

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Introduction to Programming
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	2
<b>Practical</b>	2

### Course description

This course prepare the students to learn the programming with C++. Knowing the C++ and it learns elementary programming techniques with primitive data types, expressions, and operators, selection statements, mathematical functions, characters, and strings, loops, functions, and arrays.

## Course Objectives

1. To understand computer basics, programs, and operating systems.
2. To write C++ programs that perform simple computations.
3. To write expressions using the conditional expressions.
4. To solve mathematics problems by using the C++ mathematical functions.
5. To read/write data from/to a file.
6. To write programs that execute statements repeatedly using the Loop statements.
7. To use many functions that available in C++.
8. To deal with single and multidimensional arrays.

## Course Contents

Unit Number	Unit Name	Contents
1	Introduction to Computers, Programs, and C++	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• What Is a Computer?</li> <li>• Programming Languages</li> <li>• Operating Systems</li> <li>• History of C++</li> <li>• A Simple C++ Program</li> <li>• C++ Program-Development Cycle</li> <li>• Programming Style and Documentation</li> <li>• Programming Errors</li> </ul>
2	Elementary Programming	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Writing a Simple Program</li> <li>• Reading Input from the Keyboard</li> <li>• Identifiers</li> <li>• Variables</li> <li>• Assignment Statements and Assignment Expressions</li> <li>• Named Constants</li> <li>• Numeric Data Types and Operations</li> <li>• Evaluating Expressions and Operator Precedence</li> <li>• Case Study: Displaying the Current Time</li> <li>• Augmented Assignment Operators</li> <li>• Increment and Decrement Operators</li> <li>• Numeric Type Conversions</li> </ul>

		<ul style="list-style-type: none"> <li>• Software Development Process</li> <li>• Case Study: Counting Monetary Units</li> <li>• Common Errors</li> </ul>
3	Selections	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The bool Data Type</li> <li>• if Statements</li> <li>• Two-Way if-else Statements</li> <li>• Nested if and Multi-Way if-else Statements</li> <li>• Common Errors and Pitfalls</li> <li>• Case Study: Computing Body Mass Index</li> <li>• Case Study: Computing Taxes</li> <li>• Generating Random Numbers</li> <li>• Logical Operators</li> <li>• Case Study: Determining Leap Year</li> <li>• Case Study: Lottery</li> <li>• switch Statements</li> <li>• Conditional Expressions</li> <li>• Operator Precedence and Associativity</li> <li>• Debugging</li> </ul>
4	Mathematical Functions, Characters, and Strings	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Mathematical Functions</li> <li>• Character Data Type and Operations</li> <li>• Case Study: Generating Random Characters</li> <li>• Case Study: Guessing Birthdays</li> <li>• Character Functions</li> <li>• Case Study: Converting a Hexadecimal Digit to a Decimal Value</li> <li>• The string Type</li> <li>• Case Study: Revising the Lottery Program Using Strings</li> <li>• Formatting Console Output</li> <li>• Simple File Input and Output</li> </ul>
5	Loops	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The while Loop</li> <li>• The do-while Loop</li> <li>• The for Loop</li> <li>• Which Loop to Use?</li> <li>• Nested Loops</li> <li>• Minimizing Numeric Errors</li> <li>• Case Studies</li> <li>• Keywords break and continue</li> </ul>

		<ul style="list-style-type: none"> <li>• Case Study: Checking Palindromes</li> <li>• Case Study: Displaying Prime Numbers</li> </ul>
6	Functions	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Defining a Function</li> <li>• Calling a Function</li> <li>• void Functions</li> <li>• Passing Arguments by Value</li> <li>• Modularizing Code</li> <li>• Overloading Functions</li> <li>• Function Prototypes</li> <li>• Default Arguments</li> <li>• Inline Functions</li> <li>• Local, Global, and Static Local Variables</li> <li>• Passing Arguments by Reference</li> <li>• Constant Reference Parameters</li> <li>• Case Study: Converting Hexadecimals to Decimals</li> <li>• Function Abstraction and Stepwise Refinement</li> </ul>
7	Single-Dimensional Arrays and C-Strings	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Array Basics</li> <li>• Problem: Lotto Numbers</li> <li>• Problem: Deck of Cards</li> <li>• Passing Arrays to Functions</li> <li>• Preventing Changes of Array Arguments in Functions</li> <li>• Returning Arrays from Functions</li> <li>• Problem: Counting the Occurrences of Each Letter</li> <li>• Searching Arrays</li> <li>• Sorting Arrays</li> <li>• C-Strings</li> </ul>
8	Multidimensional Arrays	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Declaring Two-Dimensional Arrays</li> <li>• Processing Two-Dimensional Arrays</li> <li>• Passing Two-Dimensional Arrays to Functions</li> <li>• Problem: Grading a Multiple-Choice Test</li> <li>• Problem: Finding a Closest Pair</li> <li>• Problem: Sudoku</li> <li>• Multidimensional Arrays</li> </ul>

Text Book:

- Y. Daniel Liang, Introduction to Programming with C++, 3<sup>rd</sup> Edition, Pearson, 2021.

---

References:

1. Hong Lei, C++ Programming for Beginners, Independently Published, 2021.
2. Saurav Sau, C++ for Beginners. 2022.
3. Schaum's Outline of programming With C++, by John Hubbard, McGraw-Hill.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Computer Networks
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course description

Every computer, smartphone or similar electronic device comes with special software called an operating system. In this Course, will be studied the operating systems in computers and smart devices. It is a general introduction to the techniques used in implementing operating systems and the types of related system programs. Topics covered will be the functions and structure of operating systems, process management; deadlock prevention, avoidance, and recovery; Main memory management Virtual memory management, we'll discuss what operating systems are, explain why they're important and explore the different types of operating systems in use today.

### Course Objectives

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication.
3. Understand the Basic functions of operating systems.
4. Explain main components of OS and their working.
5. Understanding of structure of modern OS smart device.
6. Developing students' skills in designing and programming systems which use smart devices.

Course Contents

Unit Number	Unit Name	Contents
1	History Of Operating Systems	<ul style="list-style-type: none"> <li>• The First Generation</li> <li>• The Second Generation</li> <li>• The Third Generation</li> <li>• The Fourth Generation</li> <li>• The Fifth Generation</li> <li>• Mobile Computers</li> </ul>
2	The Operating System ZOO	<ul style="list-style-type: none"> <li>• Mainframe Operating Systems</li> <li>• Server Operating Systems</li> <li>• Multiprocessor Operating Systems</li> <li>• Personal Computer Operating Systems</li> <li>• Handheld Computer Operating Systems</li> <li>• Embedded Operating Systems</li> <li>• Sensor-Node Operating Systems</li> <li>• Real-Time Operating Systems</li> <li>• Smart Card Operating Systems</li> </ul>
3	Computer Hardware Review	<ul style="list-style-type: none"> <li>• Processors</li> <li>• Memory</li> <li>• Disks</li> <li>• I/O Devices</li> <li>• Buses</li> <li>• Booting the Computer</li> </ul>
4	Operating System Concepts	<ul style="list-style-type: none"> <li>• Processes</li> <li>• Address Spaces</li> <li>• Files</li> <li>• Input/Output</li> <li>• Protection</li> <li>• The Shell</li> <li>• Ontogeny Recapitulates Phylogeny</li> </ul>
5	System calls	<ul style="list-style-type: none"> <li>• System Calls for Process Management</li> <li>• System Calls for File Management</li> <li>• System Calls for Directory Management</li> <li>• Miscellaneous System Calls</li> <li>• The Windows Win32 API</li> </ul>
6	Operating system structure	<ul style="list-style-type: none"> <li>• Monolithic Systems</li> </ul>

		<ul style="list-style-type: none"> <li>• Layered Systems</li> <li>• Microkernels</li> <li>• Client-Server Model</li> <li>• Virtual Machines</li> </ul>
7	Mobile Devices & Mobile Operating System	<ul style="list-style-type: none"> <li>• Mobile Devices</li> <li>• Function Of Mobile Operating System</li> <li>• Mobile Operating System</li> <li>• Generalized Mobile Operating System Architecture and Comparison</li> </ul>
8	Android Operating System and iOS	<ul style="list-style-type: none"> <li>• Basics of Android Operating System</li> <li>• Internal Mechanism of Android OS</li> <li>• IOS Operating System</li> </ul>
9	Other Mobile Operating System	<ul style="list-style-type: none"> <li>• Windows Phone</li> <li>• Blackberry</li> <li>• Symbain</li> </ul>

Text Book:

- Modern Operating Systems, 5th edition, Published by Pearson (October 14th 2022) - Copyright © 2023.

References:

1. Mobile operating system Dr. Babasaheb Ambedkar Open University 2019.  
[https://baou.edu.in/assets/pdf/PGDMAD\\_101\\_slm.pdf](https://baou.edu.in/assets/pdf/PGDMAD_101_slm.pdf)
2. Operating System Concepts 8th Edition, by Abraham Silberschatz , Peter B. Galvin , Greg Gagne , 2019.
3. [https://www.toppr.com/guides/computer-science/computer-fundamentals/operating-system/mobile-operating-system/#Features\\_of\\_Mobile\\_Operating\\_System](https://www.toppr.com/guides/computer-science/computer-fundamentals/operating-system/mobile-operating-system/#Features_of_Mobile_Operating_System)
4. <https://none.cs.umass.edu/~shenoy/courses/fall00/syllabus.html>



## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Introduction to Information Security
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course description

This is a graduate-level introductory course in information security. It teaches the basic concepts, principles, and fundamental approaches to secure computers and networks. Its main topics include security basics, security management and risk assessment, software security, operating systems security, database security, cryptography algorithms and protocols, network authentication and secure network applications, malware, network threats and defenses, web security, mobile security, legal and ethical issues, and privacy.

## Course Objectives

This course provides students with a background, foundation, and insight into the subject of information security. This knowledge will serve as basis for future study in selected aspects of this important field or as an important dimension to their effectiveness in the broader computer science field. The primary objectives of the course are:

1. Literacy in information security problems and issue
2. Gain ability to understand technical building blocks for security
3. Acquire background to interact and aid information security staff in both large and small organizations
4. Develop awareness of individual responsibility in maintaining secure environments
5. Appreciate consequences.

## Course Contents

Unit Number	Unit Name	Contents
1	Introduction to Information Security	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The History of Information Security</li> <li>• ?What Is Security</li> <li>• CNSS Security Model</li> <li>• Components of an Information System</li> <li>• Balancing Information Security and Access</li> <li>• Balancing Information Security and Access</li> <li>• Approaches to Information Security Implementation</li> <li>• The Systems Development Life Cycle</li> <li>• The Security Systems Development Life Cycle</li> <li>• Security Professionals and the Organization</li> <li>• Communities of Interest</li> <li>• ?Information Security: Is it an Art or a Science</li> </ul>
2	The Need for Security	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Business Needs First</li> <li>• Threats</li> <li>• Attacks</li> <li>• Secure Software Development</li> </ul>
3	Legal, Ethical, and Professional Issues in	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Law and Ethics in Information Security</li> </ul>

	Information Security	<ul style="list-style-type: none"> <li>• Relevant U.S. Laws</li> <li>• International Laws and Legal Bodies</li> <li>• Ethics and Information Security</li> <li>• Codes of Ethics and Professional Organizations</li> <li>• Key U.S. Federal Agencies</li> </ul>
4	Risk Management	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• An Overview of Risk Management</li> <li>• Risk Identification</li> <li>• Risk Assessment</li> <li>• Risk Control Strategies</li> <li>• Selecting a Risk Control Strategy</li> <li>• Quantitative Versus Qualitative Risk Control Practices</li> <li>• Risk Management Discussion Points</li> <li>• Recommended Risk Control Practice</li> </ul>
5	Physical Security	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Physical Access Controls</li> <li>• Fire Security and Safety</li> <li>• Failure of Supporting Utilities and Structural Collapse</li> <li>• Interception of Data</li> <li>• Mobile and Portable Systems</li> <li>• Special Considerations for Physical Security</li> </ul>
7	Cryptography	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Foundations of Cryptology</li> <li>• Cipher Methods</li> <li>• Cryptographic Algorithms</li> <li>• Cryptographic Tools</li> <li>• Protocols for Secure Communications</li> <li>• Attacks on Cryptosystems</li> </ul>

Text Book:

- Whitman, M.E. and Mattord, H.J., 2017. Principles of information security. Cengage Learning

References:

1. Charles P. Pfleeger and Shari Lawrence Pfleeger. Security in Computing, Fourth Edition. Prentice Hall, 2007. ISBN 0-13-239077-9.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Multimedia
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	1
<b>Practical</b>	4

### Course description

This course includes: Introduction and Multimedia Data Representations, Multimedia Data Compression, Multimedia Communications and Networking, Human-Centric Interactive Multimedia.

### Course Objectives

1. Introducing the fundamental elements of multimedia.
2. Outline some effects to keep in mind for presenting multimedia content.
3. Identifying the images, starting with 1-bit images, then 8-bit grayscale images and how to print them, then 24-bit color images and 8-bit versions of color images.
4. Identifying on the Color science, Color models in images and Color models in video.
5. Identifying on video and how they impact multimedia application.
6. Introducing the basics of digital audio.

### Course Contents

Unit Number	Unit Name	Contents
1	Introduction to Multimedia	<ul style="list-style-type: none"> <li>• What is Multimedia? (Components of Multimedia)</li> <li>• Multimedia: Past and Present (Early History of Multimedia, Hypermedia, WWW, and Internet, Multimedia in the New Millennium)</li> <li>• Multimedia Software Tools: A Quick Scan (Music Sequencing and Notation, Digital Audio, Graphics and Image Editing, Video Editing, Animation, Multimedia Authoring, Multimedia Broadcasting)</li> <li>• The Future of Multimedia</li> </ul>
2	A Taste of Multimedia	<ul style="list-style-type: none"> <li>• Multimedia Tasks and Concerns</li> <li>• Multimedia Presentation</li> <li>• Data Compression</li> <li>• Multimedia Production</li> <li>• Multimedia Sharing and Distribution</li> <li>• Some Useful Editing and Authoring Tools (Adobe Premiere, HTML5 Canvas, Adobe Director, Adobe XD)</li> </ul>
3	Graphics and Image Data Representations	<ul style="list-style-type: none"> <li>• Graphics and Image Data Types (1-Bit Images, 8-Bit Gray-Level Images, Image Data Types, 24-Bit Color Images, Higher Bit-Depth Images, 8-Bit Color Images, Color Lookup Tables (LUTs))</li> <li>• Popular File Formats (GIF, JPEG, PNG, TIFF, Windows BMP, Windows WMF, Netpbm Format, EXIF, HEIF, PS and PDF, PTM)</li> </ul>
4	Color in Image and Video	<ul style="list-style-type: none"> <li>• Color Science (Light and Spectra, Human Vision, Spectral Sensitivity of the Eye, Image Formation, Camera Systems, Gamma Correction, Color-Matching Functions, CIE Chromaticity Diagram, Color Monitor Specifications, Out-of-Gamut Colors, White Point Correction, XYZ to RGB Transform, Transform with Gamma Correction, L*a*b* (CIELAB) Color Model, More Color Coordinate Schemes, Munsell Color Naming System)</li> <li>• Color Models in Images (RGB Color Model for Displays, Multi-sensor Cameras, Camera-Dependent Color, Subtractive Color: CMY Color Model, Transformation from RGB to CMY, Undercolor Removal: CMYK System, Printer Gamuts, Multi-ink Printers)</li> </ul>

		<ul style="list-style-type: none"> <li>• Color Models in Video (Video Color Transforms, YUV Color Model, YIQ Color Model, YCbCr Color Model)</li> </ul>
5	Fundamental Concepts in Video	<ul style="list-style-type: none"> <li>• Analog Video (NTSC Video, PAL Video, SECAM Video)</li> <li>• Digital Video (Chroma Subsampling, CCIR and ITU-R Standards for Digital Video, High Definition TV (HDTV), Ultra-High-Definition TV (UHDTV))</li> <li>• Video Display Interfaces (Analog Display Interfaces, Digital Display Interfaces)</li> <li>• 360 Video (Equirectangular Projection (ERP), Other Projections)</li> <li>• 3D Video and TV (Cues for 3D Percept, 3D Camera Models, 3D Movie and TV Based on Stereo Vision, The Vergence–Accommodation Conflict, Autostereoscopic (Glasses-Free) Display Devices, Disparity Manipulation in 3D Content Creation)</li> <li>• Video Quality Assessment (VQA) (Objective Assessment, Subjective Assessment, Other VQA Metrics)</li> </ul>
6	Basics of Digital Audio	<ul style="list-style-type: none"> <li>• Digitization of Sound (What Is Sound?, Digitization, Nyquist Theorem, Signal-to-Noise Ratio (SNR), Signal-to-Quantization-Noise Ratio (SQNR), Linear and Nonlinear Quantization, Audio Filtering, Audio Quality versus Data Rate, Synthetic Sounds)</li> <li>• MIDI: Musical Instrument Digital Interface (MIDI Overview, Hardware Aspects of MIDI, Structure of MIDI Messages, MIDI-to-WAV Conversion, General MIDI, MIDI 2.0)</li> <li>• Quantization and Transmission of Audio (Coding of Audio, Pulse Code Modulation, Differential Coding of Audio, Lossless Predictive Coding, DPCM, DM, ADPCM)</li> </ul>

Text Book:

- Ze-Nian Li Mark S. Drew Jiangchuan Liu, Fundamentals of Multimedia, 3<sup>rd</sup> edition, Springer, 2021.

References:

1. Tay Vaughan, Multimedia: Making It Work, Eighth Edition, Mc Graw Hill, 2011
2. Vic Costello, Multimedia Foundations, 2nd edition, Routledge, 2016.

- 
3. T. M. Savage and K. E. Vogel, An Introduction To Digital Multimedia, 2<sup>nd</sup> edition, Jones & Bartlett Learning 2013.



## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Principles of Database
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	2
<b>Practical</b>	2

### Course description

This course introduces you to the major concepts, methodologies, tools and techniques that are required to analyse, design, and develop well-structured databases for modern organisations. Data modelling using entity-relationship diagrams is taught and applied. You will then use a DBMS to gain an appreciation of the concepts and practical application of database management systems. SQL is covered to complete the cycle of professional practice.

### Course Objectives

1. Understand the fundamentals of relational, object-oriented, and distributed database systems including data models, database architectures, and database manipulations.
2. Understand the theories and techniques in developing database applications and be able to demonstrate the ability to build databases using enterprise DBMS products such as Oracle or SQL Server.
3. Be familiar with managing database systems.
4. Understand new developments and trends in databases functional behavior.
5. Learning the DML, DCL, and Queries in SQL.

### Course Contents

Unit Number	Unit Name	Contents
1	Database System	<ul style="list-style-type: none"> <li>• Why Database?</li> <li>• Data versus information</li> <li>• Introducing the database?</li> <li>• Role and Advantages of the DBMS</li> <li>• Types of Databases</li> <li>• Why Database design is important</li> <li>• Evolution of File System Data Processing</li> <li>• Manual File Systems</li> <li>• Computerized File Systems</li> <li>• File System Redux: Modern End-User Products</li> <li>• Problem with File System Data Processing</li> <li>• Structural data Dependence</li> <li>• Data Redundancy</li> <li>• Data Anomalies</li> <li>• Database Systems</li> <li>• The Database System Environment</li> <li>• DBMS functions</li> <li>• Managing the Database system</li> </ul>
2	Data Models	<ul style="list-style-type: none"> <li>• Data Modeling and Data Models</li> <li>• The Importance of Data Models</li> <li>• Data Model Basic Building Blocks</li> <li>• Business Rules</li> </ul>

		<ul style="list-style-type: none"> <li>• Discovering Business Rules</li> <li>• Translating Business Rules into Data Model Components</li> <li>• Naming Conventions</li> <li>• The Evolution of Data Models</li> <li>• Hierarchical and Network Models</li> <li>• The Relational Model</li> <li>• The Entity Relationship Model</li> <li>• The Object-Oriented Model</li> <li>• Object/Relational and XML</li> <li>• Emerging Data Models: Big Data and NoSQL</li> <li>• Data Models: A Summary</li> <li>• Degrees of Data Abstraction 54</li> <li>• The External Model 57</li> <li>• The Conceptual Model 58</li> <li>• The Internal Model 59</li> <li>• The Physical Model 60</li> </ul>
3	The Relational Database Model	<ul style="list-style-type: none"> <li>• A Logical View of Data 69</li> <li>• Tables and Their Characteristics 69</li> <li>• Keys</li> <li>• Dependencies</li> <li>• Types of Keys</li> <li>• Integrity Rules</li> <li>• Relational Algebra</li> <li>• Formal Definitions and Terminology</li> <li>• Relational Set Operators</li> <li>• The Data Dictionary and the System Catalog</li> <li>• Relationships within the Relational Database</li> <li>• The 1:M Relationship</li> <li>• The 1:1 Relationship</li> <li>• The M:N Relationship</li> <li>• Data Redundancy Revisited</li> <li>• Indexes</li> <li>• Codd's Relational Database Rules Functional dependencies and normalization</li> <li>• Normalization techniques (1NF, 2NF, 3NF)</li> </ul>
4	Entity Relationship (ER) Modeling	<ul style="list-style-type: none"> <li>• The Entity Relationship Model</li> <li>• Entities</li> <li>• Attributes</li> <li>• Relationships</li> <li>• Connectivity and Cardinality</li> <li>• Existence Dependence</li> </ul>

		<ul style="list-style-type: none"> <li>• Relationship Strength</li> <li>• Weak Entities</li> <li>• Relationship Participation</li> <li>• Relationship Degree</li> <li>• Recursive Relationships</li> <li>• Associative (Composite) Entities</li> <li>• Developing an ER Diagram</li> <li>• Database Design Challenges: Conflicting Goals</li> </ul>
6	Normalization of Database Tables	<ul style="list-style-type: none"> <li>• Database Tables and Normalization</li> <li>• The Need for Normalization</li> <li>• The Normalization Process</li> <li>• Conversion to First Normal Form (1NF)</li> <li>• Conversion to Second Normal Form (2NF)</li> <li>• Conversion to Third Normal Form (3NF)</li> <li>• Improving the Design</li> <li>• Surrogate Key Considerations</li> <li>• Higher-Level Normal Forms</li> <li>• The Boyce-Codd Normal Form</li> <li>• Fourth Normal Form (4NF)</li> <li>• Normalization and Database Design</li> <li>• Denormalization</li> <li>• Data-Modeling Checklist</li> </ul>
7	Introduction to Structured Query Language (SQL)	<ul style="list-style-type: none"> <li>• Introduction to SQL</li> <li>• Data Types</li> <li>• SQL Queries</li> <li>• The Database Model</li> <li>• Basic SELECT Queries</li> <li>• SELECT Statement Options</li> <li>• Using Column Aliases</li> <li>• Using Computed Columns</li> <li>• Arithmetic Operators: The Rule of Precedence</li> <li>• Date Arithmetic</li> <li>• Listing Unique Values</li> <li>• FROM Clause Options</li> <li>• Natural Join</li> <li>• JOIN USING Syntax</li> <li>• JOIN ON Syntax</li> <li>• Common Attribute Names</li> <li>• Outer Joins</li> <li>• Cross Join</li> <li>• Joining Tables with an Alias</li> <li>• Recursive Joins</li> </ul>

		<ul style="list-style-type: none"> <li>• ORDER BY Clause Options</li> <li>• WHERE Clause Options</li> <li>• Selecting Rows with Conditional Restrictions Using Comparison Operators on Character Attributes</li> <li>• Using Comparison Operators on Dates</li> <li>• Logical Operators: AND, OR, and NOT</li> <li>• Old-Style Joins</li> <li>• Special Operators</li> <li>• Aggregate Processing</li> <li>• Aggregate Functions</li> <li>• Grouping Data</li> <li>• HAVING Clause</li> <li>• Subqueries</li> <li>• WHERE Subqueries</li> <li>• IN Subqueries</li> <li>• HAVING Subqueries</li> <li>• Multirow Subquery Operators: ALL and ANY</li> <li>• FROM Subqueries</li> <li>• Attribute List Subqueries</li> <li>• Correlated Subqueries</li> <li>• SQL Functions</li> <li>• Date and Time Functions</li> <li>• Numeric Functions</li> <li>• String Functions</li> <li>• Conversion Functions</li> <li>• Relational Set Operators</li> <li>• UNION</li> <li>• UNION ALL</li> <li>• INTERSECT</li> <li>• EXCEPT (MINUS)</li> <li>• Syntax Alternatives</li> <li>• Crafting SELECT Queries</li> <li>• Know Your Data</li> <li>• Know the Problem</li> <li>• Build One Clause at a Time</li> </ul>
8	Advanced SQL	<ul style="list-style-type: none"> <li>• Data Definition Commands</li> <li>• Starting Database Model</li> <li>• Creating the Database</li> <li>• The Database Schema</li> <li>• Data Types</li> <li>• Creating Table Structures</li> <li>• CREATE TABLE command</li> </ul>

		<ul style="list-style-type: none"> <li>• SQL Constraints</li> <li>• Create a Table with a SELECT Statement</li> <li>• SQL Indexes</li> <li>• Altering Table Structures</li> <li>• Changing a Column's Data Type</li> <li>• Changing a Column's Data Characteristics</li> <li>• Adding a Column</li> <li>• Adding Primary Key, Foreign Key, and Check Constraints</li> <li>• Dropping a Column</li> <li>• Deleting a Table from the Database</li> <li>• Data Manipulation Commands</li> <li>• Adding Table Rows</li> <li>• Inserting Table Rows with a SELECT Subquery</li> <li>• Saving Table Changes</li> <li>• Updating Table Rows</li> <li>• Deleting Table Rows</li> <li>• Restoring Table Contents</li> <li>• Virtual Tables: Creating a View</li> <li>• Updatable Views</li> <li>• Sequences</li> <li>• Procedural SQL</li> <li>• Triggers</li> <li>• Stored Procedures</li> <li>• PL/SQL Processing with Cursors</li> <li>• PL/SQL Stored Functions</li> <li>• Embedded SQL</li> </ul>
--	--	---

Text Book:

- Carols Coronel, Steven Morris, Database Systems (Design, Implementation, and Management), 13th edition, 2019.

References:

1. C. J. Date, "Database Systems", Addison Wesley, 8th edition
2. Elmasri, et al (2011). Fundamentals of Database Systems, 6th ed., Pearson education References
3. David M. Kroenke. (1998). Database processing, 6th ed. Prentice Hall
4. Introduction to Database systems, C.J. DATE3 Navathe, E (2000).

- 
5. Fundamentals of database Systems. 3rd ed. Delhi, Pearson Education4 Ramon A, etal. Shaum's outlines, fundamentals of relational databases5 Silbershatz A. Korth H & Sundarshan (2006).
  6. Database System concepts, 5th ed. Boston, McGraw Hill.
  7. Thomas M. Connolly and Carolyn E. beg (2004). A step-by-step approach to building databases, 2nd ed. Pearson Education Limited.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Database application Programming
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	0
<b>Practical</b>	6

### Course description

This course aims to teach the student how to create his project using Oracle forms developer using database 12 c and oracle developer 10g.

### Course Objectives



1. Develop database and websheet applications
2. Deploy the Application as a Packaged Application
3. Debug your application
4. Utilize dynamic components such as dynamic actions and plugins
5. Manage Access Control by authorizing and authenticating users
6. Utilize Team Development to drive the application development process
7. Utilize and Manage Shared Components in an Application
8. Add Page Processing components
9. Manage database objects
10. Administer Oracle Application Express Workspaces
11. Load and Unload data or application components
12. Manage users and groups
13. Build queries of database objects and incorporate in an application
14. Use and manipulate reports and forms in Oracle Application Express

## Course Contents

Unit Number	Unit Name	Contents
1	Introduction to oracle forms developer and oracle form service	<ul style="list-style-type: none"> <li>• Internet computing solution</li> <li>• Plugging into the grid</li> <li>• Oracle enterprise grid computing</li> <li>• Oracle 10g product and forms development</li> <li>• oracle application server 10g architecture and components</li> <li>• benefits of component of oracle developer suite 10g</li> <li>•</li> </ul>
2	Running a form developer application	<ul style="list-style-type: none"> <li>• Testing a form: OC4J/starting OC4J</li> <li>• Running a form/Browser</li> <li>• The java Runtime environment</li> <li>• Starting a runtime session</li> <li>• Starting a run-time session</li> <li>• The runtime engine</li> <li>• Navigating a forms developer application</li> <li>• Modes of operation</li> <li>• Retrieving data</li> </ul>

		<ul style="list-style-type: none"> <li>Retrieving restricted data</li> <li>Query/where dialog box</li> <li>Insert, updating and deleting</li> </ul>
3	Working in the forms developer environment	<ul style="list-style-type: none"> <li>Forms builder key features</li> <li>Forms builder components (object navigator, property palette, layout editor, pl/sql editor, and menu structure)</li> <li>Blocks, items, and canvases</li> <li>Navigation in block</li> <li>Data blocks</li> <li>Forms and data blocks</li> <li>Form module hierarchy</li> <li>Customizing your forms builder session</li> <li>Saving preference</li> <li>Using the online help system</li> <li>Form developer executables</li> <li>Form developer module types</li> <li>Defining forms environment variables for run time</li> <li>Defining forms environment variables for design time</li> <li></li> <li></li> </ul>
4	Creating a basic form module	<ul style="list-style-type: none"> <li>Creating a new form module</li> <li>Form module properties</li> <li>Creating new data block</li> <li>Navigating the wizards</li> <li>Lunching the data block wizard</li> <li>Data block wizard (type page, table page, and finish page)</li> <li>Layout wizard (items page, style page, and rows page)</li> <li>Data block functionality</li> <li>Template forms</li> <li>Saving a form module</li> <li>Compiling a form module</li> <li>Module types and storage format</li> </ul>
6	Creating a master-detailed form	<ul style="list-style-type: none"> <li>Form block relationships</li> <li>Data block wizard: master – detailed page</li> <li>Relation object</li> <li>Creating a relation manually</li> <li>Join condition</li> <li>Deletion properties</li> <li>Modifying a relation</li> </ul>

		<ul style="list-style-type: none"> <li>• Coordination properties</li> <li>• Running a master-detail form module</li> <li>• Modifying the structure of a data block</li> <li>• Modifying the layout of a data block</li> </ul>
7	Working with data block and frames	<ul style="list-style-type: none"> <li>• Managing object properties</li> <li>• Displaying the property palette</li> <li>• Property palette: features</li> <li>• Property controls</li> <li>• Visual attributes</li> <li>• How to use visual attributes</li> <li>• Font, pattern, and color pickers</li> <li>• Controlling data block behavior and appearance</li> <li>• Navigation properties</li> <li>• Records properties</li> <li>• Database properties</li> <li>• Scroll bar properties</li> <li>• Controlling frame properties</li> <li>• Displaying multiple property palettes</li> <li>• setting properties on multiple objects</li> <li>• Copying properties</li> <li>• Creating a control block</li> <li>• Delete a data block</li> </ul>
8	Working with text items	<ul style="list-style-type: none"> <li>• Text item overview</li> <li>• Creating a text item</li> <li>• Modifying the appearance of a text item (general and physical properties, records properties, font and color properties, and prompts)</li> <li>• Associating text with an item prompt</li> <li>• Controlling the data of a text item (format, values, copy value from item, and synchronize with item)</li> <li>• Altering navigational behavior of text items</li> <li>• Enhancing the relationship between text item and database</li> <li>• Adding functionality to a text item (conceal data property, keyboard navigable and enabled, and multi-line text item)</li> <li>• Display helpful messages: help properties</li> <li>•</li> </ul>
9	Creating LOV's and Editor	<ul style="list-style-type: none"> <li>• LOVs and Record Groups</li> <li>• Creating an LOV Manually</li> <li>• Creating an LOV with the LOV Wizard: (SQL Query Page, Column Selection Page, Column Properties</li> </ul>

		<p>Page, Display Page, Advanced Properties Page, and Assign to Item Page)</p> <ul style="list-style-type: none"> <li>• LOV Properties</li> <li>• Setting LOV Properties</li> <li>• LOVs: Column Mapping</li> <li>• Defining an Editor</li> <li>• Setting Editor Properties</li> <li>• Associating an Editor with a Text Item</li> </ul>
10	Creating Additional Input Items	<ul style="list-style-type: none"> <li>• Input Items overview</li> <li>• Check Boxes Overview</li> <li>• Creating a Check Box</li> <li>• Converting an Existing item into a Check Box</li> <li>• Creating a Check Box in the Layout Editor</li> <li>• Setting Check Box in the Layout Editor</li> <li>• List Items Overview</li> <li>• Creating a List Item</li> <li>• Creating a Radio Group</li> <li>• Converting Existing Item to Radio Group</li> <li>• Creating Radio Group in Layout Editor</li> <li>• Setting Radio Properties</li> <li>• Radio Group Mapping of Other Values</li> </ul>
11	Creating Noninput Items	<ul style="list-style-type: none"> <li>• Noninput Items Overview</li> <li>• Display Items</li> <li>• Creating a Display Item</li> <li>• Image Items</li> <li>• Image File Formats</li> <li>• Creating an Image Item</li> <li>• Setting Image-Specific Item Properties</li> <li>• Push Buttons</li> <li>• Push Button Actions</li> <li>• Creating a Push Button</li> <li>• Setting Push Button Properties</li> <li>• Calculated Items</li> <li>• Creating a Calculated Item by Setting Properties</li> <li>• Setting Item Properties for the Calculated Item</li> <li>• Summary Functions</li> <li>• Calculated Item Based on a Formula</li> <li>• Rules for Calculated Item Formulas</li> <li>• Calculated Item Based on a Summary</li> <li>• Rules for Summary Items</li> <li>• Creating a Hierarchical Tree Item</li> <li>• Setting Hierarchical Tree Item Properties</li> </ul>

		<ul style="list-style-type: none"> <li>• Bean Area Items</li> <li>• Creating a Bean Area Item</li> <li>• Setting Bean Area Item Properties</li> <li>• The JavaBean at Run Time</li> </ul>
12	Creating Windows and Content Canvases	<ul style="list-style-type: none"> <li>• Windows and Canvases</li> <li>• Window, Canvas, and Viewport</li> <li>• The Content Canvas</li> <li>• Relationship Between Windows and Content Canvases</li> <li>• The Default Window</li> <li>• Displaying a Form Module in Multiple Windows</li> <li>• Creating a New Window</li> <li>• Setting Window Properties</li> <li>• GUI Hints</li> <li>• Displaying a Form Module on Multiple Layouts</li> <li>• Creating a New Content Canvas</li> <li>• Setting Content Canvas Properties</li> </ul>
13	Working with Other Canvas Types	<ul style="list-style-type: none"> <li>• Overview of Canvas Types</li> <li>• The Stacked Canvas</li> <li>• Creating a Stacked Canvas</li> <li>• Setting Stacked Canvas Properties</li> <li>• The Toolbar Canvas</li> <li>• The MDI Toolbar</li> <li>• Creating a Toolbar Canvas</li> <li>• Setting Toolbar Properties</li> <li>• The Tab Canvas</li> <li>• Creating a Tab Canvas</li> <li>• Creating a Tab Canvas in the Object Navigator</li> <li>• Creating a Tab Canvas in the Layout Editor</li> <li>• Setting Tab Canvas, Tab Page, and Item Properties</li> <li>• Placing Items on a Tab Canvas</li> </ul>
14	Introduction to Triggers	<ul style="list-style-type: none"> <li>• Trigger Overview</li> <li>• Grouping Triggers into Categories</li> <li>• Defining Trigger Components</li> <li>• Trigger Type</li> <li>• Trigger Code</li> <li>• Trigger Scope</li> <li>• Specifying Execution Hierarchy</li> </ul>
15	Producing Triggers	<ul style="list-style-type: none"> <li>• Creating Triggers in Forms Builder</li> <li>• Creating a Trigger</li> <li>• Setting Trigger Properties</li> <li>• PL/SQL Editor Features</li> </ul>

		<ul style="list-style-type: none"> <li>• The Database Trigger Editor</li> <li>• Writing Trigger Code</li> <li>• Using Variables in Triggers</li> <li>• Forms Builder Variables</li> <li>• Adding Functionality with Built-In Subprograms</li> <li>• Limits of Use</li> <li>• Using Built-In Definitions</li> <li>• Useful Built-Ins</li> <li>• Using Triggers: When-Button-Pressed Trigger</li> <li>• Using Triggers: When-Window-Closed Trigger</li> </ul>
16	Debugging Triggers	<ul style="list-style-type: none"> <li>• The Debugging Process</li> <li>• The Debug Console</li> <li>• The Debug Console: Stack Panel</li> <li>• The Debug Console: Variables Panel</li> <li>• The Debug Console: Watch Panel</li> <li>• The Debug Console: Form Values Panel</li> <li>• The Debug Console: PL/SQL Packages Panel</li> <li>• The Debug Console: Global/System Variables Panel</li> <li>• The Debug Console: Breakpoints Panel</li> <li>• The Debug Console</li> <li>• Setting Breakpoints in Client Code</li> <li>• Setting Breakpoints in Stored Code</li> <li>• Debugging Tips</li> <li>• Running a Form in Debug Mode</li> <li>• Stepping Through Code</li> <li>• Debug Example</li> </ul>
17	Adding Functionality to Items	<ul style="list-style-type: none"> <li>• Item Interaction Triggers</li> <li>• Coding Item Interaction Triggers</li> <li>• Interacting with Check Boxes</li> <li>• Changing List Items at Run Time</li> <li>• Displaying LOVs from Buttons</li> <li>• LOVs and Buttons</li> <li>• Populating Image Items</li> <li>• Loading the Right Image</li> <li>• Populating Hierarchical Trees</li> <li>• Displaying Hierarchical Trees</li> <li>• Interacting with JavaBeans</li> </ul>
18	Run Time Messages and Alerts	<ul style="list-style-type: none"> <li>• Run-Time Messages and Alerts Overview</li> <li>• Detecting Run-Time Errors</li> <li>• Errors and Built-Ins</li> <li>• Message Severity Levels</li> </ul>

		<ul style="list-style-type: none"> <li>• Suppressing Messages</li> <li>• The FORM_TRIGGER_FAILURE Exception</li> <li>• Triggers for Intercepting System Messages</li> <li>• Handling Informative Messages</li> <li>• Setting Alert Properties</li> <li>• Planning Alerts</li> <li>• Controlling Alerts</li> <li>• SHOW_ALERT Function</li> <li>• Directing Errors to an Alert</li> <li>• Causes of Oracle Server Errors</li> <li>• Trapping Server Errors</li> </ul>
19	Query Triggers	<ul style="list-style-type: none"> <li>• Query Processing Overview</li> <li>• SELECT Statements Issued During Query Processing</li> <li>• WHERE Clause</li> <li>• ONETIME_WHERE Property</li> <li>• ORDER BY Clause</li> <li>• Writing Query Triggers: Pre-Query Trigger</li> <li>• Writing Query Triggers: Post-Query Trigger</li> <li>• Writing Query Triggers: Using SELECT Statements in Triggers</li> <li>• Query Array Processing</li> <li>• Coding Triggers for Enter-Query Mode</li> <li>• Overriding Default Query Processing</li> <li>• Obtaining Query Information at Run Time</li> </ul>
20	Validation	<ul style="list-style-type: none"> <li>• The Validation Process</li> <li>• Controlling Validation Using Properties: Validation Unit</li> <li>• Controlling Validation Using Properties: Validate from List</li> <li>• Controlling Validation Using Triggers</li> <li>• Example: Validating User Input</li> <li>• Using Client-Side Validation</li> <li>• Tracking Validation Status 1</li> <li>• Controlling When Validation Occurs with Built-Ins</li> </ul>
21	Navigation	<ul style="list-style-type: none"> <li>• Navigation Overview</li> <li>• Understanding Internal Navigation</li> <li>• Using Object Properties to Control Navigation</li> <li>• Mouse Navigate Property</li> <li>• Writing Navigation Triggers</li> <li>• Navigation Triggers</li> <li>• When-New-&lt;object&gt;-Instance Triggers</li> <li>• SET_&lt;object&gt;_PROPERTY Examples</li> </ul>

		<ul style="list-style-type: none"> <li>• The Pre- and Post-Triggers</li> <li>• Post-Block Trigger Example</li> <li>• The Navigation Trap</li> <li>• Using Navigation Built-Ins in Triggers</li> </ul>
22	Transaction Processing	<ul style="list-style-type: none"> <li>• Transaction Processing Overview</li> <li>• The Commit Sequence of Events</li> <li>• Characteristics of Commit Triggers</li> <li>• Common Uses for Commit Triggers</li> <li>• Life of an Update</li> <li>• Delete Validation</li> <li>• Assigning Sequence Numbers</li> <li>• Keeping an Audit Trail</li> <li>• Testing the Results of Trigger DML</li> <li>• DML Statements Issued During Commit Processing</li> <li>• Overriding Default Transaction Processing</li> <li>• Running Against Data Sources Other than Oracle</li> <li>• Getting and Setting the Commit Status</li> <li>• Array DML</li> <li>• Effect of Array DML on Transactional Triggers</li> <li>• Implementing Array DML</li> </ul>
23	Transaction Processinge	<ul style="list-style-type: none"> <li>• What Is Flexible Code?</li> <li>• Using System Variables for Current Context</li> <li>• System Status Variables</li> <li>• GET_&lt;object&gt;_PROPERTY Built-Ins</li> <li>• SET_&lt;object&gt;_PROPERTY Built-Ins</li> <li>• Referencing Objects by Internal ID</li> <li>• FIND_ Built-Ins</li> <li>• Using Object IDs</li> <li>• Increasing the Scope of Object IDs</li> <li>• Referencing Objects Indirectly</li> </ul>
24	Sharing Objects and Code	<ul style="list-style-type: none"> <li>• Benefits of Reusing Objects and</li> <li>• What Are Property Classes?</li> <li>• Creating a Property Class</li> <li>• Inheriting from a Property Class</li> <li>• What Are Object Groups?</li> <li>• Creating and Using Object Groups</li> <li>• Copying and Subclassing Objects and Code</li> <li>• Subclassing</li> <li>• What Are Object Libraries?</li> <li>• Benefits of the Object Library</li> <li>• Working with Object Libraries</li> </ul>



		<ul style="list-style-type: none"> <li>• What Is a SmartClass</li> <li>• Working with SmartClasses</li> <li>• Reusing PL/SQL</li> <li>• What Are PL/SQL Libraries</li> <li>• Writing Code for Libraries</li> <li>• Creating Library Program Units</li> <li>• Attach Library Dialog Box</li> <li>• Calls and Searches</li> </ul>
25	Using WebUtil to Interact with the Client	<ul style="list-style-type: none"> <li>• WebUtil Overview</li> <li>• Benefits of the WebUtil Utility</li> <li>• Integrating WebUtil into a Form</li> <li>• When to Use WebUtil Functionality</li> <li>• Interacting with the Client</li> <li>• Example: Opening a File Dialog on the Client</li> <li>• Example: Reading an Image File into Forms from the Client</li> <li>• Example: Writing Text Files on the Client</li> <li>• Example: Executing Operating System Commands on the Client</li> <li>• Example: Performing OLE Automation on the Client</li> <li>• Example: Obtaining Environment Information about the Client</li> </ul>

Text Book:

- Oracle Forms Developer 10g: Build Internet Application, First Edition, 2004.

References:

1. C. J. Date, "Database Systems", Addison Wesley, 8th edition
2. Elmasri, et al (2011). Fundamentals of Database Systems, 6th ed., Pearson education References
3. David M. Kroenke. (1998). Database processing, 6th ed. Prentice Hall
4. Introduction to Database systems, C.J. DATE3 Navathe, E (2000).
5. Fundamentals of databaseSystems. 3rd ed. Delhi, Pearson Education4 Ramon A, etal. Shaum's outlines,fundamentals of relational databases5 Silbershatz A. Korth H &Sundarshan (2006).
6. Database System concepts, 5th ed. Boston, McGraw Hill.
7. Thomas M. Connolly and Carolyn E. beg (2004). A step-by-step approach to building databases, 2nd ed. Pearson Education Limited.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Websites Development
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	1
<b>Practical</b>	4

### Course description

This course will introduce the essential topics of Internet Programming predominately with HTML programming languages. Students will develop software that manipulates different forms of data such as hypertext, graphics, video, and sound. Advanced interactive/executable web pages will be developed. Cascading Style Sheets (CSS) will be introduced for describing the presentation of Web pages, such as colors, layout, position elements, and fonts. Moreover, students will design and write interactive WWW pages using JavaScript.

## Course Objectives

1. Describe the essential topics of Internet Programming predominantly with HTML programming languages.
2. Describe Cascading Style Sheets (CSS).
3. Describe the basics of working with objects in JavaScript.
4. Understand important components of HTML5 documents.
5. Build a form using the new HTML5 input types.
6. Control a website's appearance with style sheets.
7. Write a JavaScript programs.

## Course Contents

Unit Number	Unit Name	Contents
1	Introduction to Computers and the Internet	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The Internet in Industry and Research</li> <li>• HTML5, CSS3, JavaScript, Canvas and jQuery</li> <li>• Demos</li> <li>• Evolution of the Internet and World Wide Web</li> <li>• Web Basics</li> <li>• Multitier Application Architecture</li> <li>• Client-Side Scripting versus Server-Side Scripting</li> <li>• World Wide Web Consortium (W3C)</li> <li>• Web 2.0: Going Social</li> <li>• Data Hierarchy</li> <li>• Operating Systems</li> <li>• Desktop and Notebook Operating Systems</li> <li>• Mobile Operating Systems</li> <li>• Types of Programming Languages</li> <li>• Object Technology</li> <li>• Keeping Up-to-Date with Information Technologies</li> </ul>
2	Introduction to HTML5: Part 1	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Editing HTML5</li> <li>• First HTML5 Example</li> <li>• W3C HTML5 Validation Service</li> <li>• Headings</li> <li>• Linking</li> <li>• Images</li> </ul>

		<ul style="list-style-type: none"> <li>• alt Attribute</li> <li>• Void Elements</li> <li>• Using Images as Hyperlinks</li> <li>• Special Characters and Horizontal Rules</li> <li>• Lists</li> <li>• Tables</li> <li>• Forms</li> <li>• Internal Linking</li> <li>• meta Elements</li> <li>• Web Resources</li> </ul>
3	Introduction to HTML5: Part 2	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• New HTML5 Form input Types</li> <li>• input Type color</li> <li>• input Type date</li> <li>• input Type datetime</li> <li>• input Type datetime-local</li> <li>• input Type email</li> <li>• input Type month</li> <li>• input Type number</li> <li>• input Type range</li> <li>• input Type search</li> <li>• input Type tel</li> <li>• input Type time</li> <li>• input Type url</li> <li>• input Type week</li> <li>• input and datalist Elements and autocomplete Attribute</li> <li>• input Element autocomplete Attribute</li> <li>• datalist Element</li> <li>• Page-Structure Elements</li> <li>• header Element</li> <li>• nav Element figure Element and figcaption Element</li> <li>• article Element</li> <li>• summary Element and details Element</li> <li>• section Element</li> <li>• aside Element</li> <li>• meter Element</li> <li>• footer Element</li> <li>• Text-Level Semantics: mark Element and wbr Element</li> </ul>

4	Introduction to Cascading Style Sheets™ (CSS): Part 1	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Inline Styles Embedded Style Sheets</li> <li>• Conflicting Styles</li> <li>• Linking External Style Sheets</li> <li>• Positioning Elements: Absolute Positioning, z-index</li> <li>• Positioning Elements: Relative Positioning, span</li> <li>• Backgrounds</li> <li>• Element Dimensions</li> <li>• Box Model and Text Flow</li> <li>• Media Types and Media Queries</li> <li>• Drop-Down Menus</li> <li>• (Optional) User Style Sheets</li> <li>• Web Resources</li> </ul>
5	Introduction to Cascading Style Sheets™ (CSS): Part 2	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Text Shadows</li> <li>• Rounded Corners</li> <li>• Color</li> <li>• Box Shadows</li> <li>• Linear Gradients; Introducing Vendor Prefixes</li> <li>• Radial Gradients</li> <li>• (Optional: WebKit Only) Text Stroke</li> <li>• Multiple Background Images</li> <li>• (Optional: WebKit Only) Reflections</li> <li>• Image Borders 188 5.12 Animation; Selectors</li> <li>• Transitions and Transformations</li> <li>• transition and transform Properties</li> <li>• Skew</li> <li>• Transitioning Between Images</li> <li>• Downloading Web Fonts and the @font-face Rule</li> <li>• Flexible Box Layout Module and :nth-child Selectors</li> <li>• Multicolumn Layout</li> <li>• Media Queries</li> <li>• Web Resources</li> </ul>
6	JavaScript: Introduction to Scripting	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Your First Script: Displaying a Line of Text with JavaScript in a Web Page</li> <li>• Modifying Your First Script</li> <li>• Obtaining User Input with prompt Dialogs</li> <li>• Dynamic Welcome Page</li> <li>• Adding Integers</li> <li>• Memory Concepts</li> <li>• Arithmetic</li> </ul>

		<ul style="list-style-type: none"> <li>• Decision Making: Equality and Relational Operators</li> <li>• Web Resources</li> </ul>
7	JavaScript: Control Statements I	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Algorithms</li> <li>• Pseudocode</li> <li>• Control Statements</li> <li>• if Selection Statement</li> <li>• if...else Selection Statement</li> <li>• while Repetition Statement</li> <li>• Formulating Algorithms: Counter-Controlled Repetition</li> <li>• Formulating Algorithms: Sentinel-Controlled Repetition</li> <li>• Formulating Algorithms: Nested Control Statements</li> <li>• Assignment Operators</li> <li>• Increment and Decrement Operators</li> <li>• Web Resources</li> </ul>
8	JavaScript: Control Statements II	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Essentials of Counter-Controlled Repetition</li> <li>• for Repetition Statement</li> <li>• Examples Using the for Statement</li> <li>• switch Multiple-Selection Statement</li> <li>• do...while Repetition Statement</li> <li>• break and continue Statements</li> <li>• Logical Operators</li> <li>• Web Resources</li> </ul>
9	JavaScript: Functions	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Program Modules in JavaScript</li> <li>• Function Definitions</li> <li>• Programmer-Defined Function square</li> <li>• Programmer-Defined Function maximum</li> <li>• Notes on Programmer-Defined Functions</li> <li>• Random Number Generation</li> <li>• Scaling and Shifting Random Numbers</li> <li>• Displaying Random Images</li> <li>• Rolling Dice Repeatedly and Displaying Statistics</li> <li>• Example: Game of Chance; Introducing the HTML5 audio and video Elements</li> <li>• Scope Rules</li> <li>• JavaScript Global Functions</li> <li>• Recursion</li> <li>• Recursion vs. Iteration</li> </ul>

10	JavaScript: Arrays	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Arrays</li> <li>• Declaring and Allocating Arrays</li> <li>• Examples Using Arrays</li> <li>• Creating, Initializing and Growing Arrays</li> <li>• Initializing Arrays with Initializer Lists</li> <li>• Summing the Elements of an Array with for and for...in</li> <li>• Using the Elements of an Array as Counters</li> <li>• Random Image Generator Using Arrays</li> <li>• References and Reference Parameters</li> <li>• Passing Arrays to Functions</li> <li>• Sorting Arrays with Array Method sort</li> <li>• Searching Arrays with Array Method indexOf</li> <li>• Multidimensional Arrays</li> </ul>
11	JavaScript: Objects	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Math Object</li> <li>• String Object</li> <li>• Fundamentals of Characters and Strings</li> <li>• Methods of the String Object</li> <li>• Character-Processing Methods</li> <li>• Searching Methods</li> <li>• Splitting Strings and Obtaining Substrings</li> <li>• Date Object</li> <li>• Boolean and Number Objects</li> <li>• document Object</li> <li>• Favorite Twitter Searches: HTML5 Web Storage</li> <li>• Using JSON to Represent Objects</li> </ul>

## Text Book:

- Paul Deitel, Harvey Deitel and Abbey Deitel, Internet & world wide Web How to Program, 5<sup>th</sup> Edition, Pearson, 2023.

## References:

1. HTML, CSS & JavaScript in easy steps Paperback – Special Edition, In Easy Steps Limited, 2020.
2. Jon Duckett, HTML & CSS Design and Build Websites, Wiley.
3. Erik Wilde, Wilde's WWW Technical Foundations of the World Wide Web, Springer.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Operating Systems of Computers and Smart devices
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course description

Every computer, smartphone or similar electronic device comes with special software called an operating system, In this Course, will be studied the operating systems in computers and smart devices. It is a general introduction to the techniques used in implementing operating systems and the types of related system programs. Topics covered will be the functions and structure of operating systems, process management; deadlock prevention, avoidance, and recovery; Main memory management Virtual memory management, we'll discuss what operating systems are, explain why they're important and explore the different types of operating systems in use today.



## Course Objectives

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication.
3. Understand the Basic functions of operating systems.
4. Explain main components of OS and their working.
5. Understanding of structure of modern OS smart device.
6. Developing students' skills in designing and programming systems which use smart devices.

## Course Contents

Unit Number	Unit Name	Contents
1	History Of Operating Systems	<ul style="list-style-type: none"> <li>• The First Generation</li> <li>• The Second Generation</li> <li>• The Third Generation</li> <li>• The Fourth Generation</li> <li>• The Fifth Generation</li> <li>• Mobile Computers</li> </ul>
2	The Operating System ZOO	<ul style="list-style-type: none"> <li>• Mainframe Operating Systems</li> <li>• Server Operating Systems</li> <li>• Multiprocessor Operating Systems</li> <li>• Personal Computer Operating Systems</li> <li>• Handheld Computer Operating Systems</li> <li>• Embedded Operating Systems</li> <li>• Sensor-Node Operating Systems</li> <li>• Real-Time Operating Systems</li> <li>• Smart Card Operating Systems</li> </ul>
3	Computer Hardware Review	<ul style="list-style-type: none"> <li>• Processors</li> <li>• Memory</li> <li>• Disks</li> <li>• I/O Devices</li> <li>• Buses</li> <li>• Booting the Computer</li> </ul>
4	Operating System Concepts	<ul style="list-style-type: none"> <li>• Processes</li> <li>• Address Spaces</li> <li>• Files</li> </ul>

		<ul style="list-style-type: none"> <li>• Input/Output</li> <li>• Protection</li> <li>• The Shell</li> <li>• Ontogeny Recapitulates Phylogeny</li> </ul>
5	System calls	<ul style="list-style-type: none"> <li>• System Calls for Process Management</li> <li>• System Calls for File Management</li> <li>• System Calls for Directory Management</li> <li>• Miscellaneous System Calls</li> <li>• The Windows Win32 API</li> </ul>
6	Operating system structure	<ul style="list-style-type: none"> <li>• Monolithic Systems</li> <li>• Layered Systems</li> <li>• Microkernels</li> <li>• Client-Server Model</li> <li>• Virtual Machines</li> </ul>
7	Mobile Devices & Mobile Operating System	<ul style="list-style-type: none"> <li>• Mobile Devices</li> <li>• Function Of Mobile Operating System</li> <li>• Mobile Operating System</li> <li>• Generalized Mobile Operating System Architecture and Comparison</li> </ul>
8	Android Operating System and iOS	<ul style="list-style-type: none"> <li>• Basics of Android Operating System</li> <li>• Internal Mechanism of Android OS</li> <li>• IOS Operating System</li> </ul>
9	Other Mobile Operating System	<ul style="list-style-type: none"> <li>• Windows Phone</li> <li>• Blackberry</li> <li>• Symbain</li> </ul>

## Text Book:

- Modern Operating Systems, 5th edition, Published by Pearson (October 14th 2022) - Copyright © 2023.

## References:

1. Mobile operating system Dr. Babasaheb Ambedkar Open University 2019.  
[https://baou.edu.in/assets/pdf/PGDMAD\\_101\\_slm.pdf](https://baou.edu.in/assets/pdf/PGDMAD_101_slm.pdf)
2. Operating System Concepts 8th Edition ,by Abraham Silberschatz , Peter B. Galvin , Greg Gagne , 2019.
3. [https://www.toppr.com/guides/computer-science/computer-fundamentals/operating-system/mobile-operating-system/#Features\\_of\\_Mobile\\_Operating\\_System](https://www.toppr.com/guides/computer-science/computer-fundamentals/operating-system/mobile-operating-system/#Features_of_Mobile_Operating_System)
4. <https://none.cs.umass.edu/~shenoy/courses/fall00/syllabus.html>.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Games Development
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	1
<b>Practical</b>	6

### Course Description

Using the Unity game engine to its full potential for both 3D and 2D game development—from the basics of scripting to useful tricks in gameplay, behavior, and animation.

### Course Objectives

1. Develop competence necessary for graduate students to be employed in the areas of information technology and the industry of game development.
2. Enable students to develop games individually and in teams.
3. Develop creativity and individuality in problem solving and in performing tasks.
4. Enable students to continue their studies and to introduce them to scientific work in the areas of virtual reality, computer graphics, and artificial intelligence.

### Course Contents

Unit Number	Unit Name	Contents
1	Working with Unity	<ul style="list-style-type: none"> <li>• Getting Around in Unity</li> <li>• Working with Game Objects</li> <li>• Working with Components</li> <li>• Working with Prefabs</li> <li>• Working with Scenes</li> <li>• Managing Assets</li> <li>• Building Unity Projects</li> <li>• Accessing Preferences</li> </ul>
2	Scripting	<ul style="list-style-type: none"> <li>• Working with MonoBehaviours</li> <li>• Using Callbacks</li> <li>• Creating Frame Rate–Independent Behavior</li> <li>• Accessing Components</li> <li>• Finding Objects</li> <li>• Using Coroutines</li> <li>• Singletons</li> <li>• Loading a Level (Scene)</li> <li>• Storing Data on Disk</li> <li>• Saving and Loading the Game State</li> <li>• Managing Objects Using an Object Pool</li> <li>• Storing Data in Assets Using ScriptableObject</li> </ul>
3	Input	<ul style="list-style-type: none"> <li>• Working with Keyboard Input</li> <li>• Working with Mouse Input</li> <li>• Locking and Hiding the Mouse Cursor</li> <li>• Working with Gamepads</li> <li>• Customizing Unity’s Input System</li> <li>• Responding to Pointer Events from the Event System</li> </ul>
4	Math	<ul style="list-style-type: none"> <li>• Storing Coordinates of Varying Dimensions Using Vectors</li> <li>• Rotating in 3D Space</li> <li>• Performing Transformations in 3D Space with Matrices</li> <li>• Working with Angles</li> <li>• Finding the Distance to a Target</li> <li>• Finding the Angle to a Target</li> </ul>
5	2D Graphics	<ul style="list-style-type: none"> <li>• Importing Sprites</li> <li>• Adding a Sprite to the Scene</li> <li>• Creating a Sprite Animation</li> <li>• Creating a Sprite with 2D Physics</li> <li>• Customizing Sprite Collision Shapes</li> <li>• Using a Composite Collider</li> </ul>

		<ul style="list-style-type: none"> <li>• Using the Sprite Packer</li> <li>• Applying Forces to 2D Objects</li> <li>• Creating a Conveyor Belt</li> <li>• Using a Custom Material for Sprites</li> <li>• Managing Sprite Sorting</li> <li>• Using Sorting Groups</li> <li>• Creating a 2.5D Scene</li> </ul>
6	3D Graphics	<ul style="list-style-type: none"> <li>• Creating a Simple Material</li> <li>• Controlling a Material's Property Through a Script</li> <li>• Creating an Unlit Material</li> <li>• Setting Up a Material Using Textures</li> <li>• Making a Material Use a Shader</li> <li>• Setting Up a Bloom Effect Using Post-Processing</li> <li>• Using High-Dynamic-Range Colors</li> <li>• Setting Up a Project to Use a Scriptable Render Pipeline</li> <li>• Creating a Shader Using the Shader Graph</li> <li>• Creating a Glowing Effect Using the Shader Graph</li> <li>• Exposing Properties from a Shader Graph</li> <li>• Animating a Shader over Time</li> <li>• Controlling the Speed of an Animated Shader</li> <li>• Using a Subgraph to Reuse Graph Components</li> <li>• Implementing a Dissolve Effect Using a Shader Graph</li> <li>• Using Baked Lighting and Real-Time Lighting</li> <li>• Using Baked Emission Sources</li> <li>• Making Static Objects Cast Shadows on Dynamic Objects</li> <li>• Using Light Probes to Influence Lighting</li> <li>• Using Reflection Probes</li> <li>• Faking a Dynamic Emissive Object</li> <li>• Loading Textures from Disk</li> <li>• Rendering to a Texture</li> <li>• Saving Screenshots</li> </ul>
7	Physics and Character Control	<ul style="list-style-type: none"> <li>• Understanding FixedUpdate</li> <li>• Implementing Mouselook</li> <li>• Controlling a 3D Character</li> <li>• Interacting with Switches and Objects</li> <li>• Picking Up and Putting Down Objects</li> <li>• Detecting When an Object Is Touching Another Object</li> <li>• Detecting When an Object Is in a Trigger Area</li> </ul>

		<ul style="list-style-type: none"> <li>• Implementing Moving Platforms</li> <li>• Implementing Platform Riding</li> <li>• Responding to Being Pushed by Objects</li> </ul>
8	Animation and Movement	<ul style="list-style-type: none"> <li>• Animating an Object</li> <li>• Basic Character Movement</li> <li>• Inverse Kinematics</li> <li>• Masked Movement</li> <li>• Blended Movement</li> <li>• Navigation and Animating in Sync</li> <li>• Cinematic Camera Tracking</li> <li>• Automatically Switching Cameras</li> <li>• Keeping Multiple Objects in View</li> <li>• Dollying a Camera</li> </ul>
9	Gameplay	<ul style="list-style-type: none"> <li>• Managing Quests</li> <li>• Managing Hitpoints</li> <li>• Creating a Top-Down Camera</li> <li>• Dragging a Box to Select Objects</li> <li>• Creating a Menu Structure</li> <li>• Creating a Wheeled Vehicle</li> <li>• Keeping a Car from Tipping Over</li> <li>• Creating Speed Boosts</li> <li>• Creating a Camera That Orbits Around Its Target</li> <li>• Creating Orbiting Cameras That Won't Clip Through Walls</li> <li>• Detecting When the Player Has Completed a Lap</li> </ul>
10	Behavior and AI	<ul style="list-style-type: none"> <li>• Defining a Path That AI Entities and the Player Can Follow</li> <li>• Letting Entities in Your Game Follow a Path</li> <li>• Enemies Detecting When They Can See the Player</li> <li>• Finding a Good Distribution of Random Points (Poisson Disc)</li> <li>• Enemies Detecting Where They Can Take Cover</li> <li>• Building and Using a State Machine</li> </ul>
11	Sound and Music	<ul style="list-style-type: none"> <li>• Playing Sounds</li> <li>• Setting Up a Mixer</li> <li>• Using Audio Effects</li> <li>• Using Send and Receive Effects</li> <li>• Ducking</li> <li>• Using Multiple Audio Zones</li> <li>• Playing Audio Through Code</li> <li>• Using a Sound Manager</li> </ul>
12	User Interface	<ul style="list-style-type: none"> <li>• Working with UI Controls</li> </ul>

---

---

		<ul style="list-style-type: none"><li>• Theming Controls</li><li>• Animating the UI</li><li>• Creating a List of Items</li><li>• Fading Out List Items</li><li>• Creating Onscreen Position Indicators</li><li>• Custom Editors</li><li>• Property Drawers</li><li>• Attribute Drawers</li><li>• Asset Processing</li><li>• Scripted Importers</li><li>• Wizards</li></ul>
--	--	--

Text Book:

- Paris Buttfield-Addison, Jon Manning, and Tim Nugent, Unity Game Development Cookbook, O'Reilly, 2019.

References:

1. Joseph Hocking, Unity in Action, Manning, 3rd edition 2022.
2. Nicolas Alejandro Borrromeo, Hands-On Unity 2022 Game Development, 3rd Edition, Packt Publishing, 2022.
3. Unity from Zero to Proficiency (Beginner): A Step-by-step guide to coding your first game Paperback, Patrick Felicia, 2019.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Introduction to Software Engineering
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course Description

This course focuses on the system Development algorithms and some development concepts and design. Also, it concerned on how to use data structure to solve problems in computer packages, system lifecycle and some strategic integrations tools.

### Course Objectives

1. Understand what software engineering is and why it is important; Enable students to develop games individually and in teams.
2. Understand that the development of different types of software.
3. Understand the concepts of software processes and software.
4. Understand the rationale for agile software development methods, the agile manifesto, and the differences between agile and plan-driven development. Understand the concepts of user and system requirements and why these requirements should be written in different ways.
5. Understand how graphical models can be used to represent software systems and why several types of model are needed to fully represent a system.
6. Understand the most important activities in a general, object-oriented design process.
7. Know about three distinct types of testing—component testing, system testing, and release testing.



## Course Contents

Unit Number	Unit Name	Contents
1	Introduction	<ul style="list-style-type: none"> <li>Professional software development</li> <li>Software engineering ethics</li> </ul>
2	Software processes	<ul style="list-style-type: none"> <li>Software process models</li> <li>Process activities</li> <li>Coping with change</li> <li>Process improvement</li> </ul>
3	Agile software development	<ul style="list-style-type: none"> <li>Agile methods</li> <li>Agile development techniques</li> <li>Agile project management</li> <li>Scaling agile methods</li> </ul>
4	Requirements engineering	<ul style="list-style-type: none"> <li>Functional and non-functional requirements</li> <li>Requirements engineering processes</li> <li>Requirements elicitation</li> <li>Requirements specification</li> <li>Requirements validation</li> <li>Requirements change</li> </ul>
5	System modeling	<ul style="list-style-type: none"> <li>Context models</li> <li>Interaction models</li> <li>Structural models</li> <li>Behavioral models</li> <li>Model-driven architecture</li> </ul>
6	Architectural design	<ul style="list-style-type: none"> <li>Architectural design decisions</li> <li>Architectural views</li> <li>Architectural patterns</li> <li>Application architectures</li> </ul>
7	Design and implementation	<ul style="list-style-type: none"> <li>Object-oriented design using the UML</li> <li>Design patterns</li> <li>Implementation issues</li> <li>Open-source development</li> </ul>
8	Software testing	<ul style="list-style-type: none"> <li>Development testing</li> <li>Test-driven development</li> <li>Release testing</li> <li>User testing</li> </ul>
9	Software evolution	<ul style="list-style-type: none"> <li>Evolution processes</li> <li>Legacy systems</li> <li>Software maintenance</li> </ul>

Text Book:

- Software Engineering, TENTH edition, Ian Sommerville, Pearson, 2021.

References:

1. Software Engineering for Absolute Beginners: Your Guide to Creating Software, 1st ed. Edition, Nico Loubser, Apress, 2021.
2. Clean Code: A Handbook of Agile Software Craftsmanship, 1<sup>st</sup> edition, Robert C. Martin, 2008.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Java Programming
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	0
<b>Practical</b>	6

### Course description

Class String in Java and its common built-in methods, ArrayList Class and its common built-in methods, file processing, reading/writing from and to text files using different approaches, graphical user interface using JavaFX techniques, connection to database including select, insert, update, and delete statements, and writing JavaFX applications connected with database.

### Course Objectives

1. Writing programs in Java to deal with texts
2. Ability to call modern packages and classes to build array lists
3. Writing programs that use ready-made functions to control files and folders
4. Work on projects involving reading and writing operations from and to text files
5. The ability to design graphical interfaces using JavaFX technology
6. Linking Java programs to databases
7. Writing programs for graphical interfaces that contain basic events
8. Writing programs in which graphical interfaces are linked with databases within an integrated software project

### Course Contents

Unit Number	Unit Name	Contents
12	Exception Handling and Text I/O	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Exception-Handling Overview</li> <li>• Exception Types</li> <li>• Declaring, Throwing, and Catching Exceptions</li> <li>• The finally Clause</li> <li>• When to Use Exceptions</li> <li>• Rethrowing Exceptions</li> <li>• Chained Exceptions</li> <li>• Defining Custom Exception Classes</li> <li>• The File Class</li> <li>• File Input and Output</li> <li>• Reading Data from the Web</li> <li>• Case Study: Web Crawler</li> </ul>
13	Abstract Classes and Interfaces	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Abstract Classes</li> <li>• Case Study: The Abstract Number Class</li> <li>• Case Study: Calendar and GregorianCalendar</li> <li>• Interfaces</li> <li>• The Comparable Interface</li> <li>• The Cloneable Interface</li> <li>• Interfaces vs. Abstract Classes</li> <li>• Case Study: The Rational Class</li> </ul>

		<ul style="list-style-type: none"> <li>• Class-Design Guidelines</li> </ul>
14	JavaFX Basics	<ul style="list-style-type: none"> <li>• JavaFX Basics</li> <li>• JavaFX vs Swing and AWT</li> <li>• The Basic Structure of a JavaFX Program</li> <li>• Panes, Groups, UI Controls, and Shapes</li> <li>• Property Binding</li> <li>• Common Properties and Methods for Nodes</li> <li>• The Color Class</li> <li>• The Font Class</li> <li>• The Image and ImageView Classes</li> <li>• Layout Panes and Groups</li> <li>• Shapes</li> <li>• Case Study: The ClockPane Class</li> </ul>
17	Binary I/O	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• How Is Text I/O Handled in Java?</li> <li>• Text I/O vs. Binary I/O</li> <li>• Binary I/O Classes</li> <li>• Case Study: Copying Files</li> <li>• Object I/O</li> <li>• Random-Access Files</li> </ul>
34	Java Database Programming	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Relational Database Systems</li> <li>• SQL</li> <li>• JDBC</li> <li>• PreparedStatement</li> <li>• CallableStatement</li> <li>• Retrieving Metadata</li> </ul>
35	Advanced Java Database Programming	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• A Universal SQL Client</li> <li>• Batch Processing</li> <li>• Scrollable and Updatable Result Set</li> <li>• RowSet, JdbcRowSet, and CachedRowSet</li> <li>• Storing and Retrieving Images in JDBC</li> </ul>

Text Book:

- Y. Daniel Liang, Introduction to Java Programming and Data Structures Comprehensive Version, 12<sup>th</sup> Edition, Pearson, 2020.

References:

1. David Gries and Orit Hazzan, Java in Two Semesters Featuring JavaFX, Springer, 4<sup>th</sup> Edition, 2020.
2. Paul Deitel and Harvey Deitel, Java How to Program, Early Objects (Deitel: How to Program), Pearson; 11<sup>th</sup> edition, 2017.
3. John Lewis and William Loftus, Java Software Solutions, Pearson, 9th edition, 2017.
4. <https://docs.oracle.com/javase/tutorial/>
5. Learn Java, Spring Boot and Hibernate (howtodoinjava.com)

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Object Oriented Programming
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	1
<b>Practical</b>	4

### Course description

Topics covered in this course include the programming concepts, structures, functions, objects and classes, class members and their visibility, class hierarchies, single and multiple inheritance, static data members and methods, constructors and destructors, memory management, method and operator overloading, method overriding, abstract classes, polymorphism and error handling.

### Course Objectives

1. Writing programs code using variables, constants, and mathematic and logic operations.
2. Using the choices and loops sentences.
3. Writing programs using functions that used to control the text and mathematical expressions.
4. Ability to design and call methods, passing values, Ability to design and call methods, passing values and overloading variables.
5. Writing programs that use one-dimensional and multi-dimensional arrays.
6. Work on projects involving defining object classes and creating objects.
7. Writing programs that use abstraction, encapsulation, class relations, inheritance and polymorphism, overriding and overloading.
8. The ability to handle exceptions by knowing their types, when they are used, and re-throwing them.

### Course Contents

Unit Number	Unit Name	Contents
1	Introduction to Computers, Programs, and Java	<ul style="list-style-type: none"> <li>• Introduction, What is a Computer?, Programming language</li> <li>• Operating systems</li> <li>• Java, the World Wide Web, and Beyond</li> <li>• The Java Language Specification, API, JDK, and IDE, A Simple Java Program</li> <li>• Creating, Compiling, and Executing a Java Program</li> <li>• Programming Style and Documentation</li> <li>• Programming Errors, Developing Java Programs Using NetBeans</li> </ul>
2	Elementary Programming	<ul style="list-style-type: none"> <li>• Introduction, Writing a Simple Program, Reading Input from the Console.</li> <li>• Identifiers, Variables, Assignment Statements and Assignment Expressions.</li> <li>• Named Constants, Naming Conventions.</li> <li>• Numeric Data Types and Operations.</li> <li>• Numeric Literals, Jshell, Evaluating Expressions and Operator Precedence.</li> </ul>



		<ul style="list-style-type: none"> <li>• Case Study: Displaying the Current Time.</li> <li>• Augmented Assignment Operators, Increment and Decrement Operators.</li> <li>• Numeric Type Conversions, Software Development Process.</li> <li>• Case Study: Counting Monetary, Common Errors and Pitfalls.</li> </ul>
3	Selections	<ul style="list-style-type: none"> <li>• Introduction, boolean Data Type</li> <li>• If Statements, Two-Way if-else Statements, Nested if and Multi-Way if-else Statements</li> <li>• Common Errors and Pitfalls, Generating Random Numbers</li> <li>• Case Study: Computing Body Mass Index</li> <li>• Case Study: Computing Taxes</li> <li>• Logical Operators</li> <li>• Case Study: Determining Leap Year</li> <li>• Case Study: Lottery</li> <li>• Switch Statements</li> <li>• Conditional Expressions</li> <li>• Operator Precedence and Associativity</li> <li>• Debugging</li> </ul>
4	Mathematical, functions, Characters, and Strings	<ul style="list-style-type: none"> <li>• Introduction, Common Mathematical Functions</li> <li>• Character Data Type and Operations</li> <li>• The String Type</li> <li>• Case Studies</li> <li>• Formatting Console Output</li> </ul>
5	Loops	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The while Loop</li> <li>• Case Study: Guessing Numbers</li> <li>• Loop Design Strategies</li> <li>• Controlling a Loop with User Confirmation or a Sentinel Value</li> <li>• The do-while Loop</li> <li>• The for Loop</li> <li>• Which Loop to Use?</li> <li>• Nested Loops</li> <li>• Minimizing Numeric Errors</li> <li>• Case Studies</li> <li>• Keywords break and continue</li> <li>• Case Study: Checking Palindromes</li> <li>• Case Study: Displaying Prime Numbers</li> </ul>
6	Methods	<ul style="list-style-type: none"> <li>• Introduction</li> </ul>

		<ul style="list-style-type: none"> <li>• Defining a Method</li> <li>• Calling a Method</li> <li>• void vs. Value-Returning Methods</li> <li>• Passing Arguments by Values</li> <li>• Modularizing Code</li> <li>• Case Study: Converting Hexadecimals to Decimals</li> <li>• Overloading Methods</li> <li>• The Scope of Variables</li> <li>• Case Study: Generating Random Characters</li> <li>• Method Abstraction and Stepwise Refinement</li> </ul>
7	Single-Dimensional Array	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Array Basics</li> <li>• Case Study: Analyzing Numbers</li> <li>• Case Study: Deck of Cards</li> <li>• Copying Arrays</li> <li>• Passing Arrays to Methods</li> <li>• Returning an Array from a Method</li> <li>• Case Study: Counting the Occurrences of Each Letter</li> <li>• Variable-Length Argument Lists</li> <li>• Searching Arrays</li> <li>• Sorting Arrays</li> <li>• The Arrays Class</li> <li>• Command-Line Arguments</li> </ul>
8	Multidimensional Arrays	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Two-Dimensional Array Basics</li> <li>• Processing Two-Dimensional Arrays</li> <li>• Passing Two-Dimensional Arrays to Methods</li> <li>• Case Study: Grading a Multiple-Choice Test</li> <li>• Case Study: Finding the Closest Pair</li> <li>• Case Study: Sudoku</li> <li>• Multidimensional Arrays</li> </ul>
9	Objects and classes	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Defining Classes for Objects</li> <li>• Example: Defining Classes and Creating Objects</li> <li>• Constructing Objects Using Constructors</li> <li>• Accessing Objects via Reference Variables</li> <li>• Using Classes from the Java Library</li> <li>• Static Variables, Constants, and Methods</li> <li>• Visibility Modifiers</li> <li>• Data Field Encapsulation</li> <li>• Passing Objects to Methods</li> </ul>

		<ul style="list-style-type: none"> <li>• Array of Objects</li> <li>• Immutable Objects and Classes</li> <li>• The Scope of Variables</li> <li>• The this Reference</li> </ul>
10	Object-Oriented Thinking	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Class Abstraction and Encapsulation</li> <li>• Thinking in Objects</li> <li>• Class Relationships</li> <li>• Case Study: Designing the Course Class</li> <li>• Case Study: Designing a Class for Stacks</li> <li>• Processing Primitive Data Type Values as Objects</li> <li>• Automatic Conversion between Primitive Types and Wrapper Class Types</li> <li>• The BigInteger and BigDecimal Classes</li> <li>• The String Class</li> <li>• The StringBuilder and StringBuffer Classes</li> </ul>
11	Inheritance and Polymorphism	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Superclasses and Subclasses</li> <li>• Using the super Keyword</li> <li>• Overriding Methods</li> <li>• Overriding vs. Overloading</li> <li>• The Object Class and Its toString() Method</li> <li>• Polymorphism</li> <li>• Dynamic Binding</li> <li>• Casting Objects and the instanceof Operator</li> <li>• The Object's equals Method</li> <li>• The ArrayList Class</li> <li>• Useful Methods for Lists</li> <li>• Case Study: A Custom Stack Class</li> <li>• The protected Data and Methods</li> <li>• Preventing Extending and Overriding</li> </ul>

## Text Book:

- Y. Daniel Liang, Introduction to Java Programming and Data Structures Comprehensive Version, 12<sup>th</sup> Edition, Pearson, 2020.

## References:

1. John Lewis and William Loftus, Java Software Solutions, Pearson, 9th edition, 2017.
2. <https://docs.oracle.com/javase/tutorial/>
3. Learn Java, Spring Boot and Hibernate (howtodoinjava.com)

- 
4. Tutorials List - Javatpoint
  5. Online Tutorials Library (tutorialspoint.com)

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Smart Phone Applications Development
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	0
<b>Practical</b>	6

### Course description

This course includes activities and transitions between activities, Android user interfaces and widgets, activity layouts, Android debugging and testing, fragments, shared preferences, SQLite and firebase databases, XML and JSON processing, the content provider, services, message broadcasting, async task and threading, the media player, sensors, Android Google maps, etc.

### Course Objectives

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

1. To design and implement different Android mobile applications.
2. To explain the concepts, design strategies, tools and APIs needed to create mobile applications.
3. To test and deploy applications for phone and smart devices.

### Course Contents

Unit Number	Unit Name	Contents
1	Java Review	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Language Basics (Variables, Type of Variable, Java Primitive Data Types, Default Variable Initializations, Typecasting, Type Assignment Example, Java String Class, Java Operators, Control Flow Statements, Arrays, ArrayList, Java Iterator Interface, For-Each Loop)</li> <li>• Object-Oriented Programming Concepts in Java (Classes, Objects, Interfaces, Package, Inheritance, Inheritance Example, Polymorphism, Hiding Fields, Using the Keyword Super, Subclass Constructors, Using Preserved Keyword “this”, Java Exception Handling, Generic Types, Type Parameter Naming Conventions, Autoboxing, Parameterized Types, Anonymous Classes, Object Serialization, Lambda Expressions, Variable Argument (Varargs))</li> </ul>
2	Getting Started with Android	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Starting with Android (A Brief Android History, Android Is Open Source, Android Libraries, Android Popularity, Android Development Environment, Android Developer’s Skills, Model View Controller and App Development, Android’s Main Program, Java and Android, Why Use Java for Android?, Android and Linux)</li> <li>• Download and Install Android Studio and Android SDK (Download the Android Studio, Install Android Studio, Update Android Files, Release Note, Android SDK)</li> </ul>

		<ul style="list-style-type: none"> <li>• Create a New Android Project (Start New Project, Select an Activity Template, Fill in Application Requirement, Define SDK Requirements, Finish the Project Creation)</li> <li>• Compiling and Running Android Apps (Running HelloWorld on Your Phone, Running the Android App in Android Studio, Issues Starting First App, Running HelloWorld on Emulator, Setting Up the Emulator, Do It Yourself)</li> <li>• Compiling, Building, and Packaging Technologies (Compiling Android Code, Android App Bundle, Do It Yourself, Install Android Apps, Install APK from Online, Install APK from Files, From Dalvik to ART Runtime, Gradle Build, Software Versioning Using Local or Remote Repositories)</li> <li>• Android Stack and Framework (Android Architecture, User and System Apps, Java API Framework, Native Libraries and Android Runtime, Hardware Abstraction Layer (HAL), Linux Kernel)</li> </ul>
3	Your First Android Application	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Android App Development (Early Android Development, Android Versions, Android Application Characteristic, Android Activity, R File, Android Context, Application Manifest Files, Opening Android Project in Android Studio, Cleaning Android Project Builds)</li> <li>• Create Your First Mobile App (Your App Specification, Create Activity Layout, Invoke Message on Activity, Intent Class, Using StartActivity, Create Second Activity, Project Manifest Update, Running the App, Receiving Messages/Data from an Activity, Responding to the Messages from an Activity)</li> </ul>

		<ul style="list-style-type: none"> <li>• Debugging Information (Debugging Using Log.d(),Using Logcat to View Log Messages, Do It Yourself)</li> <li>• Localize Your App and Resources (Create a Resource File for Second Language, Create Resource Entries for Languages Supported, Set Device Language)</li> <li>•</li> </ul>
4	Debugging and Testing Using Junit, Espresso, and Mockito Frameworks	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The Android Studio Debugger (Fault Handling Methods, Enable Debugger, Inspecting and Modifying Variable Values, Android Profiler, Device File Explorer, Android Debug Bridge (adb), Do It Yourself)</li> <li>• Toast and Snackbar Messages (Toast Messages, Snackbar Messages, Do It Yourself, The Log Class and Logcat Window)</li> <li>• Android App Testing (Create a Test Class, Assert Methods, Hamcrest Assert Methods, Espresso Testing, Unit Testing, Unit Testing with Mockito, Code Coverage, Code Inspection and Refactoring, Reverse Engineering)</li> </ul>
5	Activity Lifecycle and Passing Objects Between Screens Using Parcelable Interface	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Activity States (Activity and States, Transition Between States, The Launcher Activity, Implementing onCreate(),Bundle Class)</li> <li>• Understanding Activity Lifecycle (Understanding the onDestroy Method, Pausing and Resuming an Activity, Stopping and Restarting an Activity, Restoring Activity State, Do It Yourself)</li> </ul>



		<ul style="list-style-type: none"> <li>• Lifecycle Illustration App (Lifecycle Callback Methods, Callback Methods for the MainActivity, Callback Methods for the DisplayMessageActivity, Do It Yourself, Callback Method Implementations, Trigger the onPause() Method)</li> <li>• Creating and Using Parcelable Objects (Passing User-Defined Objects Between Activities, LifeCycle with Parcelable Object, Parcelable Example)</li> </ul>
6	User Interface Essential Classes, Layouts, Styles, Themes, and Dimensions	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Essential UI Classes and Properties (Android Project Structure, Views, View Class Examples, Widget, ViewGroup, App Layout)</li> <li>• Writing XML Layouts (Declare UI Elements in XML, Android Studio's Layout Editor, Defining UI Programmatically, LinearLayout Java Class, LayoutParams Java Class)</li> <li>• Details of the LayoutApplication Demo (MainActivity Layout, Activity with Linear Layout, Linear Layout XML File, Using Android Studio Design Option, strings.xml File, String Editor, String Resources, RelativeLayout, Other Layouts, Parent-Child Relationship Between Activities)</li> <li>• Styles, Themes, and Dimension (Defining a Style File, Applying Styles, Defining the App's Theme, The Difference Between a Theme and Style, Padding and Margin View Properties, Gravity and Weight View Properties, Dimensions of a Phone and UI)</li> </ul>

<p>7</p>	<p>ListView, ScrollList, Date and Time Pickers, and RecyclerView</p>	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• List Views (Adapter and ArrayAdapter Classes, ListView and ListActivity)</li> <li>• Date and Time Pickers (Date and Time Pickers, Set Date Using the DatePicker, Set Time Using the TimePicker, Pickers and Anonymous Classes)</li> <li>• Scroll Views (The ScrollView Class, Top-Level XML Element for a Scroll View, Scroll View Activity)</li> <li>• RecyclerView (Using RecyclerView, Adapter, and ViewHolder Classes, RecyclerViewActivity, Adapter and ViewHolder, Using RecyclerView with Older SDKs)</li> <li>•</li> </ul>
<p>8</p>	<p>Toolbar, Menu, Dialog Boxes, Shared Preferences, Implicit Intent, and Directory Structure</p>	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• More User Interface (ActionBar, Toolbar, Add androidx.appcompat Library to the Project, Extending AppCompatActivity, Specify a Theme with NO ActionBar, Adding Toolbar Element to the Layout, Menu Interface, Options Menu and App Bar, Context Menu, Popup Menu)</li> <li>• Dialog Boxes and the Camera App (Dialog Boxes, Custom Dialog Boxes, Access a Phone's Default Camera App, Starting Activities for Results, Activity Result in AndroidX)</li> <li>• Saving Data with SharedPreferences (SharedPreferences Interface, Layout for Shared Preferences Activity, How SharedPreferencesActivity Code Works)</li> </ul>

		<ul style="list-style-type: none"> <li>• Directory Structure and Saving Data in Files (Internal Storage Location, External Storage Location, Standard Public Directories for Data/Files, Android File IO Classes and Methods, Accessing External Storage Files, Permission to Access External Directory, Examples Using External Methods, Environment Class and getExternalStoragePublicDirectory, Locate Apps on Emulator File System)</li> </ul>
11	Android SQLite, Firebase, and Room Databas	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• The Android SQLite Database (SQLiteOpenHelper Class, SQLiteDatabase Class, Overriding Methods of the SQLiteOpenHelper Class, The Class Constructor Method, The onCreate() Method, onUpgrade Method, onDowngrade Method, onOpen() Method, Read and Read/Write Access, The execSQL Method from SQLiteDatabase Class)</li> <li>• Content Values and Cursor Objects (Content Values and Insert Method, Cursor, Query Data,.rawQuery, More Methods of the SQLiteDatabase Class)</li> <li>• DatabaseDemo Project (The Data Component, The Middle Component, The View Component, Test Your Database Using SQLiteBrowser, Use SQLiteBrowser for Database Design, Android Database Inspector)</li> <li>• Realtime Firebase Database (Firebase and JSON Tree File, Firebase Account and Project Setup, Register Your Project Using the Firebase Console, Adding Dependency to Your Project, Connecting to Database, Inserting Data into Database, Retrieving Data from Database, Deleting Data from Database, Query Data from Database, DataSnapshot and Query Classes, ChildEventListener)</li> </ul>

---

---

		<p>Interface, Querying Firebase Database Using User-Defined Classes, Querying Firebase Database Example)</p> <ul style="list-style-type: none"><li>• Other Data Storage Options (Room Database, Content Provider, Internal and External Storage)</li></ul>
--	--	--

### Text Book

- Abdul-Rahman Mawlood-Yunis ,Android for Java Programmers, 1st Edition, Springer, 2022.

### Reference:

1. , J. F. DiMarzio, Beginning Android Programming with Android Studio, J. F. DiMarzio, Fourth Edition, Wrox, 2017.
2. Rick Boyer, Android 9 Development Cookbook – Third Edition, Packt Publishing, 2018.
3. Michael Burton, Android App Development For Dummies, 3rd Edition, Wiley, 2015.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Python Programming
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	0
<b>Practical</b>	6

### Course description

Introduction to programming basics (what it is and how it works), binary computation, problem-solving methods and algorithm development. Includes procedural and data abstractions, program design, debugging, testing, and documentation. Covers data types, control structures, functions, parameter passing, library functions, arrays, inheritance and object oriented design. Laboratory exercises in Python.

### Course Objectives

1. Understand basics of binary computation
2. Understand the programming basics (operations, control structures, data types, etc.)
3. Readily use the Python programming language
4. Apply various data types and control structure
5. Understand class inheritance and polymorphism
6. Understand the object-oriented program design and development
7. Understand and begin to implement code

Course Contents

Unit Number	Unit Name	Contents
1	Input, Processing and Output	<ul style="list-style-type: none"> <li>• Designing a Program</li> <li>• Input, Processing, and Output</li> <li>• Displaying Output with the print Function</li> <li>• Comments</li> <li>• Variables</li> <li>• Reading Input from the Keyboard</li> <li>• Performing Calculations</li> <li>• More About Data Output</li> <li>• Named Constants</li> <li>• Introduction to Turtle Graphics</li> </ul>
2	Decision Structure and Boolean Logic	<ul style="list-style-type: none"> <li>• The if Statement</li> <li>• The if-else Statement</li> <li>• Comparing Strings</li> <li>• Nested Decision Structures and the if-elif-else Statement</li> <li>• Logical Operators</li> <li>• Boolean Variables</li> <li>• Turtle Graphics: Determining the State of the Turtle</li> </ul>
3	Repetition Structures	<ul style="list-style-type: none"> <li>• Introduction to Repetition Structures</li> <li>• The while Loop: A ConditionControlled Loop</li> <li>• The for Loop: A Count-Controlled Loop</li> <li>• Calculating a Running Total</li> <li>• Sentinels</li> <li>• Input Validation Loops</li> <li>• Nested Loops</li> <li>• Turtle Graphics: Using Loops to Draw Designs</li> </ul>
4	Functions	<ul style="list-style-type: none"> <li>• Introduction to Functions</li> <li>• Defining and Calling a Void Function</li> <li>• Designing a Program to Use Functions</li> <li>• Local Variables</li> <li>• Passing Arguments to Functions</li> <li>• Global Variables and Global Constants</li> <li>• Introduction to Value-Returning Functions: Generating Random Numbers</li> <li>• Writing Your Own Value-Returning Functions</li> <li>• The math Module</li> <li>• Storing Functions in Modules</li> <li>• Turtle Graphics: Modularizing Code with Functions</li> </ul>

5	Files and Exceptions	<ul style="list-style-type: none"> <li>• Introduction to File Input and Output</li> <li>• Using Loops to Process Files</li> <li>• Processing Records</li> <li>• Exceptions</li> </ul>
6	Lists and Tuples	<ul style="list-style-type: none"> <li>• Sequences</li> <li>• Introduction to Lists</li> <li>• List Slicing</li> <li>• Finding Items in Lists with the in Operator</li> <li>• List Methods and Useful Built-in Functions</li> <li>• Copying Lists</li> <li>• Processing Lists</li> <li>• Two-Dimensional Lists</li> <li>• Tuples</li> <li>• Plotting List Data with the matplotlib Package</li> </ul>
7	More About Strings	<ul style="list-style-type: none"> <li>• Basic String Operations</li> <li>• String Slicing</li> <li>• Testing, Searching, and Manipulating Strings</li> </ul>
8	Dictionaries and Sets	<ul style="list-style-type: none"> <li>• Dictionaries</li> <li>• Sets</li> <li>• Serializing Objects</li> </ul>
9	Classes and Object-Oriented Programming	<ul style="list-style-type: none"> <li>• Procedural and Object-Oriented Programming</li> <li>• Classes</li> <li>• Working with Instances</li> <li>• Techniques for Designing Classes</li> </ul>
10	Inheritance	<ul style="list-style-type: none"> <li>• Introduction to Inheritance</li> <li>• Polymorphism</li> </ul>
11	Recursion	<ul style="list-style-type: none"> <li>• Introduction to Recursion</li> <li>• Problem Solving with Recursion</li> <li>• Examples of Recursive Algorithms</li> </ul>
12	GUI Programming	<ul style="list-style-type: none"> <li>• Graphical User Interfaces</li> <li>• Using the tkinter Module</li> <li>• Display Text with Label Widgets</li> <li>• Organizing Widgets with Frames</li> <li>• Button Widgets and Info Dialog Boxes</li> <li>• Getting Input with the Entry Widget</li> <li>• Using Labels as Output Fields</li> <li>• Radio Buttons and Check Buttons</li> <li>• Drawing Shapes with the Canvas Widget</li> </ul>

- 
- Tony Gaddis, “Starting Out with Python”, Fourth Edition, Pearson, 2019.

Reference:

1. Python Tutorial, Release 3.7.0, Guido van Rossum and the Python development team, Python Software Foundation, September 02, 2018.
2. Peter Wentworth, Jeffrey Elkner, Allen B. Downey and Chris Meyers ,“How to Think Like a Computer Scientist: Learning with Python 3 Documentation”, Release 3rd Edition, Apr 17, 2020.
3. Mark Lutz , Learning Python, 4<sup>th</sup> edition, O’REILLY.
4. Jonathan Yates, PYTHON, Practical Python Programming For Beginners and Experts.



## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Practical Training
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	-
<b>Practical</b>	280 practical hours

### Course description

The field training for the applied programming major is carried out in the second semester of the second year and under direct supervision by the academic department in the college. A follow-up and evaluation form is prepared for the student to be used by the field training supervisors in the relevant institutions, in addition to preparing a program for the student that includes the basic skills and competencies required in the specialization. To be a guide for the student, the college and the relevant institution

### Course Objectives

1. The student can apply the basic skills of the Applied Programming.
2. The student be able to apply some of the skills he studied at the place of training.
3. The student should finds the Programming solutions that he encounters at work.

### Course Contents

---

---

Unit Number	Unit Name	Contents
1	Office Side	<ul style="list-style-type: none"><li>• See the programs used in the training place.</li><li>• View the used databases and how to organize the data.</li><li>• Conducting a field survey of the types of programs used in the training site.</li></ul>
2	Technical Side	<ul style="list-style-type: none"><li>• Training to deal with the software used</li><li>• Work on finding software solutions to problems in the training site</li><li>• Master the skills of downloading software on the used systems</li></ul>
3	Administrative Side	<ul style="list-style-type: none"><li>• Administrative and educational consideration: attendance, discipline, cooperation, initiative, and adherence to the regulations of the host institution.</li></ul>

#### Text Book and References

- The faculty member responsible for the training and identifies the books and references that may benefit the student in the training process.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	Graduation Project
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	-
<b>Practical</b>	6

### Course description

Enabling the student to realize the importance of the integrated artistic work, and employing knowledge information and skills in the production of the artistic work, and enabling the student to rely on himself in production.

### Course Objectives

1. Employ information and skills in producing an integrated technical project.
2. Using different technologies in the production process
3. The student should be familiar with advance planning and standard specifications for integrated productive work
4. Estimating the cost of raw materials in the production process.
5. Using and studying the programming languages that the student deals with, in producing an integrated project.

---

### Text Book and References

- The faculty member responsible for the graduation project and identifies the books and references that may benefit the student in the training process.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	نظم إدارة المعلومات
<b>Course Number</b>	
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course description

تتناول هذه المادة مفهوم البيانات والمعلومات وأنواعها، مفهوم النظام وأنواعه، مفهوم نظم المعلومات الإدارية ومراحل تطويرها، أمن وسرية نظم المعلومات الإدارية، الدور الاستراتيجي لنظم المعلومات الإدارية، تطور نظم المعلومات الإدارية وأنواعها، إدارة نظم المعلومات الإدارية، وأخلاقيات العمال وأمن المعلومات.

### Course Objectives

بعد دراسة هذه المادة يتوقع من الطالب أن يكون قادرا على تحقيق الأهداف التالية:

1. تعريف الطالب بالدور الاستراتيجي لنظم المعلومات الإدارية
2. تعريف الطالب بالنظم الوظيفية الفرعية للمعلومات
3. تعريف الطالب بمفهوم النظام وأنواعه وسماته
4. تعريف الطالب بمفهوم المعلومات وأنواعها، مفهوم البيانات ومصادرها
5. تعريف الطالب بمفهوم ومراحل تطور نظم المعلومات الإدارية
6. تعريف الطالب بمفهوم إدارة المعلومات الإدارية

Course Contents

Unit Number	Unit Name	Contents
1	الإدارة والمعلومات	<ul style="list-style-type: none"> <li>• تحليل مفهوم نظم المعلومات</li> <li>• نظم المعلومات وتكنولوجيا المعلومات</li> <li>• العوامل المؤثرة في تطور نظم المعلومات</li> <li>• مصادر البيانات</li> <li>• السمات الأساسية للمعلومات</li> <li>• مفهوم الإدارة ومستوياتها</li> <li>• العوامل التنظيمية والهيكلية للمنشأة</li> <li>• أهمية المعلومة لوظائف العملية الإدارية</li> </ul>
2	النظام ونظرية النظم	<ul style="list-style-type: none"> <li>• مفهوم وأنواع النظام</li> <li>• أحكام تحديد النظام</li> <li>• مقومات النظام</li> <li>• نظام المعلومات</li> <li>• نموذج النظم العامة</li> <li>• تصنيف النظم</li> <li>• النظر إلى المنظمة كنظام</li> <li>• توازن النظام</li> </ul>
3	مفهوم نظم المعلومات الإدارية	<ul style="list-style-type: none"> <li>• مفهوم وأهمية نظم المعلومات الإدارية</li> <li>• بيئة دراسة نظم المعلومات الإدارية</li> <li>• تحليل مفهوم نظم المعلومات الإدارية</li> <li>• مميزات وفوائد نظم المعلومات الإدارية</li> <li>• طرق بناء نظم المعلومات الإدارية</li> <li>• الدور الاستراتيجي لنظم المعلومات الإدارية</li> <li>• نظم معلومات المكاتب</li> <li>• نظم معالجة المعاملات والتقارير الإدارية</li> <li>• نظم مساندة القرارات</li> <li>• نظم مساندة القرارات الجماعية</li> <li>• نظم المعلومات التنفيذية</li> </ul>
4	أنواع نظم المعلومات الإدارية	<ul style="list-style-type: none"> <li>• نظم قواعد المعرفة</li> <li>• نظم المعلومات الإدارية والذكاء الاصطناعي</li> <li>• النظم الخبيرة</li> <li>• نظم الشبكات العصبية</li> <li>• نظم الخوارزميات الجينية</li> <li>• نظم المعلومات الوظيفية</li> </ul>
5	تكنولوجيا نظم المعلومات الإدارية	<ul style="list-style-type: none"> <li>• نظام الحاسوب</li> <li>• نظام إدارة قاعدة البيانات</li> <li>• مخطط الكيبنونات – العلاقات</li> <li>• نظم الشبكات والاتصالات</li> <li>• الموارد الانسانية</li> </ul>

6	أخلاقيات الأعمال وأمن المعلومات	<ul style="list-style-type: none"><li>• أخلاقيات الأعمال</li><li>• أخلاقيات الحاسوب</li><li>• الخصوصية والزبائن</li><li>• البرامج الإضافية والتجسس</li><li>• الأمن وجرائم الفضاء الرقمي</li><li>• جرائم الحاسوب</li><li>• حماية المحتوى</li><li>• أمن وسرية نظام المعلومات الإداري</li></ul>
---	---------------------------------	--

Text Book

- موضوعات متقدمة في نظم المعلومات - د ميثان المجالي - 2015 - طبعة - العصار العلمي للنشر

Reference:

1. أساسيات نظم المعلومات الإدارية وتكنولوجيا المعلومات - د.سعد غالب ياسين، دار المناهج للنشر والتوزيع 2006.
2. نظم المعلومات الإدارية - د.فايز جمعه النجار، دار الحامد للنشر والتوزيع.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	نظم إدارة المعلومات
<b>Course Number</b>	020700111
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course description

تتناول هذه المادة مفهوم البيانات والمعلومات وأنواعها، مفهوم النظام وأنواعه، مفهوم نظم المعلومات الإدارية ومراحل تطويرها، أمن وسرية نظم المعلومات الإدارية، الدور الاستراتيجي لنظم المعلومات الإدارية، تطور نظم المعلومات الإدارية وأنواعها، إدارة نظم المعلومات الإدارية، وأخلاقيات العمال وأمن المعلومات.

### Course Objectives

بعد دراسة هذه المادة يتوقع من الطالب أن يكون قادرا على تحقيق الأهداف التالية:

7. تعريف الطالب بالدور الاستراتيجي لنظم المعلومات الإدارية
8. تعريف الطالب بالنظم الوظيفية الفرعية للمعلومات
9. تعريف الطالب بمفهوم النظام وأنواعه وسماته
10. تعريف الطالب بمفهوم المعلومات وأنواعها، مفهوم البيانات ومصادرها
11. تعريف الطالب بمفهوم ومراحل تطور نظم المعلومات الإدارية
12. تعريف الطالب بمفهوم إدارة المعلومات الإدارية



Course Contents

Unit Number	Unit Name	Contents
1	الإدارة والمعلومات	<ul style="list-style-type: none"> <li>● تحليل مفهوم نظم المعلومات</li> <li>● نظم المعلومات وتكنولوجيا المعلومات</li> <li>● العوامل المؤثرة في تطور نظم المعلومات</li> <li>● مصادر البيانات</li> <li>● السمات الأساسية للمعلومات</li> <li>● مفهوم الإدارة ومستوياتها</li> <li>● العوامل التنظيمية والهيكلية للمنشأة</li> <li>● أهمية المعلومة لوظائف العملية الإدارية</li> </ul>
2	النظام ونظرية النظم	<ul style="list-style-type: none"> <li>● مفهوم وأنواع النظام</li> <li>● أحكام تحديد النظام</li> <li>● مقومات النظام</li> <li>● نظام المعلومات</li> <li>● نموذج النظم العامة</li> <li>● تصنيف النظم</li> <li>● النظر إلى المنظمة كنظام</li> <li>● توازن النظام</li> </ul>
3	مفهوم نظم المعلومات الإدارية	<ul style="list-style-type: none"> <li>● مفهوم وأهمية نظم المعلومات الإدارية</li> <li>● بيئة دراسة نظم المعلومات الإدارية</li> <li>● تحليل مفهوم نظم المعلومات الإدارية</li> <li>● مميزات وفوائد نظم المعلومات الإدارية</li> <li>● طرق بناء نظم المعلومات الإدارية</li> <li>● الدور الاستراتيجي لنظم المعلومات الإدارية</li> <li>● نظم معلومات المكاتب</li> <li>● نظم معالجة المعاملات والتقارير الإدارية</li> <li>● نظم مساندة القرارات</li> <li>● نظم مساندة القرارات الجماعية</li> <li>● نظم المعلومات التنفيذية</li> </ul>
4	أنواع نظم المعلومات الإدارية	<ul style="list-style-type: none"> <li>● نظم قواعد المعرفة</li> <li>● نظم المعلومات الإدارية والذكاء الاصطناعي</li> <li>● النظم الخبيرة</li> <li>● نظم الشبكات العصبية</li> <li>● نظم الخوارزميات الجينية</li> <li>● نظم المعلومات الوظيفية</li> </ul>
5	تكنولوجيا نظم المعلومات الإدارية	<ul style="list-style-type: none"> <li>● نظام الحاسوب</li> <li>● نظام إدارة قاعدة البيانات</li> <li>● مخطط الكيبنونات – العلاقات</li> <li>● نظم الشبكات والاتصالات</li> <li>● الموارد الانسانية</li> </ul>

6	أخلاقيات الأعمال وأمن المعلومات	<ul style="list-style-type: none"><li>• أخلاقيات الأعمال</li><li>• أخلاقيات الحاسوب</li><li>• الخصوصية والزيائن</li><li>• البرامج الإضافية والتجسس</li><li>• الأمن وجرائم الفضاء الرقمي</li><li>• جرائم الحاسوب</li><li>• حماية المحتوى</li><li>• أمن وسرية نظام المعلومات الإداري</li></ul>
---	---------------------------------	--

Text Book

- موضوعات متقدمة في نظم المعلومات - د ميثان المجالي - 2015 - طبعة - العصار العلمي للنشر

Reference:

1. أساسيات نظم المعلومات الإدارية وتكنولوجيا المعلومات - د.سعد غالب ياسين، دار المناهج للنشر والتوزيع 2006.
2. نظم المعلومات الإدارية - د.فايز جمعه النجار، دار الحامد للنشر والتوزيع.

## Diploma degree Program

<b>Program Name</b>	Applied Programming
<b>Course Name</b>	الرياضيات والإحصاء
<b>Course Number</b>	020700112
<b>Credit Hours</b>	3
<b>Lecture</b>	3
<b>Practical</b>	0

### Course description

تتناول هذه المادة تعريف الطالب بمجموعات الأعداد، الفترات، المتباينات، المحاور الديكارتية، متوسط التغيير، المشتقة الأولى والتفسير الهندسي لهما، إيجاد المشتقة الأولى وقواعد الإشتقاق، الإشتقاق الضمني، التزايد والتناقص، القيم القصوى، التكامل المحدود وغير المحدود، خواص التكامل، التكامل بالتعويض، تطبيقات على التكامل، مسائل على المساحة والوسط الحسابي، الاحتمالات ونظرية ذات الحدين.

## Course Objectives

<p>بعد دراسة هذه المادة يتوقع من الطالب أن يكون قادراً على تحقيق الأهداف التالية:</p> <ol style="list-style-type: none"> <li>1. أن يتعرف الطالب على جميع مجموعات الاعداد.</li> <li>2. أن يتمكن الطالب من إيجاد الحلول للمتباينات.</li> <li>3. أن يتمكن الطالب من إيجاد المشتقة الأولى وتطبيق قواعد الاشتقاق.</li> <li>4. تعريف الطالب على تطبيقات التكامل.</li> <li>5. أن يتعرف الطالب على التكامل المحدود وغير المحدود وخواص التكامل واستخداماتها في إيجاد المتكاملات.</li> <li>6. إكساب الطالب معرفة كيفية إيجاد معامل الارتباط والانحدار.</li> <li>7. إكساب الطالب معرفة المقاييس الإحصائية.</li> <li>8. أن يتمكن الطالب من المعرفة العلمية والتطبيقية لنظرية الاحتمالات.</li> <li>9. معرفة التوزيع الطبيعي.</li> </ol>
--

## Course Contents

Unit Number	Unit Name	Contents
1	الأعداد	<ul style="list-style-type: none"> <li>• مجموعة الأعداد الطبيعية</li> <li>• مجموعة الأعداد الصحيحة</li> <li>• مجموعة الأعداد النسبية</li> <li>• مجموعة الأعداد غير النسبية</li> <li>• مجموعة الأعداد الحقيقية</li> <li>• الفترات</li> <li>• اتحاد وتقاطع الفترات</li> <li>• لخصائص الجبرية للمتباينات</li> <li>• القيمة المطلقة</li> <li>• المحاور الديكارتية</li> <li>• المسافة</li> </ul>
2	التفاضل	<ul style="list-style-type: none"> <li>• متوسط التغير</li> <li>• المشتقة الأولى</li> <li>• قواعد الاشتقاق</li> <li>• قاعدة السلسلة</li> <li>• الاشتقاق الضمني</li> <li>• معادلة المماس</li> <li>• التزايد والتناقص</li> <li>• القيم القصوى</li> </ul>
3	التكامل	<ul style="list-style-type: none"> <li>• التكامل غير المحدود</li> <li>• التكامل المحدود</li> <li>• خواص التكامل</li> </ul>

		<ul style="list-style-type: none"> <li>التكامل بالتعويض</li> <li>التكامل ذات العلاقة بالاقترانات اللوغاريتمية والأسية</li> </ul>
4	مقاييس النزعة المركزية	<ul style="list-style-type: none"> <li>الوسط الحسابي</li> <li>الوسيط</li> <li>المنوال</li> <li>العلاقة بين الوسط والوسيط والمنوال</li> </ul>
5	مقاييس التشتت	<ul style="list-style-type: none"> <li>المدى ونصف المدى الربيعي</li> <li>الانحراف المتوسط</li> <li>الانحراف المعياري</li> </ul>
6	الإرتباط والانحدار	<ul style="list-style-type: none"> <li>معامل الارتباط</li> <li>معامل ارتباط بيرسون</li> <li>معامل ارتباط سبيرمان للرتب</li> <li>الانحدار</li> </ul>
7	التوزيع الطبيعي	<ul style="list-style-type: none"> <li>القيم المعيارية</li> <li>شكل المنحنى الطبيعي</li> <li>التوزيع الطبيعي المعياري</li> <li>تطبيقات ومسائل على التوزيع الطبيعي</li> </ul>
8	الاحتمالات	<ul style="list-style-type: none"> <li>الفضاء العيني</li> <li>التكرار النسبي والاحتمالات</li> <li>قانون جمع الاحتمالات</li> <li>الأحداث المستقلة</li> <li>الاحتمال الشرطي</li> <li>المتغيرات العشوائية والتوقع</li> <li>نظرية ذات الحدين</li> </ul>

## Text Book

- الرياضيات العامة - م. ريم مصطفى الدبس، مكتبة المجتمع العربي.

## Reference:

1. التفاضل والتكامل، تأليف الدكتور محمد ابو صالح، سلسلة الشامل في الرياضيات.
2. الاحصاء في الإدارة والاقتصاد - مصطفى يوسف كافي - دسمير حموده - هاني ابو عيد - طبعة 2012 - المجتمع العربي.