

EVALUATION GUIDELINES

PG-DAC August 2019

1. Evaluation

Evaluation is a necessary and essential part of conducting the PG-Diploma in Advanced Computing, as it provides important feedback and inputs to both the institute as well as the student. The institute gets an idea about the relative performance of each student, which also serves as feedback about the design and conduct of the course. The student gets a clear picture of his academic standing, individually and in comparison to his fellow students.

In order to ensure timely and efficient evaluation and certification of all students, the following guidelines are being issued and should be followed religiously.

2. Evaluation Methodology

2.1 Each centre should have a Designated Responsible Member (DRM) for Evaluation.

2.2 The DRM Evaluation would be responsible for coordinating all activities relating to evaluation at the training centre and for communicating with CDAC ACTSPune.

2.3 Evaluation is a compulsory part of the process of obtaining the PG-DAC. All students are required to pass each module of the course in order to be eligible to receive the PG-Diploma Certificate.

2.4 The faculty of every module should outline the objectives of the evaluation to be conducted for that module, so as to enable the student to prepare himself or herself properly.

3. Modular Evaluation

3.1 Credit System: The credit-based training and grading system essentially implies a redefining of the curriculum into smaller measurable entities or '**modules**' with the hours required for '**learning**' these being at the primary focus and the development of a mechanism whereby these modules can be combined in different ways so as to qualify for a certificate Diploma. In a sense, therefore, the completion of a single 'module' of learning can pave the way for learning other modules and a combination of modules in keeping with the needs and interests of the learners illustrates the much talked about 'cafeteria approach' to learning with the Learner at the centre state of all academic transactions. Following are the features of Credit System.

3.1.1 Learner-centric approach

3.1.2 Workload estimated on time invested on learning rather than time invested on teaching

3.1.3 Workload=Theory classes + Practical Classes + Assignments + Seminar + Individual Learning + Preparation for examination

3.1.4 A program is divided into various modules or units and credit is assigned to each module/unit based on its learn ability.

3.1.5 Number of credit points assigned to a module depends upon the number of hours required to learn that module.

3.1.6 Credit Transfer Mechanism: Horizontal credit transfer(between similar level programs like Diploma to Diploma), Vertical Credit Transfer (credit transfer to higher level programs like certificate to diploma)

3.1.7 Inter institution credit transfer and intra institution credit transfer

3.1.8 Assumption: 1 credit = 25 hours of learner's load

3.1.9 Credit based evaluation will have following marking system:

- 40% end term examination + 40% Practical assessment + 20% internal assessment.
- Between 1-2 credit = 50 marks evaluation,
- Between 3-6 credit = 100 marks, if there is no lab then 50 marks evaluation (T-40+IA-10)
- Between 7-10 credit = 150 marks

3.2 Grading system:

The academic performance of a student shall be graded on a ten point scale. The letter Grades awarded to a student in all the courses shall be converted into Overall Grade Point Average (OGPA) respectively.

The method of evaluation of students in the course and the actual grading of a student's performance are based on the following guidelines.

3.3 Subject wise Evaluation

3.3.1 A separate evaluation process is to be conducted for every module of the course.

3.3.2 The evaluation for each module must be completed as per guidelines given below. The mid-module /surprise test evaluation is mandatory and can be taken after discussion with the concerned faculty.

3.3.3 Students are evaluated on a continuous and throughout the duration of the course to make a fair assessment of the skills acquired by them. To have a very uniform and fair assessment. The evaluation process is divided into two parts:

- ☐ Continuous Assessment - CA (60 marks)
- ☐ Course End Examination - CE (40 marks)

Continuous Assessment: This is being done primarily by the respective faculty in the form of Lab tests, assignments, quizzes etc conducted (with the help of the respective course coordinators) at regular intervals and as and when the portions of the modules are completed. These are basically internal exams and local to the centre. This process is further categorized into two parts.

- Lab test (40 marks)
- Internal test (20 mark): Assignment/Case Studies /quiz and other valuation methods like case study, viva, group discussion depending on the subject and the faculty (20 marks)

3.3.4 It is recommended to conduct the Effective Communications & Aptitude sessions for the benefit of the students and also conduct some surprise test for Effective Communications & Aptitude sessions.

3.3.5 The figures shown below indicate the weightage of each module in the final performance statement. The examination(s) for each module must be conducted for at least that number of marks. However, the centre may conduct evaluation for a higher number of marks, in which case the marks will be scaled down. For example, if the examination for the Operating

Systems Concepts module is conducted for 100 marks, the marks earned by the student will be scaled down to out of 40.

3.3.6 The weight age for each component will normally be:

Theory	40% (Through Centralized Course Mid and End Examination i.e. CMCE and CCEE)
Laboratory	40%
Internal	20%

(Internal Assessment: Through Lab Assignment Evaluation, Surprise Tests, attendance, Viva, Seminars)

Note: Where a module does not have a practical component, the practical component weight age will be merged with the Internal Assessment component weight age.

3.3.7 A student must score a minimum of 40 percent marks in each component of the evaluation, and also in the aggregate score, in order to successfully clear the module. If a student scores more than 40% on aggregate but has scored less than 40% in one component of the evaluation, he will not be declared as passed.

3.3.8 The figures shown below indicate the weightage of each subject in the final performance statement for PG-DAC.

Term-I (Centralized Course Mid Examination)											
Subject Code	Module Name	Contact Hours			Total Workload	Credits	Marks				
		Theory	Lab	Total			Credits	Theory (CCEE)	Lab	IA	Total Marks
DAC01	Operating System Concepts	30	30	60	100	OS: 30, Linux: 10	4	40	40	20	100
DAC02	OOPs with C++ Programming	30	30	60	100	OOPs Concepts: 10, C++: 30	4	40	40	20	100
DAC03	Algorithm & Data structures	34	36	70	120	Problem Solving: 10 Data Structures: 30	5	40	40	20	100
DAC04	Database Technologies	30	30	60	75	DBMS: 10, SQL: 16, SQL Programming:10,NoSQL: 4	3	40	40	20	100
Term-II (Centralized Course End Examination)											
DAC05	Advanced Software Development Methodologies	46	44	90	125	SE: 8, DevOps: 12, UML: 3, Git:5, Testing:7,Cloud-5	5	40	40	20	100
DAC06	Advanced Web Programming	24	26	50	100	Architecture of Web: 2,HTML5: 4, CSS: 4, PHP: 10, XML: 8, JSON:6, Web Security: 4, UI: 2	4	40	40	20	100
DAC07	MEAN Stack	30	30	60	90	40	4	40	40	20	100
DAC08	Java Technologies-I (Core Java)	34	36	70	125	Core Java: 40	5	40	40	20	100
DAC09	Java Technologies-II (Web Based Java)	42	48	90	150	Advanced Java: 40	6	40	40	20	100
DAC10	Microsoft .NET 2010	32	38	70	90	Framework: 5, C#.NET: 15, ASP.NET: 20	3	40	40	20	100
DAC11	Effective Communication	50	-	50	75	-	3	-	-	50	Grade
DAC12	Aptitude & General English	50	-	50	75	-	3	-	40	10	Grade
DAC13	Project	-	120	120	275	-	11	-	-	200	Grade
	Total	-	-	900	1500	60	400	360	440	900	

3.3.9 Overall 75% attendance is required for a student to become eligible for the CCME and CCEE

3.4 General guidelines for award of grades:

The marks of all the components of a subject shall be added to get total marks out of 100. The rounding off shall be done on the higher side.

The grades shall be awarded on the basis of absolute marks, as mentioned in

Table-1.

Marks	Grade Points	Grade
90 and above	10	A
80 to Less than 90	9	B
65 to Less than 80	8	C
50 to less than 65	7	D
40 to less than 50	6	E
Less than 40	0	F

3.5 Calculation of Overall Grade Point Average:

OGPA will be calculated by following formula:

$$OGP = \frac{\sum_{i=1}^n C_i G_i}{\sum_{i=1}^n C_i}$$

Where C_i is number of credit hours assigned to a module i and grade point secured by the students is (G_i) in that module.

OGPA is calculated by multiplying the number of credit hours assigned to a module by the value of the grade earned in that module.

Formula of conversion of equivalent Percentage of OGPA:

$$\text{Percentage marks} = (\text{OGPA} - 0.5) \times 10$$

Illustrations of Calculation: - The illustration for the conversion of marks into grades in theory & practical, if any in individual module are as shown below as :-

Subject Code	Module Name	Secured Marks	Total Marks	Credits (C)	Earned Grade	Earned Grade point (G)	C*G	OGPA = $\frac{\sum C \cdot G}{\sum C}$
DAC01	Operating System Concepts	90	100	4	A	10	40	524/60=8.33
DAC02	OOPs with C++ Programming	85	100	4	B	9	36	
DAC03	Algorithm & Data structures	70	100	4	C	8	32	
DAC04	Advanced Software Development Methodologies	90	50	3	B	9	27	
DAC05	Advanced Web Programming	75	100	3	C	8	24	
DAC06	MEAN Stack	70	100	4	C	8	32	
DAC07	Database Technologies	80	100	3	B	9	27	
DAC08	Java Technologies-I (Core Java)	75	100	5	C	8	40	
DAC09	Java Technologies-II (Web Based Java)	85	100	6	B	9	54	
DAC10	Microsoft .NET	75	100	4	C	8	32	
DAC11	Effective Communication	70	50	3	C	8	24	
DAC12	Aptitude & General English	70	50	3	C	8	24	

DAC13	Project	190	200	10	A	10	100	
	Total	1150	1350	$\Sigma C = 60$			$\Sigma C * G = 524$	

Then conversion of equivalent Percentage of OGPA= $(OGP-0.5)*10 = (8.33-0.5)*10 = 78.3 \%$

4. Centralized Mid Course and Centralized Course End Examination (CMCE & CCEE):

After completion of the all module of PG-DAC, a written examination **Centralized Mid-Course Exam (CMCE) & Centralized Course End Exam (CCEE)** will be held, which will test the knowledge of the students about the theoretical concepts of each module and it is a compulsory part of the evaluation. Conducting CCEE involves performing duty with responsibility. A small mistake in the process may hamper the whole system. Everyone has to play their role in an effective manner. It is a joint effort work which has to be carried out in a combined way. Right from receiving the question paper from the national resource centres (NRCs) to sending the answer sheet for evaluation dealt with lot of responsibility.

4.1 Guidelines of CMCE and CCEE:

CMCE & CCEE will be conducted normally before the commencement of Project work of PG-DAC.

The written examination should be of 60 minutes duration. It should consist of objective questions. A typical objective type exam paper should contain the following types of questions: –

- Multiple choice
- Yes or No
- True or False

Objective questions are useful in testing the recognition and recall abilities of students. They also help in keeping the exam short and easier to evaluate.

For the pure objective type question papers, there will be 50 objective type questions with 4 maximum answer options having only one correct option. The value of each objective type question is of one mark only. There will not be any negative marks for the wrong answers given by the students.

4.2 Guidelines for setting Question Papers

While setting the question papers for theory Exam the following weightages should be assigned as per the difficulty level of the questions.

Levels	Requirements	Weightage
Level A - Easy	Requires elementary knowledge which may be obtained by attending all lectures and completion of mandatory lab assignments	25%
Level B - Intermediate	Requires thorough study of all course material, attendance at all lectures and completion of mandatory assignments	50%
Level C - difficult	Requires study and lab work beyond the prescribed course material and mandatory assignments	25%

4.3 Guidelines for generating theory questions

- 4.3.1 Question paper setter has to use sample paper format provided by C-DAC, ACTS Pune

- 4.3.2 Mention the module name without fail.
- 4.3.3 Language of the question should be easy to understand.
- 4.3.4 The answers must have relevant objective type choices and “only one” correct answer.
- 4.3.5 The questions must be prepared by referring appropriate books, reference books, reference material, and course material having good information.
- 4.3.6 The question must be created by the domain expert afresh and should not be copied directly from any book, website, existing previous question papers etc.
- 4.3.7 The question should be unique and should have not been published anywhere.
- 4.3.8 Please mention the source of the question wherever possible, as it may help us in referring the same for detailing if required.
- 4.3.9 The caliber of the question should suffice the growing need of competition.
- 4.3.10 The question paper should have questions covering the entire syllabus.
- 4.3.11 The questions have to be typed in MS Word with “Arial” having letter size 12 point. Do not bold any letter, word or sentence in any part of the question paper.
- 4.3.12 It is essential to give password to the word document and send/tell the password separately.
- 4.3.13 It is essential that utmost care is taken at your end to maintain the secrecy of the soft copy at all time.
- 4.3.14 An expert team will review all questions. The questions will be filtered as per following:
 - If the question is incomplete
 - If the answer of the question is wrong
 - If the question is not there in the syllabus
 - If the question appears more than once
 - If the question is too lengthy
 - If the question is irrelevant
 - If the options to the questions are irrelevant

4.4 Template for generation of Questions

Date:

Question generated by: Mr. /Ms.

Topic / Module:

Q. No.

Question:

<Text of the question>

Format Type 1 (If answer choices are lengthy, one or more than one line)

Answer Choices

A:

B:

C:

D:

Format Type 2 (If answer choices are short, two to three words)

Answer Choices

A:

B:

C:

D:

Format Type 3 (If answer choices are very short, one word only)

Answer Choices

A:

B:

C:

D:

Difficulty Level: Easy / Intermediate / Difficult

Reference: (Name of books)

(If question taken from book) (Mention name of the book, Author, ISBN)

Total Number of Questions Generated: _____

4.5 Template for Answer Key:

Module name			
Question No.	Answer Keys	Question No.	Answer Keys
1		21	
2		22	
3		23	
4		24	
5		25	
6		26	
7		27	
8		28	
9		29	
10		30	
11		31	
12		32	
13		33	
14		34	
15		35	
16		36	
17		37	
18		38	
19		39	
20		40	

4.6 Evaluation of Lab Exam should be based on the following criteria:

Criteria	Details	Max Marks	Marks Obtained
Algorithm	Documentation of Algorithm and Flowchart	2	
	Program adheres to the algorithm and flowchart	2	
Efficiency	Program is using only the required number of variables /conditions/loops/pointers etc and is optimal	2	
Correctness	The program produces desired output for a given input	20	
	The program handles all valid and Invalid inputs		
Software Engineering Principles	The program has meaning variable/function names	2	
	The program is commented properly (At least 20% of the code should be commented)	2	
Viva		10	
	Total Marks	40	

5. Evaluation of answer papers:

Use of OMR sheets will be useful for processing the result of multiple choice exams. OMR is an effective way to collect data, process for the result and also it takes less

time with greater accuracy in less effort. Centres need to follow the best way for scanning the OMR sheets, process the result and publish the result. Centres which are not using OMR can use OCR to conduct the exams and evaluate the students. Centre which are not using OMR or OCR can evaluate the students manually and process the result.

If a student requests for re-evaluation then the student has to pay Rs.150 /- for each module and it should be routed through training centre. The Re-evaluation fee should be paid to the respective C-DAC training Centres, in case of Authorized Training Centres associated to C-DAC, Pune, payment to be made in favour of "C-DAC, ACTS" and payable at Pune.(This is applicable only for theory exam)

6. Moderation:

Grace marks would be awarded as per the methodology below:

- Maximum of 4% of total course end theory exam marks can be awarded to a candidate.
- Maximum of 8% of individual course end module test marks (maximum marks) can be awarded per module.

S No.	Name of the course	Total Marks	Maximum grace marks for the course	Maximum Marks per module	Maximum grace marks per module
1	PG-DAC	400	16	40	3

On completion of the moderation exercise the revised marks should be updated in the marks database.

7. Re-examinations:

The following conditions will be applicable for the course end re-exam:

- Students who do not appear for an exam on the scheduled date will not have an automatic right to reexamination. Only those students who, in the opinion of the centre/course coordinator have a genuine reason for being absent may be allowed to appear for a re-exam.
- Students who have failed an exam may be allowed to appear for a re-exam.
- The re-exam should be conducted following the same process as the regular examination.
- Students, who failed/remained absent in the Course End Examination conducted by C-DAC, shall be allowed to appear in the re-examination only once.
- Students who remain absent or fail in the re-examination will not get any further chance for appearing for the re-examination. In such case the candidate can receive the Performance Statement and the certificate of participation without any grade.
- On evaluation of their answer sheets 20% of the marks obtained by the students will be deducted (towards de-rating for re-examination) for arriving at the final score, i.e. in order to clear the module test the student has to score a minimum of 48% marks instead of 40%.
- The fee for the re-exam is currently NIL.
- There will be no re-exam for the re-exam

8. Evaluation Guidelines for Effective Communication

There should be module end exam in Effective Communications (EC) for the benefit of the students. Total marks for evaluation will be 100 for Effective Communication/Presentation / Seminar / GD / Attendance. After evaluation marks need to be converted to grades as per the scale given below, which will be given in the mark sheet.

8.1 Evaluation method:

Sl. No.	Module Name	CCEE Marks	IA Marks	Total Marks
1	Communication skills	0	50	100
2	Presentation / Seminar/GD	0	40	
3	Attendance for EC sessions	0	10	

8.2 Internal evaluation Method for Effective Communication

The examination for this module must be conducted for at least that number of marks. However, the centre may conduct evaluation for a higher number of marks, in which case the marks will be scaled down. For example, if the examination for Presentation is conducted for 100 marks, the marks earned by the student will be scaled down to out of 40.

It is recommended to conduct some surprise tests for Effective Communications as Internal marks.

Centres can evaluate on Test/Assignment/Presentation/Seminar/Group discussion and may depend on subject & faculty.

8.3 Guidelines for Presentation/Seminar/Group Discussion Evaluation

Evaluation of Presentation, Seminar and Group Discussion needs to be carried out as per the following guidelines.

- a. Submission of their seminar topic along with the references.
- b. Two presentations each for 15 minutes on the seminar restricted to 15 – 20 slides followed by evaluation.
- c. Presentation/Seminar evaluation for 100 marks will be split up as follows:
 - i. Communication skills – 20
 - ii. Use of verbal and non verbal language – 10
 - iii. Presentation skills – 20
 - iv. Contents of the seminar/GD – 10
 - v. Contents Flow of Presentation – 10
 - vi. Depth of Knowledge in the topic – 10
 - vii. Viva Voce – 10
 - viii. Attendance – 10

9. Project Module:

- 9.1. Project work should start at the time of Software Application Development Tools & Techniques Module and database design should be complete at the time of Database Technologies Module.
- 9.2. At the end of first half student should be ready with all mandatory documents with database design and in the second half he will decide the language on which he wants to do the project.
- 9.3. Performance in the Project module will be awarded in grade. The Project grade will be mentioned separately on the Performance Statement and will have no effect on the overall grade obtained by a student.

9.4. Students may do industry-sponsored projects, but will be required to do the project work within the centre.

9.5. Evaluation of the Project module will take place asfollowing:

9.5.1. Internal evaluation will be take place at mid of the module

9.5.2. External evaluation will take place at the end of the module

Based on both evaluations, final grade will be awarded & communicated to C-DAC ACTS,Pune

10. Ensuring Security of Evaluation data/records:

10.1. Ensure that all data relating to evaluation of students is stored in a secure place that cannot be accessed by unauthorized personnel.

10.2. All question papers must be prepared and stored in a separate area specifically designated for the purpose.

10.3. Whenever any external faculty sets a question paper, ensures that he should follows the guidelines given by C-DAC ACTS Pune.

10.4. Ensure that only one copy of any question paper is prepared in physical (printed) form for review and revision.

10.5. When the question paper is finalized, print out one master copy and gets it signed by the paper setter, Reviewer and DRM Evaluation.

10.6. Prepare required number of photocopies of the question paper and store them in a safe and secure location before the exam.

10.7. The data relating to evaluation of students, such as soft copies of question papers and answer keys, student marks database and performance statements etc. must be kept in a separate domain/directory which is accessible only to authorized personnel. Ensure that the data is regularly backed up.

10.8. The question papers for the theory as well as the laboratory examinations at all the centres will be set by CDAC, ACTS Pune. The centres according to guidelines provided by C-DAC, ACTS Pune, will conduct the evaluation of the laboratory and assignments locally.

***Suggested Teaching Guidelines for
Operating System Concepts-PG-DAC August 2019***

Duration: 30 class room hours + 30 Lab hours (60hrs)

Objective: To introduce Advanced Operating System Concepts with Linux environment.

Prerequisites: Knowledge of Computer Fundamentals

Evaluation method: Theory exam– 40% weightage

Lab exam – 40% weightage

Internal exam– 20% weightage

List of Books / Other training material

Text Books:

1. Operating Systems Principles by Silberschatz, Galvin / Wiley
2. Unix Concepts and Applications Sumitabha Das /Tata Mcgraw Hill

Reference Books:

1. Modern operating Systems, Tanenbaum.
2. Principles of Operating Systems by Naresh Chauhan / Oxford University Press
3. Beginning Linux Programming 4th Edition, Neil Matthew, Richard Stones, wiley
4. Operating System : A Design-Oriented Approach 1st Edition, Charles Crowley /TMH

Session 1

Lecture:

- Overview of OS
- Introduction to Linux
- The Linux File System
 - Working with Files and Directories
- Getting Started to Linux
- Revision of basic Commands

Assignment –Lab:

Getting Acquainted with the Linux Environment

Use various commands in Linux system.

(ls, cp, mv, lpr, sort, grep, cat, tac, more, head, tail, man, whatis, whereis, locate, find, diff, file, rm, mkdir, rmdir, cd, pwd, ln and ln -s, gzip and gunzip, zip and unzip, tar and its variants, zcat, cal, bc and bc -l, banner date, time, wc, touch, echo, who, finger, w, whoami, who am i, alias, unalias, touch, push, pop, jobs, ps, etc.)

Session 2

Lecture:

Gaining confidence with Linux

- Access control list and chmod command
- chown and commands
- Network Commands like telnet, ftp, ssh, and sftp, finger

Suggested Teaching Guidelines for
Operating System Concepts-PG-DAC August 2019

- Use of secondary storage devices (Like: - Hard disk, Floppy, CDROM) in Linux environment and formatting of these devices.

Assignment –Lab:

Review Exercises

Session 3

Lecture: Linux shell

- What is shell?
- Different types of Linux shells
- Bourne Again Shell (BASH)
- Shell variables (environment and user defined)
- Shell files (.bashrc, .profile, .bash_profile, .bash_logout)
- Wild cards (* and ?)
- I/O redirection and tee command
- Shell meta characters
- Command line expansion

Assignment –Lab:

Review Exercises

Session 4

Lecture:

Linux shell programming

- Command line arguments
- Arithmetic in shell scripts
- Read and echo commands in shell scripts
- Taking decisions:
 - if-then-fi
 - if-then-else-fi
 - The test command (file tests, string tests)
 - Nested if-elses
 - The case control structure

Assignment –Lab:

Review Exercises

Session 5

Lecture:

Linux shell programming (Contd.)

- The loop control structure
 - The while, until and for loop structures
 - The break and continue statements
- Directory stacks manipulation

***Suggested Teaching Guidelines for
Operating System Concepts-PG-DAC August 2019***

- Job control, history and processes
- Built-in functions

Session 6: (Operating System Architecture)**Lecture:**

- Introduction to Operating Systems and Terminology
- Kernel Components and Non-kernel Components
- User-space vs Kernel-space and User-mode vs Kernel-mode
- Types of Kernel
- Interrupt Management and System calls

Session 7: (Process management, Scheduling Policies)**Lecture:**

- Process Management
- Process Scheduling / CPU Scheduling: Preemptive vs Non-preemptive
- Different types of Scheduling Policies/Algorithms – FCFS, RR, PRIO, FAIR-SHARE, EDF.
- Preemptive Kernel vs Non-preemptive Kernel
- Introduction to Threads - User-level vs Kernel-level(many-to-one vs one-to-one)
- Advantages of Multi-threading and some Real-world examples.

Session 8: (Inter-Process Communication)**Lecture:**

- Introduction to Inter-process Communication Mechanisms (IPCs) - for Processes and Threads
- IPCs - Synchronization vs Data-exchange
- IPCs - Semaphores and Mutex
- Message-queues and Shared-memory (Related to VM)
- Producer Consumer problem
- Dead-lock vs Starvation

Session 9: (Memory Management)**Lecture:**

- Memory Management
- Hardware Techniques available for Memory-management (Segmentation/Paging)

Session 10: (Virtual Memory Techniques)**Lecture:**

- Software Techniques built on top of H/W Techniques (Virtual-memory Techniques)
- Page Replacement Algorithms

Session 11: (File System Management)**Lecture:**

- File System Organization - Physical File System Organization Techniques
- (Use Linux File System - ext2/ext3 for Illustration)
- Virtual File System Manager

Suggested Teaching Guidelines for
Operating System Concepts-PG-DAC August 2019

- File System Management - File System Manager in the Kernel
- Disk structure and Disk Scheduling Algorithms

Session 12: (Process and signals)

Lecture

- Process Creation
- fork, waitpid and exec System call
- Basic Concepts of the signals
- Generating and Handling Signals

Assignment – Lab:

- Assignment based on Process creation
- Assignments Based on the Signals

Session 13: (Inter-Process Communication, Semaphores)

Lecture

- Synchronization with Semaphore
- Shared Memory Concept and Implementation
- Message queues

Assignment –Lab:

- Assignment based on implementation Semaphore
- Assignment based on the implementation of the shared memory

Session 14: (File handling)

Lecture

- Working with Files and Directories
- Hard and soft links
- Pipes and FIFOs implementation

Assignment –Lab:

- Assignment based on the implementation of the pipes, fifos

Session 15: (POSIX Threads and Mutex)

Lecture

- Thread Management
- Thread Safety
- Thread Synchronization with Mutex

Assignment –Lab:

- Assignment based on the Implementation of the threads and Mutex

Suggested Teaching Guidelines for
OOPs with C++ Programming – PG-DAC August 2019

Duration: 30 class room hours + 30 lab hours (60hrs)

Objective: To reinforce knowledge of Object Oriented Concepts and C++ Programming.

Prerequisites: Knowledge of C Programming

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Test Book:

1. C++ Primer Plus by Stephen Prata / Pearson

Reference:

1. Thinking in C++ by Bruce Eckel
2. The C programming Language by Kernighan and Retchie
3. Advanced Unix Programming by Rochkind
4. The C++ Programming Language, Bjarne Stroustrup;
5. Object-oriented Analysis And Design Using Umlan Introduction To Unified Process And Design Patterns 1st Edition by Mahesh P. Matha / PHI

Session 1 & 2 & 3: (Linux Command and Revision of C Programming)

Lecture

- Linux introduction
- Basic Linux commands required for writing and executing C++ programs
- Revision of C Programming
 - Pointers
 - Functions (Call by value and reference)
 - Recursion
 - Arrays using Pointers
 - Structures
 - Union
 - Enumeration and Typedef
 - File handling

Assignment – Lab:

- Practice of Linux commands
- Writing different C programs based on functions, structures, file and executing them

Session 4: (OOPs Concepts)

Lecture

- Difference between C and C++
- Discussion on Object oriented concepts
- Classes and Objects, Access Specifiers, Overloading, Inheritance, Polymorphism
- Constructors and Destructors

Suggested Teaching Guidelines for
OOPs with C++ Programming – PG-DAC August 2019

- Namespaces

Assignment – Lab:

- Write a Student class and use it in your program. Store the data of ten students and display the sorted data according to their roll numbers, date of births, and total marks.

Session 5: Beginning with C++**Lecture**

- Introduction of advanced CPP concepts and Feature of C++ 17th edition
- C++ Tokens
- Initialization
- C++ Operators
- Static Members
- Constant Members
- Expressions
- Control Structure

Assignment – Lab:

- Implement all C++ operators
- Declare members and implement in your programs.
- Implement all control structures through your program
- Implement a program which accepts command line arguments from main function.

Session 6: Functions in C++

- Different forms of functions
- Function prototyping
- Call by Reference
- Inline Functions
- Math library functions etc.

Assignment – Lab:

- Implement functions through your program
- Declare function and call it by reference and note the observations
- Implement Inline functions in your program

Session 7 & 8: Memory Management and pointers**Lecture**

- Introduction to memory management in C++
- Pointers in C++
- Arrays using pointers
- Enumeration
- Typedef
- Using New operator
- Class pointer
- this pointer
- Comparison of new over malloc, calloc and realloc etc.
- Memory freeing using Delete operator

Assignment – Lab:

Suggested Teaching Guidelines for
OOPs with C++ Programming – PG-DAC August 2019

- Assignments using pointers, arrays of pointers
- Assignments on passing pointers in functions.
- Using pointers write your own functions for the following;
 - String comparison
 - String concatenate
 - String copy
 - String length

Note: Do not include <string.h> in your program and Implement delete operator in your program

Session 9: Constructor and Destructor**Lecture**

- Constructors
- Parameterized constructors
- Multiple constructors in class
- Dynamic initialization of objects
- Copy Constructors
- Destructors

Assignment – Lab:

- Implement constructor and destructors through your program
- Write program to implement inner class in C++

Session 10: Inheritance – extending class**Lecture**

- Introduction
- Types of inheritance
- Single inheritance
- Multiple inheritance
- Multilevel inheritance
- Hierarchical inheritance
- Hybrid inheritance etc
- Virtual base class
- Constructors in derived class

Assignment – Lab:

- Design a hierarchy of computer printers. Use multiple inheritance in your hierarchy. Also use friend functions and classes in your program.

Session 11: Polymorphism**Lecture**

- Types of Polymorphism
- Overloading functions
- Overloading Operators
- Friend Functions
- Constant functions

Assignment – Lab:

Suggested Teaching Guidelines for
OOPs with C++ Programming – PG-DAC August 2019

- Write Date and Time classes that allows you to add, subtract, read and print simple dates in dd/mm/yyyy and time in hh:mm:ss forms. Use function overloading in your program.
- Assignments to overload =, ==, +, ++, --, <<, >> and [] operators.

Session 12: Virtual Functions and Abstract Class**Lecture**

- Run Time Polymorphism
- Virtual Functions and Pure virtual functions
- dynamic_cast, static_cast, const_cast, reinterpret_cast
- Interfaces
- Abstract class

Assignment – Lab:

- Implement Abstract classes in your program
- Using virtual and pure virtual functions implement hierarchy of computer printers.
- Implement diamond problem with real life example

Session 13: Exception Handling**Lecture**

- Exception Handling Introduction
- Exception handling – throwing, catching, re-throwing an exceptions
- Specifying exceptions etc.

Assignment – Lab:

- Implement exceptions and do proper management through your program
- Implement Custom exception class

Session 14: Managing Console I/O Operations**Lecture**

- Introduction
- C++ streams
- C++ stream classes
- Unformatted I/O operations
- Formatted I/O operations
- Managing output with manipulators

Assignment – Lab:

- Implement console I/O operations through your program

Session 15: File Handling in C++**Lecture**

- Definition of file
- File handling in C++

Suggested Teaching Guidelines for
OOPs with C++ Programming – PG-DAC August 2019

- Doing read, write operation in files

Assignment – Lab:

- Assignments on files doing different operations.

Suggested Teaching Guidelines for Algorithm and Data Structures – PG-DAC August 2019

Duration: 34 class room hours + 36 lab hours (70hrs)

Objective: To reinforce knowledge of Problem solving techniques, Data Structure Concepts and analysis of different Algorithm.

Prerequisites: Knowledge of Programming in C/C++ with Object Oriented concepts

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Test Book:

1. Fundamentals of Data Structures in C++ by Horowitz, Sahani & Mehata / Orient Longman

Reference:

1. Problem Solving: Best Strategies to Decision Making, Critical Thinking and Positive Thinking by Thomas Richards / Kindle Edition
2. Data Structures, Algorithms and Applications in C++ by Sartaj Sahni
3. Object-oriented Analysis And Design Using Umlan Introduction To Unified Process And Design Patterns 1st Edition by [Mahesh P. Matha](#) / PHI
4. Introduction to Algorithms by Cormen, Leiserson, Rivest and Stein
5. Data Structures Via C++ Objects by Evolution by A Michael Berman / Oxford University Press
6. Design and Analysis of Algorithm by S Sridhar / Oxford University Press
7. Fundamentals of Computer Algorithms 2nd Edition by Sanguthevar Rajasekaran, Sartaj Sahni, Ellis Horowitz / Orient Longman
8. Introduction to Algorithms by Al. Cormen / PHI Learning
9. The Algorithm Design Manual by Steven S. Skiena / Springer
10. Algorithmic Puzzles by Anany Levitin, Maria Levitin / Oxford University Press

Session 1 & 2: Introduction to STL

Lecture

- Object Design and Templates
- STL (Standard Type Libraries)

Assignment – Lab:

- Find the number of students who are passes or failed using MAP.
- Find the prime numbers from 2 to n using sieves algorithm, use SET

Session 3 & 4: Problem Solving & Computational Thinking

Lecture

- Define the problem
 - Identify the problem

**Suggested Teaching Guidelines for
Algorithm and Data Structures – PG-DAC August 2019**

- Introduction to Problem Solving
 - Problem solving basics
 - Defining creativity v/s innovation
- Find Creative Solutions using creativity tools
 - Effective problem solving approaches
 - Critical thinking and information analysis
 - Brainstorming, Reverse Brainstorming, Imagineering, Mind Mapping,
 - Six Thinking Hats: A Tool to Strengthen Critical Thinking, Collaboration, Communication, and Creativity Skills
 - Analyzing the situation, Gathering information, Identifying solution criteria
 - Decision Making Methods
 - Charts and Diagrams
 - Applying outcome-based thinking
 -
- Evaluate and Select solution
 - Pro's and Con's, Force field analysis, Feasibility/Capability Analysis,
 - Decision analysis, evaluating problems
 - Choosing among alternatives
 - Qualitative analysis, discussing qualitative analysis techniques
 - Establishing objectives
 - Assigning weight to objectives in order to make the best decision
 - Creating a satisfaction scale to choose between alternatives
- Implementing Decisions
 - Create an action plan
 - Break solution into action steps
 - Prioritize actions and assign roles (setting priorities for taking action)
 - Follow-up at milestones

Assignment – Lab:

- Faculties need to assign different problems, mostly real world problems
- Students (by team wise, there are two students in a team) need to analyze as per the techniques learned
- Students need to solve by the thinking approaches learned.
- Based on the above problems students need to select as per the selection criteria learned
- They need to implement the selected solution and need to do the documentations.

Session 5 & 6: Algorithm design

Lecture

- How to write efficient Algorithm
- Introduction to algorithm design techniques
- Algorithm Design techniques
- Analysis of an Algorithm
 - Asymptotic analysis
 - Algorithm analysis
- Analysis of different type of Algorithms

Suggested Teaching Guidelines for
Algorithm and Data Structures – PG-DAC August 2019

- Divide and Conquer Algorithm
- Greedy Algorithm
- Dynamic Programming Algorithm
- Brute force Algorithm
- Backtracking algorithms
- Branch-and-bound algorithms
- Stochastic algorithms
- Complexity
 - Complexity Analysis
 - Space complexity of algorithm
 - Time complexity of algorithm
- Case study on Algorithm Design techniques
- Application of Data structures

Assignment – Read:

- Study on different Algorithms
- Compare different Algorithms previously programmed and do the analysis

Session 7 & 8: Algorithm & Data Structures

Lecture:

- Introductory Concepts
- Algorithm Constructs
- OO design: Abstract Data Types (ADTs)
- Basic Data Structures
 - Arrays
 - Stacks
 - Queues
 - Circular Queues
 - Priority Queues
 - Deques

Assignment – Lab:

- Implement Stack through Array
- Implement C-Stack, C2-Stack and CN-Stack in same memory block.
- Implement Queues with inserting element at different location (First, Last and at specific location)
- Implement circular queue, Priority Queues and Dqueue
- Implement program to convert infix expression into postfix expression & evaluate postfix expression.

Session 9 & 10: Linked List Data Structures

Lecture

- Linked lists
 - Single Linked Lists
 - Double Linked Lists
 - Circular Linked Lists

Suggested Teaching Guidelines for
Algorithm and Data Structures – PG-DAC August 2019

- Node-based storage with arrays

Assignment – Lab:

- Implement circular queue using linked list
- Design an iterator using circular linked list

Session 11 & 12: Trees & Applications

Lecture

- Introduction to trees
- Trees and Terminology
- Tree traversals
- Ordered trees
- Binary trees
- Complete binary trees
- Search trees
- Binary search trees
- Introduction to self balancing tree & variants

Assignment – Lab:

- Write a program to implement a binary search tree and the following operations on it:
 - Create()
 - InsertNode()
 - Tree traversals (Inorder(), Preorder(), Postorder())
 - deleteNode()
- Design a threaded binary tree and implement the orders.

Session 13 & 14: Searching & Sorting algorithms

Lecture

- Objectives of Searching
 - The Sequential Search
 - Analysis of Sequential Search
 - The Binary Search
- Analysis of Binary Search
- Introduction to sorting
 - Selection sort
 - Insertion sort
 - Bubble sort
 - Heap sort
 - Merge sort
 - Quick sort
- Analysis of sorting algorithms

Assignment – Lab:

- Writing program to search an item through sequential search technique.
- Implement to find an item in a list through binary search
- Implement sorting algorithm for selection sort, Bubble sort, heap sort and quick sort

Suggested Teaching Guidelines for
Algorithm and Data Structures – PG-DAC August 2019

- Write a program to merge two sorted linked lists

Session 15: Hash functions and hash tables**Lecture**

- Hashing & Introduction to hash tables
- Hash functions
- Mapping down to $0 \dots M - 1$
- Chained hash tables
- Scatter tables
- Open addressing
- Linear probing
- Quadratic probing
- Double hashing
- Poisson distribution
- Collision Resolution
- Analysis of Hashing

Assignment – Lab:

- Implement hashing techniques in different programs solved earlier
- Implement collision and solution to it on any previous solved problem
- Write a program to implement Hash table

Session 16 & 17: Graph & Applications**Lecture**

- Introduction to graph theory
- Graph Terminology
- Different types of Graphs
- Representation of Graphs
 - Connectedness, Single source un-weighted path length, identifying bipartite graphs
 - Graph Traversal Algorithms (Breadth First Search, Depth First Search)
 - Single-source shortest path algorithms, Dijkstra's algorithm, A* search algorithm, Bellman-Ford algorithm
 - All-pairs shortest path, Floyd-Warshall algorithm, Johnson's algorithm
 - Maximum flow algorithms, Ford-Fulkerson algorithms
- Spanning Trees
 - Minimum spanning tree algorithms, Prim's algorithm, Kruskal's algorithm

Assignment – Lab:

- Implement a graph using adjacency links and traverse using Depth First Search.
- Write a program using STL to implement Dijkstra's Shortest Path Algorithm.

Suggested Teaching Guidelines for
Advanced Software Development Methodologies – PG-DAC August 2019

Duration: 46 class room hours + 44 lab hours (90hrs)

Objective: To reinforce knowledge of Advanced Software development Methodologies.

Prerequisites: Fundamentals of Computers

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Test Book:

1. Software Engineering by Chandramouli

Reference:

1. Software engineering by Ian Sommerville
2. Agile Project Management with Scrum by Ken Schwaber
3. The Mythical Man-Month: Essays on Software Engineering by Frederick P. Brooks Jr.
4. User Stories Applied: For Agile Software Development 2016 by Mike Cohn
5. Continuous Delivery, Integration, and Deployment with DevOps: Dive into the core DevOps strategies by Sricharan Vadapalli

Software Engineering

Session 1

Lecture

- Introduction to software engineering
- Importance of Software engineering
- Software Development Life Cycle

Assignment –Lab

- Prepare software requirement specification for web application

Session 2 ,3

- Design and Architectural Engineering
 - a. Characteristics of Good Design
 - b. Function Oriented vs Object Oriented System
 - c. Modularity, Cohesion, Coupling, Layering
 - d. Design Models
 - e. UML
- Object Oriented Analysis and Design

Session 4

Lecture

- Introduction to Agile development model
- Agile development components
- Benefits of Agile
- Introduction to Atlassian Jira
 - Add Project

Suggested Teaching Guidelines for
Advanced Software Development Methodologies – PG-DAC August 2019

- Add Tasks and sub-tasks
- Create sprints with tasks

Session 5

Lecture

- Introduction to different tools used for agile web development
- Case study of developing web application using agile methodology

Assignment –Lab

- Create different sprints in Atlassian Jira for different features

DevOps

Session 1

Lecture

- Introduction to DevOps
- DevOps ecosystem
- DevOps phases

Lab

- Read about DevOps Tools

Session 2

Lecture

- Introduction to Microservices
- Microservice Architecture
- Fragmentation of business requirement
- Deployment pattern
- API gateway
- Service Discovery
- Database Management for Microservices

Lab

- Create micro-services

Session 3 and 4

Lecture

- Introduction to containerisation
- Introduction to docker
- Creating docker images using Dockerfile
- Container life cycle

Lab

- Install and configure docker
- Create docker image using Dockerfile
- Start docker container
- Connect to docker container
- Copy the website code to the container
- Use docker management commands to
 - List the images
 - List the containers
 - Start and stop container
 - Remove container and image

Suggested Teaching Guidelines for
Advanced Software Development Methodologies – PG-DAC August 2019

Session 5 and 6

Lecture

- Introduction to YAML
- Introduction to Docker Swarm and Docker Stack
- Introduction to Kubernetes
- Creating Kubernetes cluster
- Creating service in Kubernetes
- Deploying an application using dashboard
- Introduction to Istio Service Mesh

Lab

- Configure Kubernetes
- Configure Kubernetes Dashboard
- Setup a Kubernetes cluster
- Access application using Kubernetes service
- Deploy the website using Dashboard

Session 7

Lecture

- Introduction to delivery pipeline
- Introduction to Jenkins
- Jenkins management
- Adding slave node to Jenkins
- Building a delivery pipeline
- Selenium integration with Jenkins
-

Lab

- Install and configure Jenkins
- Build a pipeline job using Jenkins
- Create a maven project for TestNG
- Add TestNG test suite in the project
- Integrate it with Jenkins

Git

Session 1

Lecture

- Developing an application in a team
- Issues developers face when working in a team
- Introduction to code versioning system
- History of code versioning system
 - Different tools available for versioning
 - Software development workflow

Session 2

Lecture

- Introduction to git
- Introduction git repository and git structure
- Adding code to git

Suggested Teaching Guidelines for
Advanced Software Development Methodologies – PG-DAC August 2019

- Creating and merging different git branches

Assignment –Lab

- Create a local git repository
- Commit the initial code
- Update the code
- Use git commands to
 - Get the updated files
 - List the changes
 - Create branch
 - Merge branch

Session 3

Lecture

- Introduction to GitHub
- Pushing and pulling the code to remote repository
- Creating pull requests
- Reviewing code and merging the branches on GitHub

Assignment –Lab

- Create a repository on GitHub
- Push the local changes to GitHub
- Pull the code from GitHub
- Create and checkout a branch
- Add a new feature, commit and push changes to GitHub
- Create a pull request
- Review the changes on GitHub and merge the branch into the main branch

Testing

Session 1

Lecture

- Introduction to software testing
- Why testing code is important
- Verification and validation
- Quality Assurance vs Quality Control vs Testing
- Principles of software testing

Assignment –Lab

- Read more testing concepts used in the industry

Session 2

Lecture

- Introduction to STLC and V Model
- Types of testing: manual and automation
- Tools used for automation testing
- Introduction to testing methods: white-box, black-box and grey-box
- Introduction to functional testing: (* students are supposed to learn the concepts)
- Introduction to non-functional testing: (* students are supposed to learn the concepts)

Assignment –Lab

- Create a test plan for project
- Document the use cases
- Create test case document for different sprints (designed in SE)

Suggested Teaching Guidelines for
Advanced Software Development Methodologies – PG-DAC August 2019

Session 3 and 4

Lecture

- Introduction to Selenium (use Eclipse IDE)
- Load web driver
- Create selenese commands
 - locators: by ID, name, class, tag name, XPath
- Add interactions
 - text box
 - radio button selection
 - check box selection
 - drop down item selection
 - keyboard actions
 - mouse actions
 - multi select

Assignment –Lab

- Download and configure Selenium
- Create a test suit
- Add commands and interactions

Session 5

Lecture

- Introduction to TestNG
- Introduction to TestNG annotations
 - BeforeSuite, AfterSuite, BeforeClass, AfterClass, BeforeTest, AfterTest
 - BeforeGroups, AfterGroups, BeforeMethod, AfterMethod
 - DataProvider, Factory, Parameters, Test
- HTML test result reporting

Assignment –Lab

- Download and configure TestNG
- Create test suit
- Add TestNG annotations

Cloud

Session 1

Lecture

- Introduction to Cloud
- Introduction to Virtualization
- Virtualization types: type1, type2
- Containerisation
- Cloud Computing, Cloud SPI Model, Cloud Computing Types (Public, Private and Hybrid), Cloud Security (SLA and IAM).
- Virtualization, Hardware Virtualization, Para-Virtualization, Cloning, Snapshot and Template
- Containerization, Operating System Virtualization

Assignment –Lab

- Create and configure VM using VBox
- Deploy code on VM

Suggested Teaching Guidelines for
Advanced Software Development Methodologies – PG-DAC August 2019

Session 2

Lecture

- Cloud architecture
- Service models: IaaS, PaaS, SaaS
- Deployment models: Private, Public, Hybrid
- Services provided by Cloud (Compute, Database, Developer Tools, Storage, Media, Mobile, Web, Security, Integration etc.)
- Cloud development best practices

Assignment –Lab

- Exploring various services provided by cloud providers like App Services, Web apps, API Apps, Search, Database Servers on VMs, VM Scale Sets, Bot Services and other cloud applications.

Session 3

Lecture

- Introduction to AWS
- Services provided by AWS: EC2, Lambda, S3

Assignment –Lab

- Create AWS EC2 instance
- Create AWS Lambda
- Create AWS S3 bucket
- Create AWS VPC

Suggested Teaching Guidelines for
Advanced Web Programming–PG-DAC August 2019

Duration: 26 class room hours + 24 lab hours (50hrs)

Objective: To introduce the students the HTML, JavaScript, PHP, XML, JSON, Ajax & Drupal and practical relevance of all these technologies.

Prerequisites: Knowledge of Internet

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam–20% weightage

List of books/other training Material:

Text Book:

1. HTML5 covers CSS3, Javascript, XML, XHTML, Ajax, PHP and jQuery- Black Book/Dreamtech Press

Reference:

1. Internet & World Wide Web How to Program by Deitel, Deitel & Nieto; Pearson Education
2. XML - How to Program - Deitel and Deitel - Prentice Hall
3. Ajax In Action by Dave Crane, Eric Pascarello, Dreamtech Press
4. Web Application Security: A Beginner's Guide, Bryan Sullivan, TMH
5. Joomla: 24-Hour Trainer (With CD) by Jen Kramer/ Wiley India Pvt Ltd (2011)
6. Node.js, MongoDB and AngularJS Web Development by Dayley Brad Dayley / Pearson
7. Pro AngularJS by Klaus Freeman Adam Freeman Freeman / Apress
8. AngularJS Services by Paperback / Shroff Publishers & Distributors
9. Angularjs in Action by Lukas Ruebbelke / Dreamtech Press
10. JavaScript: The Good Parts 1st Edition by Douglas Crockford / O'Reilly
11. JavaScript Step by Step 2nd Edition by Steve Suehring/ PHI Learning
12. HTML5, JavaScript and jQuery by Dane Cameron/Wiley India
13. Sams Teach Yourself JavaScript in 24 Hours (English) 1st Edition Phil Ballard, Michael Moncur/Pearson Education
14. High Performance JavaScript 1st Edition by ZAKAS / O'Reilly

Session 1: Architecture of Web

- Brief history of the Internet
- How the internet works?
- Internet protocol; HTTP protocol; Domain names; Domain Names Service Servers
- HTTP Protocols
 - Difference HTTP 1.0, HTTP 1.1 & HTTP 2.0 and ES7 standard
 - Stateless nature of the protocol
 - Methods (GET, POST, HEAD, PUT, DELETE)
 - HTTP session
 - Status codes
 - Persistent connections
 - HTTPS
- Architecture of the Web

Advanced Web Programming–PG-DAC August 2019

- Web servers; IIS; Apache server

Assignment Lab:

- Exploring different BrowserPlugins
 - Firefox: Firebug, Web Developer Toolbar
 - Chrome: Developer Tools (no plugin necessary)
 - Safari: Web Inspector (no plugin necessary)
- Exploring different TextEditor
 - Windows: Notepad++
 - Linux: Gedit + Plugins
- Exploring about FTP
 - FileZilla
 - Another option: SSH

Session 2 : HTML

- Introduction to HTML5
- Basic HTML
- Introduction to basic HTML Aligning the Headings
 - Anchor Tag
 - Paragraph
 - Images and Pictures
 - Tables
- Framesets
 - New features in HTML5
 - New element
 - New attribute
 - Link relations
 - Microdata
 - ARIA accessibility
 - Multimedia
 - 2D and 3D drawing Support
- HTML Forms
- HTML Controls
 - INPUT
 - Text Area
 - Radio Button
 - Check Box
 - Dropdown
 - List box
 - Submit button
 - Set button
 - Button
- List and Tables
- Document Object Model (DOM)
- Elements
- Events

Advanced Web Programming–PG-DAC August 2019

- HTML 5
 - Elements
 - Objects
 - Events
 - Canvas
 - HTML5 Validation
 - Audio & Video Support
 - Geo-location Support

Assignment – Lab:

- Create a webpage with HTML describing your department. Use paragraph and list tags. Apply various colors to suitably distinguish key words. Also apply font styling like italics, underline and two other fonts to words you find appropriate. Also use header tags. Create links on the words e.g. “Wi-Fi” and “LAN” to link them to Wikipedia pages. Insert an image and create a link such that clicking on image takes user to other page. Change the background color of the page. At the bottom create a link to take user to the top of the page.
- Create your bio-data in an HTML Page. Divide it into following sections – Personal information, Family Background, Academic Qualifications, and Experience. Now divide a HTML page into three frames as upper, left and right (main) frames. Write a Heading in the upper frame and put the bio-data sections links in the left frame and on click the section links the respective detail information should be displayed into the right main frame.
- Create a HTML Strict document. Use HTML comments, the title element, and meta elements to provide information about a document. Use headings, paragraphs, lists (definition, bullet, and numbered), blockquote, address, and preformatted block elements. Use line breaks, hypertext links, images, strong, emphasis, code, superscript, and subscript inline elements. Use id's and internal anchor links to jump within a page
 - For Lab, there are two members in a team, one team member (the driver) has control of the keyboard/mouse and actively implements the program
 - The other team member (the navigator) continuously observes the work of the driver to identify tactical defects (such as syntactic and spelling errors, etc.) and also thinks strategically about the direction of the work. Please check as follows for lab work:
 - Create a structured XHTML 1.0 Strict document with html, head, and body elements
 - Use HTML comments, title, and meta elements to a web document
 - Switch pair programming roles
 - Use headings to create a document outline
 - Switch pair programming roles
 - Format text with block tags
 - Format text with inline and anchor tags
 - Screen Snapshots

Session 3: Working with CSS

- Introduction to CSS
- Styling HTML with CSS
- Structuring pages with CSS
- Cascading style sheet
- Inline Styling (Inline CSS)

Advanced Web Programming–PG-DAC August 2019

- External Styling (ExternalCSS)
- CSS Fonts
- The CSS Box Model
- The id Attribute
- The class Attribute
- HTML Style Tags
- Linking a style to an HTML document
- Internal style sheet
- Multiple styles

Assignment Lab:

- Considering the lab assignments of HTML
 - Write a CSS rule that changes the color of all the elements with attribute CLASS = "Green-Move" to green and shift them down 25 pixels and right 15 pixels.
 - Set the page body's margins to 10 pixels on all sides with no padding. Use a Verdana sans-serif font and set the default font-size to 95%
 - Set all divisions to display with a 1 pixel, solid, black border (to help with debugging)
 - In the Header Area
 - Set its heading 1's text to display centered, in a 'Courier New' monospace font, and a font size of .8 em
 - Set its heading 2's text to display centered with a font size of 1.5 em
 - In the Navigation Area
 - Make its width 25% of the screen size and float it left
 - Style its unordered list of anchor elements to look like a menu
 - Do not add the background image or onfocus attribute (final optional steps) in tutorial
 - use a background color of red and a text color of white
 - center the text
 - In the Main Content Area
 - Set the right and left margins to 26% of the screen size
 - Make its heading 2's text centered with a font size of 1.2 em
 - For Printing, write the style rules to
 - Set display to none for the navigation and advertisement areas so that only the header and main content are printed.
 - You do not need to specify style rules for the header and main content elements.
 - The browser should use the default settings when printing your Web page
- Create a form to submit a resume
- Using in line CSS. Write all the above styling in CSS in different file (.css) and link it to your webpage such that changes made in CSS file are immediately reflected on the page. Group paragraphs into single class and add styling information to the class in CSS. Add few form elements such as radio buttons, check boxes and password field. Add a submit button at last.
- Apply inline CSS to change colors of certain text portion, bold, underline and italics certain words in your HTML web page. Also change background color of each paragraph
- Create a simple form to submit user input like his name, age, address and favorite subject, movie and singer.

Suggested Teaching Guidelines for
Advanced Web Programming–PG-DAC August 2019

Session 4: PHP

- Introduction to PHP
- Basic rule of PHP
- PHP in action
- Working with text, variable and numbers
- Making decisions and repeating yourself
- Arrays
 - Working with Arrays
 - Looping through array
 - Sorting arrays
- Functions

Assignment – Lab:

- Write a simple program in PHP.
- Write a program in PHP that uses the increment operator (++) and combined multiplication (*=) operator to print out the numbers from 1 to 5 and powers of 2 from $2(2^1)$ to $32(2^5)$

Session 5: PHP

- Making web forms
 - Form processing with functions
 - Validating data
 - Display default value
- Working with cookies and Sessions
 - Login and User Identification
 - Parsing, display date and times

Assignment – Lab:

- Write a simple program to remembering user with cookies and Sessions

Session 6: PHP

- Storing information with databases
 - Connection to database
 - Create a table
 - Inserting and retrieving data from database
 - Inserting and retrieving form data safely
 - MySQL with out PEAR DB
- XML
 - Generating and Parsing an XML Document
 - Advanced XML processing
- Debugging
 - Fixing parsing error and database error

Assignment – Lab:

- Write a program to implement various databases queries.

Advanced Web Programming–PG-DAC August 2019

Session 7: PHP

Lecture:

- Working with files
 - File permissions
 - Reading and writing files
 - Working with CSV Files
 - Checking for errors
- Command line PHP
- Running Shell command
- IMAP, POP3 and NNTP
- Graphics, PDF
- Sending and receiving mails

Assignment – Lab:

- Write a program to implement various file operation.
- Write a program to implement Command line PHP
- Write a program in PHP for Sending and receiving mails

Session 8: Working with XML

- Introduction to XML.
- Revision of HTML and Client-Server Architecture
- Reason for XML
- XML and the browser
- XML Validation
- XML as the mode of data transfer
- XML Parsers
- XML SAX Parser
- A basic XML Document
- Parts of the XML Document
- XML Schema
- Namespaces
- The Root Element
- Tags and Elements
- Attributes and Values
- CDATA and PCDATA
- XML Tree Structure
- XML style language

Assignment – Lab:

- Create some basic XML documents and check them out in the different browsers for validity.

Session 9: DTD

- XML DTD
- Creating Document Type Declarations.
- Creating Document Type Definition.
- Internal and External DTD's
- Associating DTD's with XML documents. (The XML Declaration and DOCTYPE declaration.)

Advanced Web Programming–PG-DAC August 2019

- Validating documents against a DTD
- Internal and External General Entities.
- Element Type (ELEMENT) Declarations.
- Attribute (ATTLIST) Declarations.
- Using INCLUDE and IGNORE.

Assignment – Lab:

- Create a XML document and write DTD for it
- Create a XML document and validate it

Session 10: W3C DOM and ~~Data Binding~~

- XML DOM
- Introduction to DOM
- The DOM Document Object and its properties.
- Getting Elements byname
- Editing attributes and values.
- Handling Events on client side.
- ~~Data Binding in Internet browser.~~
- ~~Using Data Source Objects (MSHTML, TDC, XML DSO, Data Islands).~~
- XML and XSLT
- XML DOM parsers
- XML WSDL
- RSS Feed
- ~~Introduction to Web security~~
- ~~SQL Injection~~
- ~~Cross-Site Scripting (XSS)~~
- ~~Angular java-script Services for Ajax and Promises~~
- ~~Security standards (OWASP)~~

Assignment 1 – Lab:

- Create an HTM page with two frames using XML document
 - The top frame should have input text boxes for search criteria. The textboxes are:
Marks greater than, Marks less than:
 - The lower frame will contain a grid, which will load the results of the above query.

Assignment 2 – Lab:

- Create an HTM page representing a departmental store bill:

Header info: Name of customer
 Date
 Bill No.
 Bill Details

Consider a purchase of the following items:

Name	Qty	Rate	Amount
Apples	1	24	
Lux Soap	4	15	
Room freshner	1	200	

Advanced Web Programming–PG-DAC August 2019**Assignment 3 – Lab:**

- Prepare a single XML representing the above data. Use data binding to display it. Calculate the total amount thru JavaScript and assign it to the total amount label at bottom.

Session 11 & 12: Working with JSON

- Introduction to the JavaScript Object Notation (JSON)
- Need of JSON
- JSON Syntax Rules
- JSON Data - a Name and a Value,
- JSON Objects,
- JSON Arrays,
- JSON Uses JavaScript Syntax,
- JSON Files
- JSON & Security Concerns, Cross Site Request Forgery (CSRF), Injection Attacks,
- JS XMLHttpRequest functions
- JavaScript XMLHttpRequest & Web APIs
- JSON & Client Side Frameworks
- JSON & Server Side Frameworks
- Replacing XML with JSON
- JSON parsing
- Introduction to Ajax
- Ajax Framework
- Ajax using HTML, CSS, JavaScript and DOM
- Ajax Architecture
- Creating a windows web services
- XMLHttpRequest
- Web services and Ajax
- JPSpan
- DWR
- AJAX using JSON and jQuery

Assignment Lab:

- Replacement of code used for designing of a Login and Registration pages
- Reads JSON data from a web server running PHP and MySQL
- Create an array “myArray” of objects. Use an array literal to declare an array of objects. Give each object two properties: display and url. Create a function to display the array.
- Create a function and put the function call in an external file. Using these display the objects of array and do some modifications in the array objects.
- Design a Login and Registration system using Ajax

Session 13: Responsive Web Design

- Introduction of UI Scripting
- The Best Experience for All Users
 - Desktop

Advanced Web Programming–PG-DAC August 2019

- Tablet
- Mobile
- Bootstrap
 - Overview of Bootstrap
 - Need to use Bootstrap
 - Bootstrap Grid System, Grid Classes, Basic Structure of a Bootstrap Grid
 - Typography
 - Tables, Images, Jumbotron, Wells, Alerts, Buttons, Button Groups, Badges/Labels, Progress Bars, Pagination, List Groups, Panels, Dropdowns, Collapse, Tabs/Pills, Navbar, Forms, Inputs
 - Bootstrap Grids, Grid System, Stacked/Horizontal
 - Bootstrap Themes, Templates

Assignment Lab:

- Design of a Login and Registration pages using above Bootstrap tools
- Testing of web pages designed in HTML5 and modifying the UI as per the above features.

Suggested Teaching Guidelines for MEAN Stack –PG-DAC August 2019

Duration: 30 class room hours + 30 lab hours (60hrs)

Objective: : To introduce the student to scripts and Facilitate Hands On Experience on Scripting language.

Prerequisites: Knowledge of Internet

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of books / other training Material:

Text Book:

1. HTML5 covers CSS3 , Java script, XML , XHTML, Ajax, PHP and jQuery- Black Book/Dreamtech Press

Reference:

1. Node.js, MongoDB and AngularJS Web Development by Dayley Brad Dayley / Pearson
2. Pro AngularJS by Klaus Freeman Adam Freeman Freeman / Apress
3. AngularJS Services by Paperback / Shroff Publishers & Distributors
4. Angularjs in Action by Lukas Ruebbelke / Dreamtech Press
5. JavaScript: The Good Parts 1st Edition by Douglas Crockford / O'Reilly
6. JavaScript Step by Step 2nd Edition by Steve Suehring/PHI Learning
7. HTML5, JavaScript and jQuery by Dane Cameron/Wiley India
8. Sams Teach Yourself JavaScript in 24 Hours (English) 1st Edition Phil Ballard , Michael Moncur/Pearson Education
9. High Performance JavaScript 1st Edition by ZAKAS / O'Reilly

Session 1: Introduction to JavaScript

- Introduction to JavaScript
- Advantages of using Java Script on client side over VB Script
- Variables in Java Script
- Statements
- Operators
- Comments
- Constructs
- Functions
- Expressions
- JavaScript console
- Scope
- Events
- Strings, String Methods
- Numbers, Number Methods

Suggested Teaching Guidelines for
MEAN Stack –PG-DAC August 2019

- Dates, Date Formats, Date Methods
- Arrays, Array Methods
-

Assignment Lab:

1. You are having sample html file with a submit button. Now modify the style of the paragraph text through javascript code
2. Writing programs to design a form to input student's data for registering a course.
3. Display a complete date with the name of the Session and name of the month
4. Write a program to sort input strings.
5. Implementing mathematical assignments like factorial, Matrix Multiplication, etc. in Java Script

Session 2:

- Booleans
- Comparisons
- Control Structures
- Conditions
- Switch
- Loop For
- Loop While
- Break
- Functions, Function Definitions, Function Parameters, Function Invocation, Function Closures

Assignment Lab:

1. Write a JavaScript function to validate whether a given value type is Boolean or not
2. Write a JavaScript function to check whether given value types are same or not.
3. Write a JavaScript program to sort a list of elements using Quick sort.
4. Write a JavaScript program to sort a list of elements using Insertion sort.
5. Write a JavaScript program to sort a list of elements using Bubble sort
6. Writing programs for implementing user inputs and displaying the same. Use HTML to design input and output forms. Write functions to implement the above assignments.

Session 3 :

- Objects, Object Definitions, Object Properties, Object Methods, Object Prototypes
- Object Oriented Programming
 - Method
 - Constructor
 - Inheritance
 - Encapsulation
 - Abstraction

Suggested Teaching Guidelines for
MEAN Stack –PG-DAC August 2019

- Polymorphism

Assignment – Lab:

1. Write a JavaScript program to get the website URL
2. Write a JavaScript program to list the properties of a JavaScript object

Session 4:

- JavaScript Validations
- Document Object Model, Document and Events (DOM Manipulation)
- HTML DOM
- DOM Intro
- DOM Methods
- DOM Document
- DOM Elements
- DOM HTML
- DOM CSS
- DOM Animations
- DOM Events
- DOM EventListener
- DOM Navigation
- DOM Nodes
- DOM Nodelist
- Object hierarchy in Java Script

Assignment – Lab:

1. Validate the resume form using the Java Script

Session 5:

- Debugging
- Type Conversion, Regular expressions
- Errors, Debugging
- Forms
 - Forms Validation
 - Forms API

Assignment – Lab:

1. Write a JavaScript function to get the values of First and Last name of the following form
2. Write a JavaScript function to validate whether a given value is RegExp or no

Suggested Teaching Guidelines for
MEAN Stack –PG-DAC August 2019

Session 6:

- JS Browser BOM, Window, Screen, Location, History, Navigator, Popup Alert, Timing, Cookies
- Javascript Windows
- Pushing code quality via JSLint tool
- Security in Java Script

Assignment – Lab:

1. Design Different windows forms to input user commands and process them through output screen

Session 7: JQuery

- Introducing to jQuery
- Basics of jQuery coding
- Selecting the elements
- Bringing pages to life with jQuery
- jquery selection and events
- JQuery Events

Assignment – Lab:

1. Write a jQuery Code to get a single element from a selection, you have the HTML page.
2. Write jQuery Code to add a tag at the beginning of the list item, containing the index of the list item
3. You are having sample data for the link. Write jQuery Code to change the hyperlink and the text of an existing link.

Session 8 & 9: JQuery

- Energizing pages with animations and effects
- jQuery Effects
- jquery traversal and manipulation
- Data attributes and templates
- DOM with jQuery utility functions
- jQuery Plugins
- jQuery / Google Web Toolkit

Assignment – Lab:

1. Write jQuery for the following:
 - Attach a click and double-click event to the <p> element.
 - Hide all headings on a page when they are clicked
 - Find the position of the mouse pointer relative to the left and top edges of the document

Suggested Teaching Guidelines for
MEAN Stack –PG-DAC August 2019

- Count the number of milliseconds between the two click events on a paragraph.
- Display the window width while (or after) it is resized
- Count the number of specific elements
- Set a timer to delay execution of subsequent items in the queue.
- Create a Zebra Stripes table effect
- Print a page using jQuery
- Make first word bold of all elements

Session 10:

- Introduction to Node.js
- Node modules
- Selectors
- Syntax
- Developing node.js web application
- Event-driven I/O server-side JavaScript

Assignment – Lab:

1. Installing node.js
2. Working with Node modules
3. Write a recursive function in Node.js
4. Write a Node program that prints all the numbers between 1 and 100, each on a separate line. A few caveats:
 5. if the number is divisible by 3, print “fizz”
 6. if the number is divisible by 5, print “buzz”
 7. if the number is divisible by both 3 and 5, print “fizzbuzz”
8. Develop node.js web application

Session 11 : Express

- Introduction to Express
- First Express Application
- Application, Request and Response Objects
- Implementing MVC Pattern
- Express application configuration
- Rendering Views

Assignment – Lab:

1. Installing Express
2. Use Node and Express to write a simple web application that checks whether a number is a prime or not and display message accordingly.
3. Build any previous assignment using Node.js web application using Express Application
4. Create a web application using Node and Express framework

Suggested Teaching Guidelines for **MEAN Stack –PG-DAC August 2019**

Session 12 & 13: AngularJS

- Introduction
- Angular 6 Architecture
- Language Choices
- Introduction to Components, Component Interaction & Component Styles
- Templates, Interpolation, and Directives
- Forms, User input, Template-driven Forms, Form Validations
- Data Bindings and Pipes
- Building Nested Components
- Services and Dependency Injection
- Retrieving Data Using HTTP
- Navigation and Routing Basics
- Angular Modules

Assignment – Lab:

Exercise: Setup & Creating Project in Angular

- Creating a new project
- Exploring the File Structure
 - TypeScript Configuration File (tsconfig.json) & TypeScript Definitions File (typings.json)
 - npm Package File (package.json)
 - SystemJS Configuration File (systemjs.config.js)
 - index.html File Scripts
- Bootstrapping

Exercise: Handling Events with Event Binding, Two-Way Data Binding

Exercise: Building Out the application with Components, Services, and Providers

Exercise: Working with HTTP, Observables and Reactive Extensions, Sending an HTTP Request, Subscribing to an Observable

Exercise: Configuring Routes, Tying Routes to Actions, Placing the Views

Session 14 & 15: Testing Web Applications

- Introduction to Testing web Applications
- JavaScript Application Testing
- Testing Express Applications
- Testing AngularJS Applications
- Testing based on
 - Functionality Testing
 - Performance Testing
 - Usability Testing
 - Server Side Interface
 - Client Side Compatibility
 - Security

Suggested Teaching Guidelines for
MEAN Stack –PG-DAC August 2019

Assignment – Lab:

- Assignment for testing of different web applications
 - Test if jQuery is loaded.
 - Testing for express application
 - Testing for AngularJS application

Suggested Teaching Guidelines for **Database Technologies – PG-DAC August 2019**

Duration: 30 classroom hours + 30 lab hours (60hrs)

Objective: To introduce the student to RDBMS and Facilitate Hands On Experience on (Object Relational Database)

Prerequisites: Knowledge of programming in C++ Clarity of Object Oriented concepts

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Text Book:

1. MySQL Developer's Library by Paul DuBois / Pearson

Reference:

1. Professional NoSQL [eBook] by Shashank Tiwari / Wiley
2. Mysql: The Complete Reference 1st Edition by Vikram Vaswani / Mcgraw Hill Education
3. MySQL High Availability 1st Edition by Charles Bell O'reilly
4. Oracle 9i DBA Certification Guide
5. Oracle 9i Beginners Guide PL/SQL Guide – Orilley's
6. Creating your MySQL Database: Practical Design Tips and Techniques by Marc Delisle / Packt Publishing
7. High Performance MYSQL, 3rd Edition by SCHWARTZ / Shroff Publishers
8. MURACH'S MYSQL by Joel Murach / Shroff Publisher
9. MySQL Cookbook: Solutions for Database Developers and Administrators by DuBois / O'Reilly Media
10. Nosql for Dummies by Fowler Adam Fowler / John Wiley & Sons
11. Beginning Databases with PostgreSQL: From Novice to Professional 2nd Revised edition by Richard Stones Neil Matthew Stones Matthew / Apress
12. Practical PostgreSQL 1st Edition by Joshua Drake John Worsley / O'Reilly

Session 1:

Lecture

- Introduction to DBMS – What is DBMS, Its need
- Areas where DBMS are used
- Types of DBMS:
Introduction to Hierarchical Model, Network and Relational Models,
Object Oriented Database
- Data models (conceptual physical and logical)

Assignment – Reading:

Read and understand Chapters for the next session

Session 2:

Lecture:

Suggested Teaching Guidelines for Database Technologies – PG-DAC August 2019

- Data Integrity & integrity rules
- Codd's 12 rules for a Relational Database (conclusion)
- Need for Normalization.

Assignment – Reading:

Read and understand Chapters for the next session

Session 3:

Lecture:

- Various normalization forms 1st normal form, 2nd normal form
- 3rd normal form,
- Introduction to 4th, BCNF, etc
- Need for De-normalization

Assignment – Reading:

Read and understand Chapters for the next session

Session 4:

Lecture:

- DDL Commands
- DML & DCL Commands
- Inbuilt Functions

Assignment – Lab:

SQL Practice Questions Including:-

DDL Commands: Create/Alter/Drop/Grant/Revoke

DML Commands: Select/Insert/Update/Delete/Truncate

DCL Commands: RollBack Commit

Number Functions: -

Single Value Functions: NVL, ABS, CEIL etc

Group Value Functions: AVG, COUNT, MAX etc

Assignment – Reading:

Read and understand Chapters for the next session

Case study on Oracle Architecture

Session 5:

Lecture:

- Grouping Things Together (Group By, Having Clause)
- Advance Sub-queries (Correlated Sub query, Outer Joins)
- Set Operators (UNION, UNION ALL, INTERSECT, MINUS)

Assignment – Lab:

SQL Practice Questions:-

Queries containing Group By, Having Clause,

Correlated Queries, SubQueries, Outer Joins and Set Operators

Assignment – Reading:

Suggested Teaching Guidelines for **Database Technologies – PG-DAC August 2019**

Read and understand Chapters for the next session

Session 6

Lecture:

- Types of Views
- Creating Complex View
- Using Temporary Tables
- Family Trees & Connect BY

Assignment – Lab:

SQL Practice Questions Including
Creating Simple/Complex Views,
Creating Temporary Tables,
Use of Connect by Clause

Assignment – Reading:

Read and understand Chapters for the next session

Session 7

Lecture:

- Introduction to MySQL Programming
- Exceptions

Assignment – Lab:

SQL Practice Questions covering MySQL programming & Exceptions

Assignment – Reading:

Read and understand Chapters for the next session

Session 8

Lecture:

- Cursors
- Procedures
- Functions

Assignment – Lab:

SQL Practice Questions covering Cursors, Procedures & Functions

Assignment – Reading:

Read and understand Chapters for the next session

Session 9

Lecture:

- Triggers
- Indexes

Assignment – Lab:

SQL Practice Questions covering index and Triggers

Assignment – Reading:

Read and understand Chapters for the next session

Suggested Teaching Guidelines for Database Technologies – PG-DAC August 2019

Session 10

Lecture:

- Clusters
- Snapshots

Assignment – Lab:

SQL Practice Questions including Index, clusters, snapshots

Assignment – Reading:

Read and understand Chapters for the next session

Session 11 & 12: NO SQL

Lecture:

- Introduction to No SQL database
- Data Persistence on Server-Side via NoSQL
- Does not use SQL-like query language (which supports the idea of "tables")
- Longer persistence
- Store massive amounts of records
- Many systems are optimized for retrieval (query) and appending (write) operations
- Systems can be scaled
- High availability
- Semi-structured data
- Support for numerous concurrent connections
- Indexing of records for faster retrieval
- Difference between a RDBMS and a No SQL database
- Understanding the Storage Architecture
 - Working with Column-Oriented Databases
 - Document Store Internals

Assignment – Lab:

Practice Questions including SQL write Operations.

Assignment – Reading:

Read and understand Chapters for the next session

Session 13 & 14:

Lecture: NO SQL

- How to Configure and access the store using the Admin Console
- Performing CRUD Operations
 - Creating Records
 - Accessing Data
 - Updating and Deleting Data
- Working with Language Bindings
- Querying NoSQL Stores
- NoSQL database MongoDB
- The big ideas: documents, collections
- MongoDB is a document-based database system

Suggested Teaching Guidelines for Database Technologies – PG-DAC August 2019

- The abstract idea of a document - data encapsulation into some file format. In the case of MongoDB, JSON is used.
- If you are familiar with relational databases and SQL, then analogies: tables => collections; records (or rows) => documents
- Each document can have different fields from other documents
- All documents must have an _id field
 - Similarities Between SQL and MongoDB Query Features
 - Accessing Data from Column-Oriented Databases Like HBase
- Schema structure for Oracle NoSQL database
 - Changing Document Databases
 - Schema Evolution in Column-Oriented Databases
 - Data Evolution in Key/Value Stores
- What is MongoDB Internals
- Essential Concepts behind a Database Index
- Indexing and Ordering in MongoDB
- Creating and Using Indexes in MongoDB

Assignment – Lab:

Practice Questions including CRUD Operations
 MongoDB Operations
 Practice Questions including Column-Oriented Databases
 Selecting a database to use
 Showing all collections in database
 Finding / querying documents in a collection
 Inserting documents into a collection
 Deleting documents in a collection
 Update documents in a collection
 Practice to create and using Indexes in MongoDB

Session 15

Lecture:

- What is BigData
- Using MySQL as a NoSQL Solution
- Mostly Immutable Data Stores
- Migrating from RDBMS to NoSQL

Assignment – Lab:

Practice how to Migrating from RDBMS to NoSQL.

Suggested Teaching Guidelines for
Java Technologies-I (Core Java)–PG-DAC August 2019

Duration: 36 classroom hours +34 lab hours (70hrs)

Objective: To introduce the student to Core Java Technologies

Prerequisites: Knowledge of object oriented programming

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Text Book:

1. Core and Advanced Java Black Book Black by Dreamtech Press

Reference:

1. Java 8 Programming Black Book by Dreamtech Press
2. Core Java : Fundamentals - Volume 1 Gary Cornell, Cay S. Horstmann/ Pearson
3. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
4. Core Java : Advanced Features - Volume 2 Gary Cornell, Cay S. Horstmann/ Pearson
5. Beginning Java 2 by Ivor Horton; Wrox Publication
6. The Complete Reference Java Eight Edition, Herbert Schidt/ TMH
7. Object-Oriented Analysis and Design with applications by Booch
8. Core Java 8 for Beginners by Sharanam Shah, Vaishali Shah / Shroff Publishers & Distributors
9. Murach's Java Programming 4th edition by Joel Murach / Shroff Publishers & Distributors
10. Advanced Java programming by Uttam K Roy / Oxford University press
11. Sun Certified Enterprise Architect For Java EE Study Guide by Cade, 2nd Edition (Paperback)
12. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
13. **Professional Java EE Design Patterns by Murat Yener, Alex Theedom, Reza Rahman (Paperback)**

Session 1:

Lecture

- Java 8 Basics :Overview of Java, Features of Java, Scope of variables
- JDK and its usage (Java Compiler, Java Runtime, Java Debugger, Java doc)
- Working with Data Types: Structure of a Java Class, Importing Packages, Difference between object reference variables and primitive variables, how to read or write to object fields)

Session 2:

Lecture

- Object's lifecycle(creation, reassignment, garbage collection: new, finalize)
- Wrapper classes (Boolean, Double and Integer)

Suggested Teaching Guidelines for
Java Technologies-I (Core Java)–PG-DAC August 2019

- Operators (Unary, Binary, Arithmetic, Assignment, Compound, Relational, Logical, Equality) and Control Statements (if, if-else, for, while, switch, do-while, break and continue, ternary constructs)

Session 3:**Lecture**

- Packages and classpath
- Arrays
- Understanding of String Class, StringBuilder Class, StringBuffer class
- Methods and Encapsulation: Methods, Access Modifiers, Method Overloading, Passing Data, Creating Constructors, Immutable Classes

Assignment – Lab:

Get yourself acquainted with java environment. Build a class Emp, which contains details about the employee and compile and run its instance

Assignment – Reading:

Study the book Java FAQ

Assignment – Tutorial:

Compare syntactical similarities and dissimilarities between Java and C++

Session 4:**Lecture**

- Class Inheritance, Abstract Classes, Inner Classes, Interface and Implementation classes.
- Understanding Polymorphism: Object vs Reference, Object Casting, Virtual Methods, Method Overriding

Assignment – Lab:

Create an inner class for a manager, which contains information about the manager. Use the appropriate interfaces. Create an anonymous inner class for Tech. Members using the Session one assignment

Session 5:**Lecture**

- Exception-Handling: Basics, Role of Exceptions, Types
- Using try and catch, Multiple Catch, Nested try (throw, throws, finally)
- Built-in Exceptions, Runtime Exceptions Checked Exceptions, Errors
- Creating own Exception Subclasses

Assignment – Lab:

Create a user defined exception to check whether your employee exist in your data structure and using the catch and finally block. Redeem an appropriate solution.

Session 6:**Lecture**

- Enumerations, Auto boxing, and Annotations

Suggested Teaching Guidelines for
Java Technologies-I (Core Java)–PG-DAC August 2019

- Lambda Expressions
- Java 8 New Features

Session 7 & 8:**Lecture**

- Java API: java.util, java.lang, java.math

Assignment – Lab:

Create an appropriate data structures to store your employee object and use the java.util.package properties.

Session 9 & 10:**Lecture**

- Generics and Collections

Assignment – Lab:

1. Implement String class and util package
2. Using the collection framework define an appropriate interface to your above application

Assignment – Lab:

Create a user defined exception to check whether your employee exist in your data structure and using the catch and finally block. Redeem an appropriate solution.

Session 11:**Lecture**

- Java NIO (NIO 2) Overview
- NIO classes: Fundamentals, Path Interfaces, Manage metadata of a file or directory,
- Byte Buffers & Channels
- UDP, TCP and IP
- Communication with TCP/IP Protocol

Assignment – Lab:

- Implement to Send File Contents (two way communication Java)
- A Simple Java TCP Server and TCP Client

Session 12:**Lecture**

- Java Concurrency: Using threads in Java, Life cycle of thread
- Advantages and issues
- Thread class, thread groups
- The Runnable interface

Session 13:**Lecture**

- Synchronization, Inter-Thread communication

Suggested Teaching Guidelines for
Java Technologies-I (Core Java)–PG-DAC August 2019

- Executor Framework overview

Assignment – Lab:

Using Multi-Threading create concurrent java application , to write data to file in a thread safe manner.

Apply Thread safety to Collection Framework API classes

Session: 14 & 15**Lecture**

- The java.io Package
- Files
- Byte Streams and Unicode Character Streams
- Persistence of objects
- Object Serialization Methods

Assignment – Lab:

Make your above Employee, manger classes objects persistent.

Session: 16**Lecture: Reflection in Java**

- Java Reflection Classes, Methods, Getter Setters, Constructors, Annotations, generics, Arrays, Dynamic method invocation

Assignment – Lab:

Create a new array, whose size and component type are not known until runtime, and then modify the array's components

Session: 17: Java Virtual Machine**Lecture**

- What is a Java Virtual Machine?
- The Lifetime of a Java Virtual Machine
- The Architecture of the Java Virtual Machine
- Java Mail
 - javax.mail.internet Class ContentType
 - Method
 - java.lang.String, getBaseType()
 - java.lang.String, getParameter(java.lang.String name)

Assignment – Lab

- Configuring JavaMail API and sending test mails
- Implement to send multiple mails, mails with attachments, calendar appointment etc.

Suggested Teaching Guidelines for
Java Technologies-I (Core Java)–PG-DAC August 2019

Session 18:

Lecture

- Introduction of JDBC API
- JDBC Architecture
- JDBC Drivers
- Drivers, Connection, Statement , PreparedStatement and Result Set interfaces and their relationship to provider implementations
- Writing JDBC Application along with DAO & POJO Layers
- Stored Procedures and functions invocation

Suggested Teaching Guidelines for
Java Technologies-II (Web Based Java)–PG-DAC August 2019

Duration: 46 classroom hours + 44 lab hours **(90 hrs)**

Objective: To introduce the student to Java Technologies

Prerequisites: Knowledge of object oriented programming

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Text Book:

1. Core and Advanced Java, Black Book by Dreamtech Press .

Reference:

1. Java 8 Programming Black Book by Dreamtech Press
2. Core Java : Fundamentals - Volume 1 Gary Cornell, Cay S. Horstmann/ Pearson
3. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
4. Core Java : Advanced Features - Volume 2 Gary Cornell, Cay S. Horstmann/ Pearson
5. Beginning Java 2 by Ivor Horton; Wrox Publication
6. The Complete Reference Java Eight Edition, Herbert Schidt/ TMH
7. Object-Oriented Analysis and Design with applications by Booch
8. Core Java 8 for Beginners by Sharanam Shah, Vaishali Shah / Shroff Publishers & Distributors
9. Murach's Java Programming 4th edition by Joel Murach / Shroff Publishers & Distributors
10. Advanced Java programming by Uttam K Roy / Oxford University press
11. Sun Certified Enterprise Architect For Java EE Study Guide by Cade, 2nd Edition (Paperback)
12. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
13. Professional Java EE Design Patterns by Murat Yener, Alex Theedom, Reza Rahman (Paperback)

Suggested Teaching Guidelines for
Java Technologies-II (Web Based Java)–PG-DAC August 2019

Session 1: J2EE Overview**Lecture**

- J2EE Container
- Packaging Web applications
- J2EE compliant web application
- Deployment tools.
- Web application life cycle
- Deploying web applications.
- Web Services Support

Session 2,3,4**Lecture**

- Servlets : Dynamic Content Generation
- Advantages of Servlets over CGI
- Servlet Life cycle
- Servlet API & Deployment
- Servlet Annotations
- The Servlet interface
- The HttpServlet, HttpServletRequest, HttpServletResponse
- Exception Handling
- Servlet , DAO , POJO DB Layers
- Session
- Session Management
- Session Tracking with
 - Cookies
 - HttpSession
- Request Dispatcher
- Page Navigation
- Complete Case study Servlet Based

Assignment – Lab:

Implement exception handling in Servlet.

Use Java Servlets technology in designing and implementing an Air Ticket reservation system.

Incorporate Sessions in the Air Ticket reservation system.

Assignment – Reading:

Know more about the HTTP protocol at www.w3c.org

Assignment – Tutorial:

Compare which way of session tracking is better Cookies or HttpSession.

Session 5:**Lecture**

- Internationalization and Localization: Basics
- Read and set the locale by using the locale object
- Create and read Properties file

Suggested Teaching Guidelines for
Java Technologies-II (Web Based Java)–PG-DAC August 2019

- Build a resource bundle for each locale and load a resource bundle in an application

Assignment – Lab:

Deploy structured web application.

Session 6 & 7:**Lecture**

- JSP: Separating UI from Content generation code
- MVC architecture
- Life cycle of a JSP page
- Directives, Implicit and Explicit Objects, Scriptlets, Expressions, Expression Language
- Scope
- JSP Error Page handling
- Session Tracking
- JSP Using JavaBeans
- Custom Actions and Tag Libraries in JSP

Assignment – Lab:

Separate UI code from the controller code in your Air Ticket reservation system by incorporating JSP and Servlets. Complete the implementation of Air-ticket reservation system

Session 8 :**Lecture****JavaBeans**

- JavaBean Component
Model of MVC architecture
- Writing JavaBeans Components
 - Properties
 - Methods
 - Events
- JavaBeans Component Design Conventions
- Creating and Using a JavaBeans Component
- Setting JavaBeans Component Properties
- Retrieving JavaBeans Component Properties
- JSP Using JavaBeans

Assignment – Lab:

1. Creating a Project by Simple Bean code
2. Implement MVC based web application using servlet, JSP, JavaBeans

Session 9 & 10 : JNDI , Annotations ,Transaction Management**Lecture**

- JNDI API
- JNDI Overview
- Java Annotations : Purpose, Basics, Annotation Elements

Suggested Teaching Guidelines for
Java Technologies-II (Web Based Java)–PG-DAC August 2019

- Retention Policy
- Built-in Annotations
- Java Custom Annotations
- Using Custom Annotation
- Transaction Management
- Transaction Timeouts , Transactions in Web Components
- Transactions and Concurrency
- Transaction Management example

Assignment – Lab:

Develop Courier Tracking system implementing annotation.

Session 11:**Lecture**

- Design Patterns in Java: Overview, Usage, Types of Design Patterns
- Creational: Factory, Singleton, Builder, Prototype
- Structural: Adapter, Composite, Proxy, Facade, Bridge, Decorator
- Behavioral: Template method, Mediator, Chain of Responsibility, Observer, Strategy, Command, State, Visitor
- MVC Pattern, Data Access Object Pattern
- Front Controller Pattern
- Service Locator Pattern
- Transfer Object Pattern

Assignment – Lab:

Develop candidate examination system implementing above design pattern.

Session 12 & 13:**Lecture**

- Apache Maven: Overview, Environment Setup, Ant vs Maven
- POM, Build Life Cycle, Build Profiles
- Maven Repository
- Create, Build and Test Project & Build Automation
- Manage Dependencies, Deployment Automation

Assignment – Lab:

Configure Apache Maven in web application.

Develop a web application using Apache Maven.

Session 14 & 15:**Lecture**

- Hibernate Framework
 - Introduction to Hibernate Framework
 - Architecture
- Hibernate in IDE
 - Creating web application using Hibernate API

Suggested Teaching Guidelines for
Java Technologies-II (Web Based Java)–PG-DAC August 2019

- Life-cycle of Hibernate Entities
- HB with annotation example
- Hibernate Mappings and Relationships
- Collection and Component Mapping
- HQL ,Named Queries , Criteria Queries

Assignment – Lab:

Develop a web application (Online Bookshop) using Hibernate Persistence

Study Hibernate architecture from www.hibernate.org/docs

Session 16 :

Lecture

Spring Boot Essentials

- Why Spring Boot
- Spring Boot Overview
- Building web application with Spring Boot
- Building RESTful web service using Spring Boot
- Overview of Spring Data JPA

Assignment – Lab

- Design & deploy online stock trading system using spring Boot MVC

Session 17 & 18 :

Lecture

- Overview of Spring4/5 Architecture.
- AOP Overview
- Spring Modules Overview
- Spring MVC architecture
- Understanding Spring 4 annotations
- Spring Application
- Spring in IDE
- Spring in Eclipse
- Dependency Injection

Assignment Reading

Understand key features of Spring Architecture & design simple Java application to test dependency injection.

Session 19 & 20:

Lecture

- What is IoC(Inversion of Control)
- IOC container
- Dependency Injection
- Spring AOP

Suggested Teaching Guidelines for
Java Technologies-II (Web Based Java)–PG-DAC August 2019

- AOP Concepts
- Spring ORM
- Spring MVC
- Model, Model & View , HandlerMapping, ViewResolver, Front Controller
- Deployment of web application using Spring MVC Form with CRUD, File Upload example
- Integration of Spring MVC with Hibernate
- Spring Boot Basics , Overview & Demo

Assignment – Lab

Design & deploy online stock trading system using spring MVC module
Modify earlier assignment to support persistence via Hibernate

Session 21 & 22:**Lecture****Building REST services with Spring**

- Introduction to web services
- SOAP Vs RESTful web services
- RESTful web service introduction
- Create RESTful web service in java using Spring framework
- RESTful web service JSON example
- RESTful web service CRUD example
- AngularJS and Spring based RESTful web service CRUD Integration

Assignment – Lab

Design & deploy online stock trading system using spring MVC module
Modify earlier assignment to support persistence via Hibernate

Session 23:**Lecture**

- Testing in Spring
- Unit Testing of Spring MVC Controllers:
- Unit Testing of Spring Service Layer
- Integration Testing of Spring MVC Applications: REST API
- Unit Testing Spring MVC Controllers with REST

Assignment – Lab

Design & test Spring Application.

Suggested Teaching Guidelines for **Microsoft .NET– PG-DAC August 2019**

Duration: 40 class room hrs. + 30 lab hrs (70 Hrs)

Objective: To acquire the knowledge of MS .NET Framework 4.5 or higher version

Prerequisites: Students are expected to have knowledge of COM development and XML

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

Note: Training will be carried out on .Net 4.5 or latest version of the software

List of Books / Other Training Material:

Text Book:

1. ASP.Net 4.5 Covers C# and VB Codes: Black Book by Kogent Learning Solutions Inc / Dreamtech Press

Reference books:

1. Beginning ASP .NET 4.5 in C# by Matthew MacDonald/Apress
2. .Net 4.5 Programming 6-in-1: Black Book by Kogent Learning Solution/ Wiley
3. Pro C# 5.0 and the .NET 4.5 Framework by Andrew Troelsen / Apress
4. .NET Programming 4.0, Wrox Publications
5. Introducing Microsoft .NET, Wrox Publications
6. CLR via C# by Jeffrey Richter / Microsoft Press
7. Pro C# 2010 And The .NET 4 Platform, Andrew Troelsen
8. C# 2010 Programming: Covers .Net 4.0, Black Book
9. C# 2010 For Programmers, Paul J. Deitel, Harvey M. Deitel
10. Professional ASP.NET MVC 4 Jon Galloway Phil Haack Brad Wilson K. Scott Allen /Wiley
11. Beginning ASP .Net 4.5 in C# and VB by Imar Spaanjaars /Wiley India
12. Pro ASP.Net 4 In C# 2010, Matthew Macdonald, Adam Freeman, Mario Szpuszta
13. Asp .Net MVC Framework by Stephen Walther /Pearson
14. Murach's ASP.NET 4.5 web programming with C# 2012 5th edition by Mary Delamater, Anne Boehm / Shroff Publishers & Distributors
15. Professional ASP.NET 4.5 in C# and VB Gaylord J N / Wiley India Pvt. Ltd

Duration: 10 class room hrs.

Session 1:

Lecture:

- Introduction to .Net Framework
- Difference between .NET 2.0, .NET 3.5 and .NET 4.0/4.5
- .NET Framework Overview
 - ✓ Objectives of .NET Framework 4.5
 - ✓ Components of .NET Framework 4.5
 - ✓ Development in .NET Framework 4.5

Assignment Reading:

Read about NET 4.5 Framework from MSDN.

Session 2:

Lecture:

- Execution Process in .NET Environment
- Inside .NET Framework

Suggested Teaching Guidelines for
Microsoft .NET– PG-DAC August 2019

- ✓ CLR
- ✓ Managed and unmanaged code
- ✓ MSIL
- ✓ CTS
- ✓ MetaData
- JITters

Session 3:**Lecture:**

- Assemblies-The Building Blocks
- Assembly Benefits
- Assembly Contents
- Different between a normal .EXE File and a PE file
- Search order of an assembly
- Using reflection to build a dynamically extensible application.
- Discovering type Defined in an assembly

Assignment Lab:

- ✓ Create a Satellite Assembly in NET 4.5 Framework
- ✓ Create a global Assembly in NET 4.5 Framework

Session 4:**Lecture:**

- Application Domain
 - ✓ Accessing objects Across AppDomain Boundaries
 - ✓ AppDomain Unloading
 - ✓ CLR hosting
 - ✓ How host use AppDomain
 - ✓ Managing CLR by using Managed Code
 - ✓ How a Host Gets its Thread back.
- Language Interoperability
- . NET Framework Class Library
- The IL dis-assembler
- Migrating to MS .NET

Session 5:**Lecture:**

- Windows Presentation Foundation
- Windows Workflow Foundation
- Windows Communication Foundation

Assignment Reading:

Read about WPF, WWF, and WCF from MSDN.

Suggested Teaching Guidelines for
Microsoft .NET– PG-DAC August 2019

Duration: 12 class room hrs. + 12 lab hrs

Objective: To acquire the knowledge of C#. NET 4.5

Prerequisites: Knowledge of C++ programming, OOPs, Database Programming and XML is also a must.

Session 1:

Lecture:

- Need of C#
- Strengths of C#
- C# Basics
 - Program Structure
 - Data Types
 - Variables, Constants, Operators
 - Flow Control in C#

Assignment Reading:

Read about C# specification from MSDN.

Assignment Lab:

Write a program in C# to generate prime numbers between 1 and 1000.

Session 2:

Lecture:

- Operators
- Type Safety
- Object Comparison
- Operator overloading
- User defined cast

Assignment Reading:

Read about C# specification from MSDN.

Assignment Lab:

Implement that topics in your program

Session 3:

Lecture:

- Namespaces & Assemblies
- Arrays
 - Simple, multidimensional and jagged array
 - Array class
- Enumerations
- Properties & Indexers

Assignment Reading:

Read about C# specification from MSDN.

Assignment Lab:

Create namespace and assemblies.

Hands on Arrays and properties and indexes

Suggested Teaching Guidelines for **Microsoft .NET– PG-DAC August 2019**

Session 4:

Lecture:

- Preprocessors
 - Preprocessor Usage
 - Conditional Compilation
 - Assemblies
 - What are Manifests??
 - Writing Private Assemblies
 - Writing Shared Assemblies
- Delegates and Events
- Boxing and Unboxing
- Reflection and the Type class

Assignment Reading:

Read about C# specification from MSDN.

Assignment Lab:

Work with delegates, events, boxing, unboxing Manifest.

Session 5:

Lecture:

- Regular Expression
- Attributes
- Genetics Concepts
- Generics Collections
 - Collection Interface
 - List, Queues, Stack,
 - Dictionaries, Hash set

Assignment Reading:

Read about C# specification from MSDN.

Assignment Lab:

Implement Collections classes in C#

Session 6:

Lecture:

- Error Handling (Exceptions Handling)
 - Checked & Unchecked Statements
 - The try, catch, finally
 - Dos & Don'ts of Exception Handling
- User Defined Exception classes
- WPF
 - XAML & Controls
 - Layouts , Styles , Resources
 - Data Binding
 -

Assignment Reading:

- Read about C# specification from MSDN

Assignment Lab:

Write a program and do all the exception

Suggested Teaching Guidelines for
Microsoft .NET– PG-DAC August 2019

Duration: 18 class room hrs + 18 lab hrs

Objective: To acquire the knowledge of ASP.NET 4.5

Prerequisites: Knowledge of COM development, Database Programming and XML is also a must.

Session 1 & 2:

Lecture:

- **Introduction to Asp .net MVC**
 - Architecture of an ASP .Net MVC application
 - Understanding Folder structures and configuration files
- **Understanding Controllers and Action**
 - Create a controller
 - How actions are invoked
 - HttpGet , HttpPost , NoAction Attributes
 - Running Action result.
- **Understanding views & Models**
 - Creating Razor Views
 - HTML Helper Functions
 - Understanding ViewBag
 - Create a view using viewbag
- **Understanding Routing & Request Life Cycle**
 - Routing Engine & Routing Table
 - Understanding and configuring Routing Pattern in RouteConfig File
 - Understanding 404 error and resource not found.
 - Using Attributes Routing
 - Understanding Request Life Cycle
- **Layouts , Bundle , Minification**
 - Creating Layout and using with associated views
 - Understanding Bundling and Minification
 - Using BundleConfig file
 - Attaching css , js , bootstrap in bundles
- **Understanding Models & Views**
 - Creating Models & ViewModel
 - Validation using Data Annotations
 - Client side and server side validation
 - Self validated model
 - Creating Strongly Types Views
 - Using Various Scaffold Templates
 - CRUD operation using Model
- **MVC State Management**
 - ViewBag , TempData , Session , Application
 - Cookies , QueryString
- **MVC Module**
 - Partial View
 - Action Method and child action

Suggested Teaching Guidelines for **Microsoft .NET– PG-DAC August 2019**

- Custom Helper Function
- Asynchronous Actions
- Error Handling in MVC with Log Entry
- Filters and Custom Action Filter

- **MVC Security**
 - Using Authorize & Allow Anonymous attributes
 - Implementing Forms Based Authentication
 - Preventing Forgery Attack using AntiForgeryToken
 - Preventing Cross Site Scripting Attack
 - Intro of OAuth

- **Entity Framework**
 - Introduction to EF
 - Different Approaches
 - Using Code First Approach
 - Using various Data Annotations
 - Using Validation, Primary Key , Foreign Key etc
 - Using Fluent APIs
 - Database Migrations
 - Queuing data using Lambda Expression
 - CRUD operation using EF

- **Developing MVC application using EF Code First Approach**

- **Web APIs**
 - Creating Web APIs
 - Different Verbs
 - Consuming using a client
 - Using Newtonsoft APIs

- **Understanding [ASP.Net](#) MVC Core**
 - Difference between MVC and MVC Core
 - Creating a simple MVC Core Application

Session 3:

Lecture

- **Threading**
 - Threading & Synchronization
 - Life Cycle of a Thread
 - Synchronizing critical data using Synchronization objects
 - Thread Pool
 - Task Programming using a async & await keyword
- **Querying with LINQ**
 - LINQ to objects
 - LINQ to XML
 - LINQ to SQL

Suggested Teaching Guidelines for
Microsoft .NET– PG-DAC August 2019

- **Working with XML and LINQ to XML**
 - Basic of XML
 - XML InfoSet
 - XSD-XML Scheme Definition
 - Editing and XML Schema in Visual Studio 2008/2010
 - XmlReader and XmlWriter
 - XmlDocument and XPathdocument
 - DataSets
 - The XmlDataSource Control
 - XSLT
 - Database and XML
 - For XML AUTO
 - SQL Server 2008/2012 and XML Data Type
- **Using Lambda Expression**
 - Understanding Lambda Expression
 - Queuing data using Lambda
 - First , Single , where , join , group by etc

Session 4:**Lecture:**

- **Data Management with ADO.NET 4.5**
 - Basic ADO.Net features
 - Connection object, Command object, DataReader, DataAdapter, DataSet and DataTable.
 - Asynchronous command Execution
 - Asynchronous Connections

Session 5:**Lecture:**

- **Files I/O and Streams**
 - Working with drivers, Directories, and Files
 - Reading and Writing files
 - Working with Serial ports
 - Networks Communications
 - HttpClient API
 - Sending mail
- **WebServices**
 - The Need for Web Services
 - Introducing Web Services
 - The Web Technology Stack and .NET
 - The .NET Alternatives to WebServices
 - Common Web Service Scenarios
 - Implementing a Web Service
 - Creating a Web Service Project

Suggested Teaching Guidelines for
Microsoft .NET– PG-DAC August 2019

- Implementing Web Service Methods, exposing them and controlling their behavior
- Managing State in an ASP .NET Web Service
- Debugging Web Services

Session 6:

Lecture:

- **Windows Communication foundation**
 - WCF Overview
 - Contracts
 - Service Contracts
 - Data Contracts
 - Message Contracts
 - Channel
 - Channel Shapes
 - Operation Contract and Channel Shapes
 - Channel Listeners
 - Channel Factories
 - ChannelFactory<>
 - ICommunicationObject
 - Binding
 - Cross-Machine Communication Between .NET Applications
 - Local Machine Communication Between .NET Applications
 - Communication Using Basic Web Services
 - Communication Using Advanced Web Services
 - wsHttpBinding
 - ws2007HttpBinding
 - wsDualHttpBinding

Assignment Lab:

- ✓ Creating a simple WCF service

Session 7:

Lecture:

- **Windows Communication foundation**
 - Binding
 - Comparing Binding Performance and Scalability
 - Creating a Custom Binding
 - User-Defined Bindings
 - Binding Element
 - Exposing a Service Contract over Multiple Bindings
 - Behaviors
 - Concurrency and Instancing (Service Behavior)
 - Exporting and Publishing Metadata (Service Behavior)
 - Implementing Transactions (Operation Behavior)
 - Hosting
 - Hosting a Service in Windows Process Activation Services

Suggested Teaching Guidelines for
Microsoft .NET– PG-DAC August 2019

- Hosting a Service in IIS 7
- Enabling ASMX Features in an IIS-Hosted Service
- Self-Hosting
- Self-Hosting in a Managed Windows Service
- Hosting Multiple Services in One Process
- Defining Service and Endpoint Addresses
- **Understanding REST Service**
 - Creating REST Service with Get & Post
 - HTML Helper Functions
 - Understanding ViewBag
 - Create a view using viewbag

Assignment Lab:

- ✓ Create a online shopping portal in ASP.NET and WCF service for communicating between asp.net application and database, Host the WCF service and Asp.NET application on IIS.

Session 8:**Lecture:**

- **Localization in MVC**
 - Cultures and regions
 - Understand Culture Type
 - Server side culture declaration
 - Client Side culture declaration
 - Asp .Net 4.5 MVC recourse files
 - Making use of local resources
 - Making use of global resources
 - Looking at the resource editor
- **IIS7**
 - Modular Architecture of IIS7
 - IIS7 and asp .net Integrated pipeline
 - Building a customized Web Server
 - IIS Manager
 - Application pool, web sites, delegation
 - Deploying a Simple Site
- **Deploying ASP .NET MVC application**
 - Configuring IIS for ASP .Net MVC
 - Mixing ASP .net Web Form and Asp .net MVC
 - Bin Deploying an Asp .NET MVC application

Assignment Lab:

- ✓ Create a Virtual directory for an ASP .NET 4.5 MVC application
- ✓ Create a simple Currency Converter web application that uses the user culture to determine the type of conversion to convert one language to other language.
- ✓ Deploy Sample Asp.Net MVC application on IIS.

Suggested Teaching Guidelines for
Effective Communication – PG-DAC August 2019

Duration: 50 class room hours

Objective: To reinforce knowledge of communication

Prerequisites: Knowledge of English.

Evaluation method: Internal Exam – 100% weightage

List of Books / Other training material

Reference:

1. Business Communication by H S Mukerjee / Oxford University Press
2. Business Communication by R K Madhukar / Vikas Publishing House Pvt. Ltd.
3. Business Communication Essentials A skills-Based Approach to Vital Business English by Courtland Bovee, John Thill / Pearson
4. Effective Business Communication by Asha Kaul / Prentice Hall of India
5. Fundamental of Technical Communications by Meenakshi Raman, Sangeeta Sharma / Oxford University Press
6. English is easy by Chetan Anand Singh/ BSC publication Co. Pvt. Ltd
7. Communication Skills Publication Year 2011 Sanjay Kumar, Pushp Lata / Oxford University Press
8. Professional Communication Skills Praveen S R Bhatia / S. Chand Publishing

Note: Each session having 2 Hours

Session 1:

Lecture

Discussion on Soft Skills
Intro to communication
Effective Communication

Session 2:

Lecture

Communication Skills and its practice
Barriers to communication
Kind of communication

Session 3:

Lecture

Pre assessment Test
Official & General Conversation
Confidence building

Session 4:

Lecture

Work place Etiquettes
Official Letter Writing
Confidence building

Session 5:

Lecture

Suggested Teaching Guidelines for
Effective Communication – PG-DAC August 2019

Official Emailing
Event Reporting
Confidence building

Session 6:

Lecture

Non-verbal Communication
Gestures

Session 7:

Lecture:

Fluency and vocabulary
Essay Writing

Session 8:

Lecture:

Pronunciation
Conversation practice
Polite Conversation

Session 9:

Lecture:

Greeting
Formal Etiquettes
Body Language

Session 10:

Lecture

Writing Covering letter
Email

Session 11:

Lecture

Writing Resume

Session 12:

Lecture

Post Assessment Test
Personal Goal Setting & Career Planning

Session 13:

Lecture

Presentation Skill

Session 14:

Lecture

Formal Speaking (Telephone, Face-to-Face, Public Speaking)
Oral & Digital Presentation Skills

Session 15:

Lecture

Presentation Skill
Developing Positive Attitude

Session 16:

Lecture

What is group discussion?
Group Discussions

Suggested Teaching Guidelines for
Effective Communication – PG-DAC August 2019

Session 17:

Lecture

Group discussion on at least 4 topics

Session 18:

Lecture

Group discussion on at least 4 topics

Session 19:

Lecture

Interview skills

Job Search Process

Session 20:

Lecture

Interview skills

Listening Skills

Session 21:

Lecture

Interview skills

Audio Synthesis

Session 22:

Lecture

Practice of Mock interview

Session 23:

Lecture

Practice of Mock interview

Session 24:

Lecture

Practice of Mock campus

Session 25:

Lecture

Mock campus

Cross-Cultural Communication

Technology-enabled Communication

Suggested Teaching Guidelines for
Aptitude & General English – PG-DAC August 2019

Duration: 50 class room hours

Objective: To reinforce knowledge of general Aptitude & English

Prerequisites: Knowledge of Mathematics & English.

Evaluation method: Internal Theory exam– 80% weightage
Internal assessment– 20% weightage

List of Books / Other training material

Reference:

1. Quicker math by M. Tyra (BSC publication co. Pvt. Ltd)
2. Quantitative Aptitude by RS Aggarwal
3. Verbal & Non-Verbal Reasoning: RS Aggarwal
4. Quantitative Aptitude - Quantum CAT : Sarvesh K Verma
5. High School English Grammar & Composition Revised Edition Wren, Martin / S. Chand Publisher
6. How to prepare GRE by Barron's / galgotia publications pvt. Ltd
7. Oxford Guide to English Grammar 01 Edition John Eastwood / Oxford University Press

Website to refer: www.indiabix.com

Note: Each session having 2 Hours

Part I – Aptitude

Session 1:

Lecture

- ° Analogy
- ° Series Completion (Number, Alphabet, Letter Series)
- ° Coding-Decoding for Number, Alphabet and Letter

Session 2:

Lecture

- ° Blood Relations
- ° Puzzle Test
 - Classification Type questions
 - Compression Type questions
 - Sequential order questions
 - Section based on given conditions
 - Questions involving family members

Session 3:

Lecture

- ° Alphabet test
- ° Order of words
- ° Letter words problems

Suggested Teaching Guidelines for
Aptitude & General English – PG-DAC August 2019

- Rule detection
- Alphabetical quibble
- Word formation
- Logical sequence of words

Session 4:

Lecture

- Number, Ranking and time Sequence Test
- Mathematical operations
- Arithmetic reasoning

Session 5:

Lecture

- Logical reasoning
- Statement-Arguments
- Statement-Assumptions

Session 6:

- Statement-courses of Action
- Statement-Conclusions
- Deriving conclusion from passages

Session 7:

- HCF and LCM
- Fraction
- Number system
- Permutation & combination

Session 8:

- Ratio & Preparation
- Partnership

Session 9:

Lecture

- Average
- Percentage

Session 10:

Lecture

- Clock
- Probability

Session 11:

Lecture

- Pipes and cisterns
- Problem on streams

Session 12:

Lecture

- Time and work
- Work and Wages

Session 13:

Lecture

- Problem on Trains

Suggested Teaching Guidelines for
Aptitude & General English – PG-DAC August 2019

- ° Problem on Speed and Velocity

Session 14:

Lecture

- ° Problem on Ages
- ° Profit and loss

Session 15:

Lecture

- ° Simple Interest,
- ° Compound Interest

Part II – General English

Session 16:

Lecture

- ° The Sentence
 - Subject and Predicate
 - Phrase and Clause
 - Parts of Speech
- ° The Noun: Kinds of Nouns
 - The Noun: Gender
 - The Noun: Number
 - The Noun: Case

Session 17:

Lecture

- ° The Adjective
 - Comparison of Adjectives
 - Adjectives Used As Nouns
 - Position Of The Adjectives
 - The Correct Use Of Some Adjectives
- ° Articles
 - Personal Pronouns
 - Reflexive And Emphatic Pronouns
 - Demonstrative, Indefinite And Distributive Pronouns
 - Relative Pronouns
 - Interrogative Pronouns

Session 18:

Lecture

- ° The Verb
 - Verbs Of Incomplete Predication
 - Active And Passive Voice
- ° Mood
 - Tenses: Introduction
 - The Uses of The Present And Past Tenses
 - The Future
 - The Verb: Person And Number
 - The Infinitive

Suggested Teaching Guidelines for
Aptitude & General English – PG-DAC August 2019

- The Participle
- The Gerund
- Irregular Verbs
- Auxiliaries And Modals
- Conjugation of The Verb Love

Session 19:

Lecture

- ° The Adverb
- ° Comparison Of Adverbs
- ° Formation Of Adverbs
- ° Position Of Adverbs
- ° The Preposition
- ° Words Followed By Prepositions
- ° The Conjunction
- ° Some Conjunctions And Their Uses
- ° The Interjection
- ° The Same Word Used As Different Parts Of Speech

Session 20:

Lecture

- ° Composition
- ° Analysis, Transformation and Synthesis
- ° Analysis of Simple Sentences
- ° Phrases
- ° Clauses
- ° Sentences: Simple, Compound and Complex
- ° More about Noun Clauses
- ° More about Adjective Clauses
- ° More about Adverb Clauses

Session 21:

Lecture

- ° Analysis of Complex Sentences
- ° Analysis of Compound Sentences
- ° Transformation of Sentences
- ° Transformation of Sentences
- ° Synthesis of Sentences
- ° Synthesis of Sentences
- ° Synthesis of Sentences
- ° The Sequence of Tenses
- ° Direct and Indirect Speech

Session 22:

Lecture

- ° Agreement of The Verb With The Subject
- ° Nouns and Pronouns
- ° Adjectives
- ° Verbs

Suggested Teaching Guidelines for
Aptitude & General English – PG-DAC August 2019

- Adverbs
- Preposition
- Conjunctions

Session 23:

Lecture

- Order of Words
- Synonyms & Antonyms
- Punctuation
- Spelling Rules
- The Formation of Words
- Figures of Speech Exercise

Session 24:

Lecture:

- Verb Patterns
- Question Tags, Short Answers, Etc.
- More Structures

Session 25:

Lecture

- Words of Idioms & phrases
- Sentence Construction
- Fill up the blanks