Handling Large Data Files

Getting started with lists

1. What is a list?
   - Define List
   - List index
2. Create:
   - List with elements
   - Empty list
   - List within a list
3. Find out the list length using len function
   - Access elements using their index numbers
   - Append elements to list using the function append
   - Delete element from list using the del and remove function

Getting started with for

1. For loop syntax
2. Example to use For loop

3. Indentation in for loop

4. Create blocks in python using for

5. Iterate over a list using for loop

6. How to get out of the block

7. use of Range() function

8. Range function in for loop

9. How to use Python interpreter

10. IPython interpreter to specify blocks

Getting started with strings

1. What are strings?

2. How are strings denoted in Python?

3. String concatenation

4. Multiply a string with an integer

5. Accessing individual elements of a string

6. Accessing elements of a string using negative indices

7. Split() function

8. Join() function

9. Define a string in different ways

10. Print a string repeatedly

Getting started with files

1. Open a file

2. Open() function

3. Different Modes of opening a file

4. Read() method

5. Read the content of the file line by line

6. Read the entire content of the file

7. Append the lines of a file to a list

8. Close a file

9. Demonstration using a txt file

10. Splitlines() method

Parsing data

1. What is Parsing data?

2. split function and its syntax

3. What is string tokenizing?

4. How to split a string on whitespace?

5. split function with argument

6. strip function and example

7. Converting string into floats and integers

8. Example to read a huge .txt file line by line and parse each record

9. Perform computations on the .txt file

10. Execute the file using %run command

Statistics

1. Statistical operations in Python

2. Installation of Numpy for mathematical and logical operations

3. Installation of pip to install python libraries

4. loadtxt() function with example

5. Getting the shape of an array

6. Getting the sum of a column in an array

7. How to calculate mean?

8. Calculate mean across each of the axis of the array

9. How to calculate median?

10. How to calculate standard deviation?

Module 5: Arrays and Matrices

Getting started with arrays

1. Overview of array

2. Usage of numpy library

3. How to create arrays

4. How to create two dimensional array

5. arange() method

6. reshape() method

7. How to find the shape of an array?

8. Create a new array with elements of different datatypes

9. Identity matrix

10. Zeros method

Accessing parts of arrays

1. Create a one-dimensional array

2. Create a two-dimensional array

3. Accessing individual elements of an array

4. How to change the value of an array

5. How to change more than one elements at a time

6. Negative indexing of arrays

7. Slicing of an array

8. Striding of an array

9. Access only the odd rows and columns of an array

10. Examples to demonstrate all the manipulations of arrays

Image manipulation using Arrays

1. Read images into arrays

2. How to access parts of an array?

3. imread command

4. imshow command

5. show command

6. How to check the dimensions of an array?

7. Example to access parts of an image

8. How to stride over an array?

9. Example to access an RGB image

10. Slice an image of different dimension

Basic Matrix Operations

1. Create matrices from lists

2. asmatrix method

3. arange and reshape methods

4. Basic matrix operation

5. Addition, subtraction and multiplication of a matrix

6. Determinant of a matrix

7. eye(), allclose() functions

8. Inverse of a matrix

9. eigenvalues and eigenvectors of a matrix

10. diag() function

Advanced Matrix Operations

1. flatten() function

2. Example to convert a multidimensional matrix to single dimension matrix

3. Frobenius norm of a matrix

4. Demonstration of Frobenius norm of a matrix

5. Inverse of a matrix

6. Infinity norm of a matrix

7. norm documentation

8. Singular value decomposition

9. svd() function

10. smat function

Least square fit

1. Generating a Least Square fit line
2. Generating L vs t square
3. loadtxt function
4. Usage of numpy library
5. Plotting L vs t square
6. Steps for least square fit line
7. Matrix formulation tsq=A.p
8. Generating the two matrices tsq and A
9. Finding transpose of a matrix
10. lstsq() function

Module 6: Python Language: Basics

Basic datatypes & operators

1. Data types in Python
2. Demonstration of int, float and complex data types with examples
3. Different functions associated with int data type
4. Complex numbers and their functions
5. Boolean operations with examples
6. Operator precedence with parentheses
7. Different operators available in Python3
8. Modulo operator with examples
9. How to do exponent operation in Python?
10. How to find the square root of a number in Python?

Sequence datatypes

1. List, string and tuple sequence data types with examples
2. How to access a list using index numbers?
3. Access the string elements
4. Access the tuple elements
5. How to add different sequence data types?
6. How to find the length of a variable?
7. Find the sum of a list
8. Convert list to tuple
9. Convert tuple to list
10. Convert string to list and list to string

Input/output

1. Input Output in Python

2. Various output statements

3. Print a string

4. Print a string with newline character

5. How to use format operators?

6. Example for integer format

7. Example for string format

8. Example for float format

9. Getting input from the user using Input()function

10. Display a prompt to get the input

11. Save the script as filename.py and execute using %run command

Conditionals Statements

1. if condition statement

2. Demonstration of if statement with example

3. if/else condition statement with example

4. Importance of indentation in a program

5. Usage of colon in program

6. Condition statement using elif

7. Examples using if/elif/else block

8. Ternary conditional statement

9. Difference between if/else and ternary conditional statements

10. How to use pass statement?

Loops

1. Explanation of while loop

2. Demonstration of while loop with example

3. Print the squares of all the even numbers below 10 using while loop

4. How to use for loop?

5. Print the squares of all the even numbers below 10 using for loop

6. for loop with range function

7. How to use break statement in for loop

8. pass statement in for loop

9. continue statement in for loop

10. Demonstration of pass, break and continue statements

Module 7: Python Language: Datastructures
Manipulating lists

1. Various manipulation in lists

2. Slicing of lists

3. Syntax and demonstration of slicing of lists

4. How to use step value in slicing

5. Striding of list

6. Examples with various parameters in striding of list

7. sort method in list

8. Usage of sorted() built-in function

9. Reverse a list

10. Striding with negative values

11. How to store a new reversed list in another variable

Manipulating strings

1. How to slice a string

2. Various way to get substrings using index

3. Reverse a string

4. How to check if a given string is a palindrome or not

5. Replace characters in a string

6. Convert a string to uppercase

7. Convert a string to lowercase

8. How to use for loop in a list

9. Join method

10. Join list elements to form a string

Getting started with tuples

1. What are tuples?

2. How to declare tuples?

3. Examples to declare tuples

4. Demonstration of creating tuple

5. Accessing tuples by their index positions

6. Iteration over tuples

7. Demonstration of immutability property of tuples

8. How to swap values in tuples

9. Similarities of tuples with lists

10. Tuple packing and unpacking

Dictionaries

1. Overview of dictionaries

2. Creating an empty dictionary

3. Creating a non empty dictionary

4. About key:value pair

5. How to access the dictionary elements

6. Demonstration of wrong key

7. Add, delete and modify an item in a dictionary

8. Usage of method in

9. Retrieve the keys and values by using the methods keys() and values()

10. Iterate over elements of a dictionary using a for loop

Sets in Python

1. What are sets in python?

2. Input sets

3. Create sets

4. How to create empty sets?

5. Operations on sets

6. Add and remove methods

7. Union and intersection methods

8. Difference and symmetric_difference methods

9. Subset and superset

10. Length and containership on sets

Module 8: Python Language: Advanced

Getting started with functions

1. About Functions

2. How to define a function

3. Example for defining a function

4. Calling a function with arguments

5. Calling a function without arguments

6. Return values from a function

7. Indentation in coding

8. Documenting or commenting code

9. How to use docstrings in python function

10. How to write a function circle to return area and perimeter with radius r

Advanced features of functions

1. Functions with default arguments

2. Various examples for default arguments

3. Interchanging the default and non-default arguments

4. Call a function with keyword arguments

5. Call a function without keyword arguments

6. Functions with positional arguments

7. Functions with arbitrary arguments

8. Demonstration of arbitrary arguments

9. Usage of * and ** in defining a function

10. Python built-in-functions

Using python modules

1. Python modules

2. Run a Python script from command line

3. How to import modules in python scripts?

4. How to import required functions from a module?

5. Usage of namespace

6. Advantages of using import functions

7. Using alias to the module

8. Demonstration of import functions

9. Run python scripts in ipython interpreter

10. Python standard library of modules

Writing python scripts

1. About Python modules

2. What is importing?

3. How to write a function and save it as a script

4. Run a python script

5. Import a module

6. Example to import a module

7. usage of __name__ variable

8. Write test condition using the name variable

9. How importing works in new IPython console

10. Different ways of running the Python script

Testing and debugging

1. What is software testing?

2. Write a simple function

3. How to write test cases?

4. Create simple tests for a function

5. Run the script and test the code

6. Automate tests

7. Example for test case fail

8. Coding style

9. How to give meaningful names in coding

10. Python coding standards

Handling Errors and Exceptions

1. Errors in Python
2. Syntax errors and exception
3. Exceptions with example
4. Syntax error with example
5. Demonstration of ValueError exception
6. Demonstration of ZeroDivisionError exception
7. try except clause in Python
8. What is debugging?
9. Using %debug for debugging in ipython
10. try except with else clause

For and On behalf of
Spoken Tutorials,
Indian Institute of Technology, Bombay

Mrs. Akanksha Saini
National Coordinator
Spoken Tutorial Project, IIT Bombay