

**Florida Department of Education
CURRICULUM FRAMEWORK**

Program Title: Web Programming Services
Program Type: Job Preparatory
Occupational Area: Business Technology Education
Components: Four Occupational Completion Points

	<u>Secondary</u>	<u>PSAV</u>
Program Number:	* Not offered at the secondary level	B079200
CIP Number:		0507039907
Grade Level:		30, 31
Standard Length:		1200 hours
Certification:		BUS ED @4 1 @2 VOE @7 TEACH CBE @7 BUS DP @7 G ELECT DP @7 G TEC ELEC \$7 G
Additional certifi- cation accepted:		COMP SCI @6 @2
CSO:		Phi Beta Lambda BPA
Coop Method:		Yes
Apprenticeship:		No
Facility Code:		212
Basic Skills:		
Math		11
Language		11
Reading		11
SOC Code:		15-1021

* Note: A similar program, Academy of Database and Programming Essentials, is offered at the secondary level.

- I. **PURPOSE:** This program is designed to prepare the student for entry-level employment as an Web Developer Assistant, Database Specialist, Junior Application Developer, and Web Application Developer and to provide supplemental training for persons previously or currently employed in any of these occupations.

Web Programming Services offers a foundation in the area of designing and developing interactive, database-driven Web-based applications using an enterprise level Relational Database Management System (RDBMS) software such as Oracle, Microsoft SQL Server, or MySQL and a Web/database connectivity middleware technology such as PHP, ColdFusion, ASP.NET, or Java.

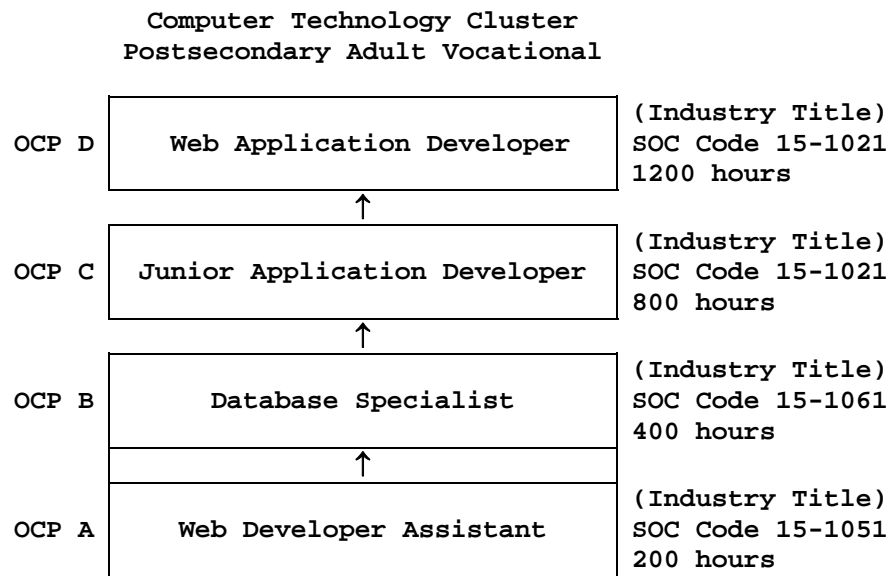
Additional Notes: This program framework is designed to accommodate the use of software from various vendors. It is expected that the instructor will select and focus instruction on a single vendor's RDBMS software together with a single middleware computer language (e.g. Oracle/Java, MS SQL Server/ASP.NET, MySQL/PHP or other combination).

In some cases it may be beneficial to teach OCP C and OCP D concurrently when the language chosen does not have core components traditionally taught separately from Web components (e.g. ColdFusion and PHP).

The curriculum may be used to provide instruction to students that will prepare them to take certification exams offered by industry.

- II. **PROGRAM STRUCTURE:** This program is a planned sequence of instruction consisting four occupational completion points. A student who completes the applicable competencies at any occupational completion point may either continue with the training program or exit as an occupational completer.

The following diagram illustrates the Web Programming Services program structure:



- III. **LABORATORY ACTIVITIES:** Laboratory activities are an integral part of this program and include the use of computers, computer software, and peripheral equipment. Dependable high-speed access to the Internet is required.

- IV. **SPECIAL NOTES:** Phi Beta Lambda (Postsecondary), and Business Professionals of America are the appropriate Career Student Organizations (CTSO) for providing leadership training and for reinforcing specific career and technical skills. Career Student Organizations, when provided, shall be an integral part of the career and technical instructional program, and the activities of such organizations are defined as part of the curriculum in accordance with Rule 6A-6.065, FAC.

Cooperative training - OJT is appropriate for this program. Whenever cooperative training - OJT is offered, the following are required for each student: a training plan, signed by the student, teacher, and employer, which includes instructional objectives and a list of on-the-job and in-school learning experiences; a workstation that reflects equipment, skills, and tasks that are relevant to the occupation which

the student has chosen as a career goal. The student must receive compensation for work performed.

In accordance with Rule 6A-10.040, FAC., the minimum basic skills grade levels required for postsecondary adult vocational students is: Mathematics 11.0, Language 11.0, Reading 11.0. These grade level numbers correspond to a grade equivalent score obtained on a state designated basic skills examination.

This program may be offered in courses. Vocational credit shall be awarded to the student on a transcript in accordance with Section 230.643 F.S.

The standard length of this program is 1200 hours.

To be transferable statewide between institutions, this program must have been reviewed, and a "transfer value" assigned the curriculum content by the appropriate Statewide Course Numbering System discipline committee. This does not preclude institutions from developing specific program or course articulation agreements with each other.

When a secondary student with a disability is enrolled in a vocational class with modifications to the curriculum framework, the particular outcomes and student performance standards, which the student must master to earn credit, must be specified on an individual basis. The job or jobs for which the student is being trained should be reflected in the student's desired post school outcome statement on the Transition Individual Educational Plan (Transition IEP).

SCANS Competencies: Instructional strategies for this program must include methods that require students to identify, organize, and use resources appropriately; to work with each other cooperatively and productively; to acquire and use information; to understand social, organizational, and technological systems; and to work with a variety of tools and equipment. Instructional strategies must also incorporate the methods to improve students' personal qualities and high-order thinking skills.

21st-Century Skills: Instructional strategies for this program must include methods that require students to acquire 1. Digital-Age Literacy - basic scientific, mathematical, and technological literacies - visual and information literacies - cultural literacy and global awareness; 2. Inventive Thinking - adaptability/ability to manage complexity - curiosity, creativity, and risk taking - higher order thinking and sound reasoning; 3. Effective Communication - teaming, collaboration, and interpersonal skills - personal and social responsibility - interactive communication; 4. High Productivity - ability to prioritize, plan, and manage for results - effective use of real-world tools - relevant, high-quality products.

Equipment List: A generic equipment list is available for this program.

V. INTENDED OUTCOMES: After completing the following competencies, the student will be able to:

OCCUPATIONAL COMPLETION POINT - DATA CODE A

WEB DEVELOPER ASSISTANT - (Industry Title) SOC Code 15-1051

After completing the following competencies, the student will have achieved the occupational completion point of Web Developer Assistant and the student will be able to:

- 01.0 Demonstrate proficiency navigating the Internet, the WWW and other TCP/IP-based networks.
- 02.0 Demonstrate proficiency using HTML and XHTML to create Web content.
- 03.0 Demonstrate proficiency using Cascading Style Sheets (CSS) to format Web pages.
- 04.0 Demonstrate proficiency using basic client-side scripting to control the content and the behavior of HTML and XHTML documents.

OCCUPATIONAL COMPLETION POINT - DATA CODE B

DATABASE SPECIALIST - (Industry Title) SOC Code 15-1061

After completing the following competencies, the student will have achieved the occupational completion point of Database Specialist and the student will be able to:

- 05.0 Develop an awareness of the general definition, history and main features of the Relational Database Model.
- 06.0 Compare the Relational Database Model with the Hierarchical Database Model.
- 07.0 Develop an awareness of the most popular RDBMS software products currently available and the main features of each product.
- 08.0 Develop skills related to the process of determining an organization's current and future data storage needs as they relate to the organization of a database.
- 09.0 Develop an understanding of database terminology related to conceptual database models and physical databases.
- 10.0 Design Entity Relationship Models (ERMS) that accurately reflect the information needs of an organization.
- 11.0 Verify the ERM design by applying the rules of normalization (e. g. first, second and third normal form).
- 12.0 Create relational database tables based on ERMS using RDBMS software.
- 13.0 Test relational database designs by adding sample data to the database tables, verifying that data integrity is maintained and business rules are enforced.
- 14.0 Demonstrate proficiency writing and executing SQL (Structured Query Language) Select statements to retrieve data using a SQL language interface and reporting tool.
- 15.0 Demonstrate proficiency manipulating data by using Data Manipulation Language (DML) statements and transaction control statements.
- 16.0 Demonstrate proficiency creating, altering and deleting database objects using SQL Data Definition Language (DDL) statements. (e. g. tables, constraints, views synonyms, indexes, sequences)
- 17.0 Demonstrate an understanding of basic database security concepts.
- 18.0 Demonstrate the ability to develop data-centric programs to manipulate data using a language that extends SQL (such as PL/SQL).
- 19.0 Demonstrate the ability to create, execute and maintain database program units including stored procedures and database triggers.

OCCUPATIONAL COMPLETION POINT - DATA CODE C

JUNIOR APPLICATION DEVELOPER - (Industry Title) SOC Code 15-1021

After completing the following additional competencies, the student will have achieved the occupational completion point of Junior Application Developer and the student will be able to:

- 20.0 Demonstrate an understanding of basic concepts related to the development of software.
- 21.0 Demonstrate an understanding of the key features of a Web/database connectivity middleware computer programming language.
- 22.0 Write, compile (if applicable), and execute simple programs utilizing fundamental programming constructs.
- 23.0 Write programs that utilize arrays and other data structures (e.g., vectors, lists, structures, etc.).
- 24.0 Design and develop programs structured on an Object Oriented model.
- 25.0 Incorporate error handling in programs.
- 26.0 Write programs that write data to files and read data from files.

OCCUPATIONAL COMPLETION POINT - DATA CODE D

WEB APPLICATION DEVELOPER - (Industry Title) SOC Code 15-1021

Intended outcomes of OCP A and OCP B must be completed previously; OCP C may be taught concurrently. After completing the following additional competencies, the student will have achieved the occupational completion point of Internet Application Programmer and the student will be able to:

- 30.0 Demonstrate a basic understanding of the role of Web/database connectivity middleware in the development of dynamic Web content.
- 31.0 Develop basic Web applications.
- 32.0 Develop database-driven Web applications.
- 33.0 Develop Web applications that incorporate session tracking.
- 34.0 Implement Web application security techniques.
- 35.0 Demonstrate the use of an architectural model to design Web applications.
- 36.0 Demonstrate the ability to create and consume XML-based Web services.

Florida Department of Education
STUDENT PERFORMANCE STANDARDS

Program Title:

Postsecondary Number: B079200

OCCUPATIONAL COMPLETION POINT - DATA CODE A

WEB DEVELOPER ASSISTANT - (Industry Title) SOC Code 15-1051

01.0 DEMONSTRATE PROFICIENCY NAVIGATING THE INTERNET, THE WWW, AND OTHER TCP/IP-BASED NETWORKS. - The student will be able to:

- 01.01 Identify and describe Web terminology.
- 01.02 Describe the history of the Internet.
- 01.03 Describe the difference between a client and a server.
- 01.04 Describe the difference between the Internet, the WWW, and private intranets.
- 01.05 Describe the different methods by which information may be accessed on the Internet or on an intranet (e.g., HTTP, FTP).
- 01.06 Locate information on the Internet using a Web browser.
- 01.07 Demonstrate proficiency in using the basic features of GUI browsers (e.g., setting bookmarks, basic configurations, e-mail configurations, address book).
- 01.08 Identify and use Internet search engines and directories to locate information.
- 01.09 Describe the syntax, usage, and construction of Uniform Resource Locaters (URLs).
- 01.10 Demonstrative proficiency in using an FTP client to transfer and manage files and directories.

02.0 DEMONSTRATE PROFICIENCY USING HTML AND XHTML TO CREATE WEB CONTENT - The student will be able to:

- 02.01 Use storyboarding techniques for designing a Web site (e.g., linear, hierarchical).
- 02.02 Identify elements of a Web page.
- 02.03 Create Web pages using HTML and XHTML tags that create basic elements (e.g., links, lists, formatted text, tables).
- 02.04 Create Web pages that utilize tables to achieve complex layout.
- 02.05 Add graphic content to Web pages.
- 02.06 Develop, integrate, and apply the use of frames in Web site design.
- 02.07 Create Web pages that utilize client-side image maps.
- 02.08 Develop, integrate, and apply the use of forms in Web site design.
- 02.09 Optimize Web content for desirable search engine placement.
- 02.10 Demonstrate an understanding of browser compatibility issues by designing pages that comply with the current Web Content Accessibility Guidelines issued by the World Wide Web Consortium (W3C).
- 02.11 Demonstrate an understanding of Web accessibility issues by developing pages that meet Bobby standards.
- 02.12 Explain basic XML syntax and how XHTML conforms to the XML standard.
- 02.13 Use a WYSIWYG editor to develop and manage a Web site.
- 02.14 Use markup validation tools to test HTML and XHTML documents for well-formedness and make all corrections necessary to ensure compliance with W3C standards.

- 02.15 Analyze and modify HTML and XHTML source code developed by others.
- 03.0 DEMONSTRATE PROFICIENCY USING CASCADING STYLE SHEETS (CSS) TO FORMAT WEB PAGES. - The student will be able to:
- 03.01 Explain the advantages and disadvantages of using Cascading Style Sheets (CSS) to format Web pages.
 - 03.02 Describe the difference between linked, embedded, imported and inline styles and explain how styles are inherited.
 - 03.03 Utilize CSS properties within Web pages to control page layout, fonts, colors, backgrounds, and other presentation effects.
- 04.0 DEMONSTRATE PROFICIENCY USING BASIC CLIENT-SIDE SCRIPTING TO CONTROL THE CONTENT AND THE BEHAVIOR OF HTML AND XHTML DOCUMENTS. - The student will be able to:
- 04.01 Describe the difference between server-side and client-side processing.
 - 04.02 Describe the term "scripting language" and explain how scripting languages differ from compiled languages.
 - 04.03 Code a simple script within a Web page.
 - 04.04 Declare, assign values to, and utilize variables within a script.
 - 04.05 Code and utilize functions of various types including ones that use parameters and return values.
 - 04.06 Utilize mathematical, assignment, comparison and logical operators within script code.
 - 04.07 Utilize conditional statements and looping structures within script code.
 - 04.08 Define and access arrays and array elements within a script.
 - 04.09 Code scripts that make use of event handlers.
 - 04.10 Code scripts that perform validation on HTML/XHTML form elements.

OCCUPATIONAL COMPLETION POINT - DATA CODE B
DATABASE SPECIALIST - (Industry Title) SOC Code 15-1061

- 05.0 DEVELOP AN AWARENESS OF THE GENERAL DEFINITION, HISTORY AND MAIN FEATURES OF THE RELATIONAL DATABASE MODEL. - The student will be able to:
- 05.01 Briefly discuss the history of the development of the Relational Database Model.
 - 05.02 Identify the advantages of the Relational Database Model related to data integrity, data independence from application code, and ease of data retrieval.
- 06.0 COMPARE THE RELATIONAL DATABASE MODEL WITH THE HIERARCHICAL DATABASE MODEL. - The student will be able to:
- 06.01 Discuss the basic concepts of the Hierarchical Database Model.
 - 06.02 Identify the limitations of the Hierarchical Database Model, including problems related to data redundancy and interdependence with application code.
- 07.0 DEVELOP AN AWARENESS OF THE MOST POPULAR RDBMS SOFTWARE PRODUCTS CURRENTLY AVAILABLE AND THE MAIN FEATURES OF EACH PRODUCT. - The student will be able to:

- 07.01 List several shared-file database products and the advantages and disadvantages of their use.
 - 07.02 List several client/server database products and the advantages and disadvantages of their use.
 - 07.03 List factors that might determine the selection of a particular database product for a particular application.
- 08.0 DEVELOP SKILLS RELATED TO THE PROCESS OF DETERMINING AN ORGANIZATION'S CURRENT AND FUTURE DATA STORAGE NEEDS AS THEY RELATE TO THE ORGANIZATION OF A DATABASE. – The student will be able to:
- 08.01 Analyze existing databases, paper-based databases, and business processes used to collect and utilize data.
 - 08.02 Conduct interviews with database users and management in order to determine an organization's data management requirements and business rules.
 - 08.03 Compile a list of the fundamental data requirements of an organization that can function as the starting point for the design of a new database.
- 09.0 DEVELOP AN UNDERSTANDING OF DATABASE TERMINOLOGY RELATED TO CONCEPTUAL DATABASE MODELS AND PHYSICAL DATABASES. – The student will be able to:
- 09.01 Define terms used to describe an Entity Relationship Model.
 - 09.02 Define terms used to describe physical database values (e. g., null, not null, empty string, etc.).
 - 09.03 Define terms related to the structure of a physical database.
 - 09.04 Define terms related to the relationships of physical database structures.
 - 09.05 Define terms related to general, physical (including data types, length, etc.), and logical field specification elements.
- 10.0 DESIGN BASIC ENTITY RELATIONSHIP MODELS (ERMS) THAT ACCURATELY REFLECT THE INFORMATION NEEDS OF AN ORGANIZATION. – The student will be able to:
- 10.01 Discuss the advantages and goals of conceptual modeling.
 - 10.02 Determine the entities and attributes required to construct a particular ERM.
 - 10.03 Determine the attribute or attributes that uniquely identify entities.
 - 10.04 Determine the degree of the relationships (e.g., one-to-one, one-to-many, and many-to-many) and the optionality of relationships between the entities.
 - 10.05 Resolve many-to-many relationships that can be resolved through the use of hidden entities.
 - 10.06 Construct ERM diagrams that reflect the entities, attributes and relationships required to produce a conceptual model of a database.
 - 06.07 Design ERMs that correctly model the dimension of time as required by transactions and/or the maintenance of historical data.
 - 06.08 Design ERMs that correctly model network structures such as a Bill of Material.
 - 06.09 Design ERMs that correctly model hierarchical structures.
- 11.0 VERIFY THE ERM DESIGN BY APPLYING THE RULES OF NORMALIZATION (E. G. FIRST, SECOND AND THIRD NORMAL FORM). – The student will be able to:
- 11.01 Discuss the principles, advantages, and purpose of database normalization.

- 11.02 Following the first normal form, verify that all attributes of an entity in an ERM are single-valued.
 - 11.03 Following the second normal form, verify that all attributes of an entity in an ERM are dependent on the attribute(s) that uniquely identify the entity.
 - 11.04 Following the third normal form, verify that no attribute that is not the unique identifier is dependent on any other attribute that is also not the unique identifier.
- 12.0 CREATE RELATIONAL DATABASE TABLES BASED ON ERMS USING RDBMS SOFTWARE. – The student will be able to:
- 12.01 Use an RDMS management tool to create and configure the tables required to physically represent the entities depicted in an ERM.
 - 12.02 Use an RDMS management tool to create, properly name, and configure the table columns to physically represent the attributes of an entity.
 - 12.03 Use an RDMS management tool to map the relationships depicted in an ERM through the use of primary keys, foreign keys, and intersection (or linking) tables.
 - 12.04 Use an RDMS management tool to apply the necessary constraints by configuring column specifications.
 - 12.05 Determine which columns are suitable for indexing and use an RDMS management tool to configure the proper type of index.
- 13.0 TEST RELATIONAL DATABASE DESIGNS BY ADDING SAMPLE DATA TO THE DATABASE TABLES, VERIFYING THAT DATA INTEGRITY IS MAINTAINED AND BUSINESS RULES ARE ENFORCED. – The student will be able to:
- 09.01 Gather sample data and use an RDMS management tool to enter data into database tables.
 - 09.02 Evaluate the database confirming that all required information is properly represented and business rules are enforced.
 - 09.03 Evaluate the database assuring that no data redundancy exists in the database.
 - 09.04 Evaluate the database assuring that no data is derivable from other data.
 - 09.05 Discuss and defend the database design.
- 14.0 DEMONSTRATE PROFICIENCY WRITING AND EXECUTING SQL (STRUCTURED QUERY LANGUAGE) SELECT STATEMENTS TO RETRIEVE DATA USING A SQL LANGUAGE INTERFACE AND REPORTING TOOL. – The student will be able to:
- 14.01 Demonstrate proficiency customizing, configuring and using a SQL language interface and reporting tool.
 - 14.02 Code and execute basic select statements that return a varying number of columns.
 - 14.03 Code and execute select statements that limit the rows returned using a where clause with comparison and logical operators.
 - 14.04 Code and execute select statements that sort data based on single and multiple columns using an order by clause.
 - 14.05 Demonstrate the use of column aliases.
 - 14.06 Use arithmetic operators in select statements to create expressions on numeric and date data.
 - 14.07 Use concatenation operators in select statements to link columns and/or literal strings.
 - 14.08 Use the distinct keyword in select statements to eliminate duplicate rows.
 - 14.09 Use single rows functions in select statements in order to manipulate data values.

- 14.10 Use conversion functions in select statements in order to explicitly convert from one data type to another data type.
 - 14.11 Code and execute select statements to access data from multiple tables using equality, non-equality and outer joins.
 - 14.12 Code and execute select statements that perform self-joins.
 - 14.13 Code and execute select statements that incorporate single-row and multiple-row subqueries.
 - 14.14 Execute SQL statements that utilize group functions to aggregate data.
- 15.00 DEMONSTRATE PROFICIENCY MANIPULATING DATA BY USING DATA MANIPULATION LANGUAGE (DML) STATEMENTS AND TRANSACTION CONTROL STATEMENTS. – The student will be able to:
- 15.01 Code and execute DML statements to insert rows of data into an existing table.
 - 15.02 Code and execute DML statements to modify and remove existing rows in a table.
 - 15.03 Describe a database transaction and identify types of statement and conditions that will implicitly end a transaction.
 - 15.04 Code and execute transaction control statements to explicitly end or rollback a transaction.
- 16.00 DEMONSTRATE PROFICIENCY CREATING, ALTERING AND DELETING DATABASE OBJECTS USING SQL DATA DEFINITION LANGUAGE (DDL) STATEMENTS. – The student will be able to:
- 16.01 Describe the concept of database schema and access database objects that exist in different schemas.
 - 16.02 Describe the primary types of database objects including tables, views, sequences, synonyms and indexes.
 - 16.03 Code and execute DDL statement to create tables assigning proper data types and applying constraints.
 - 16.04 Code and execute DDL statements to create, drop, and rename database objects.
- 17.0 DEMONSTRATE AN UNDERSTANDING OF BASIC DATABASE SECURITY CONCEPTS. – The student will be able to:
- 17.01 Explain and describe the basic concepts of database system level security and database object level security.
 - 17.02 Code and execute (Data Control Language) DCL statements to create users, grant privileges and revoke privileges on database objects.
- 18.0 DEMONSTRATE THE ABILITY TO DEVELOP DATA-CENTRIC PROGRAMS TO MANIPULATE DATA USING A LANGUAGE THAT EXTENDS SQL (SUCH AS PL/SQL). – The student will be able to:
- 18.01 Explain the benefits of using a language that extends SQL in order to achieve more sophisticated processing of data.
 - 18.02 Properly name, declare and initialize variables and constants using the data types specified by the language.
 - 18.03 Code and execute executable statements using proper syntax.
 - 18.04 Use functions provided by the language that extend SQL including functions that convert data types.
 - 18.05 Embed SQL select statements, DML statements and transaction control statements in blocks of code.
 - 18.06 Use constructs such as if statements and loops to control program flow.
 - 18.07 Utilize complex data types provided by the language as required.

18.08 Trap and handle exceptions.

19.0 DEMONSTRATE THE ABILITY TO CREATE, EXECUTE AND MAINTAIN DATABASE PROGRAM UNITS INCLUDING STORED PROCEDURES AND DATABASE TRIGGERS. – The student will be able to:

19.01 Explain the uses and benefits of database stored procedures and triggers.

19.02 Create, execute and maintain database-stored procedures.

19.03 Create, execute and maintain database triggers.

OCCUPATIONAL COMPLETION POINT DATA CODE C

JUNIOR WEB APPLICATION DEVELOPER - INDUSTRY TITLE SOC Code 15-1021

Intended outcomes of OCP A and B must be completed previously or concurrently.

20.0 DEMONSTRATE AN UNDERSTANDING OF BASIC CONCEPTS RELATED TO THE DEVELOPMENT OF SOFTWARE. – The student will be able to:

20.01 Demonstrate an understanding of computer hardware, a computer operating system, and software, as well as how these elements relate to each other.

20.02 Describe the basic differences between standard GUI (Graphical User Interface), text-based, and browser-based application interfaces.

20.03 Define an algorithm and explain how it relates to developing a computer program.

20.04 Explain how computers store numeric and character data.

20.05 Compare and contrast computer machine language and various high-level languages.

21.0 DEMONSTRATE AN UNDERSTANDING OF THE KEY FEATURES OF A WEB/DATABASE CONNECTIVITY MIDDLEWARE COMPUTER PROGRAMMING LANGUAGE. – The student will be able to:

21.01 Briefly describe the history of the development of a programming language.

21.02 Compare and contrast the advantages, disadvantages and key features of various Web/database connectivity middleware products and languages.

21.03 Use product documentation and Internet resources as research tools to obtain detailed information about a Web/database connectivity middleware product.

22.0 WRITE, COMPILE (IF APPLICABLE), AND EXECUTE SIMPLE PROGRAMS UTILIZING FUNDAMENTAL PROGRAMMING CONSTRUCTS. – The student will be able to:

22.01 Install and configure software tools for coding and managing programs (e.g., IDE or text editor).

22.02 Code and execute a basic program using proper formatting and syntax.

22.03 Include comments to embed explanatory information in a program.

22.04 Properly declare, assign values, and utilize variables and constants of various data types.

22.05 Build expressions using variables, literals and operators.

22.06 Code, call and utilize different types of methods (or functions), including those that use parameters and return values.

22.07 Display information using methods that print to the screen.

- 22.08 Use mathematical, assignment, comparison and logical operators to manipulate values and perform comparisons in program code.
 - 22.09 Use conditional and nested conditional statements in program code.
 - 22.10 Use for and while looping structures in program code.
 - 22.11 Use nested loops that incorporate break/continue statements and labeled break/continues statements.
 - 22.12 Use advanced conditional constructs such as switch statements.
 - 22.13 Utilize debugging tools and techniques to locate the source of errors in program code.
- 23.0 WRITE PROGRAMS THAT UTILIZE ARRAYS AND OTHER DATA STRUCTURES (E.G., VECTORS, LISTS, STRUCTURES ETC.).
– The student will be able to:
- 23.01 Create single and multidimensional arrays and other data structures particular to the language.
 - 23.02 Assign values to arrays and other data structures particular to the language.
 - 23.03 Access elements using associated methods both sequentially and randomly.
- 24.0 DESIGN AND DEVELOP PROGRAMS STRUCTURED ON AN OBJECT-ORIENTED MODEL. – The student will be able to:
- 24.01 Demonstrate an understanding of objects, how objects implement abstraction and encapsulation, and how objects interact with each other through messaging.
 - 24.02 Declare classes from which objects can be instantiated that include constructors, instance variables (or attributes) and instance methods.
 - 24.03 Instantiate objects from classes and call the instance methods in an object's class.
 - 24.04 Explain the difference between primitive data types and object data types.
 - 24.05 Derive new classes utilizing the concept of inheritance.
 - 24.06 Design and code classes that contain static variables (or attributes) and static methods.
 - 24.07 Utilize access modifiers to control access to variables and methods.
 - 24.08 Design and code classes that utilize advanced Object Oriented concepts such as polymorphism, abstract classes and interfaces.
 - 24.09 Develop a model for a simple application constructed from objects using a notational language such as UML.
- 25.0 INCORPORATE ERROR HANDLING IN PROGRAMS. – The student will be able to:
- 25.01 Identify common runtime errors and their basic causes.
 - 25.02 Identify code blocks having the potential of producing a runtime error.
 - 25.03 Incorporate error-handling code into programs to effectively catch and handle error conditions.
- 26.0 WRITE PROGRAMS THAT WRITE DATA TO FILES AND READ DATA FROM FILES.
– The student will be able to:
- 25.01 Demonstrate an understanding of file input and output, the use of streams and I/O errors.
 - 25.02 Describe the difference between text files and binary files.
 - 25.03 Code classes designed to write to and read from text files.
 - 25.04 Code classes designed to write to and read from binary files.

OCCUPATIONAL COMPLETION POINT DATA CODE - D

WEB APPLICATION DEVELOPER - INDUSTRY TITLE SOC Code 15-1021

Intended outcomes of OCP A and OCP B must be completed previously; OCP C may be taught concurrently.

30.0 DEMONSTRATE A BASIC UNDERSTANDING OF THE ROLE OF WEB/DATABASE CONNECTIVITY MIDDLEWARE IN THE DEVELOPMENT OF DYNAMIC WEB CONTENT. - The student will be able to:

- 30.01 Describe the difference between static Web content and Web applications.
- 30.02 Describe the function of a Web server and how it processes static Web pages and dynamic Web pages.
- 30.03 Describe the characteristics of the HTTP protocol demonstrating an understanding of HTTP requests and HTTP responses.
- 30.04 Compare and contrast competing technologies and languages used to produce dynamic Web content.

31.0 DEVELOP BASIC WEB APPLICATIONS. - The student will be able to:

- 31.01 Write dynamic code embedded within HTML/XHTML code utilizing fundamental elements such as variables, looping structures, conditional processing, data structures, etc.
- 31.02 Properly document code using comments and other standard methods.
- 31.03 Write code to validate and process HTML/XHTML form elements.
- 31.04 Incorporate error-handling techniques, including custom error pages, into a basic Web application.
- 31.05 Effectively troubleshoot Web application code by properly evaluating error messages generated by the language environment and the Web server.

32.0 DEVELOP DATABASE-DRIVEN WEB APPLICATIONS. - The student will be able to:

- 32.01 Obtain and install database drivers.
- 32.02 Establish a connection with a database.
- 32.03 Write applications that incorporate SQL code to return and display a result set.
- 32.04 Write applications that insert, update, and delete data in the database.
- 32.05 Write applications that implement data drill-down interfaces and display returned data using "next-n" records pagination.
- 31.05 Effectively troubleshoot Web application code by properly evaluating error messages generated by the database driver.

33.0 DEVELOP WEB APPLICATIONS THAT INCORPORATE SESSION TRACKING. - The student will be able to:

- 33.01 Describe the term "stateless" as it applies to the HTTP protocol and explain why session tracking is necessary.
- 33.02 List several common types of Web applications that require the use of session tracking.
- 33.03 Describe how sessions are tracked using cookies and URL rewriting.
- 33.04 Write code that can write, read, manipulate and delete cookies.
- 33.05 Create Web applications that utilize cookies for session tracking.
- 33.06 Create Web applications that utilize URL rewriting for session tracking.

- 34.0 IMPLEMENT WEB APPLICATION SECURITY TECHNIQUES. - The student will be able to:
- 34.01 Describe basic concepts related to the SSL (Secure Sockets Layer) protocol.
 - 34.02 Describe the process of authentication in providing a secure connection.
 - 34.03 Identify common certificate authorities that issue digital secure certificates.
 - 34.04 Write a Web application the uses form-based login authentication in combination with session tracking to restrict access.
- 35.0 DEMONSTRATE THE USE OF AN ARCHITECTURAL MODEL TO DESIGN WEB APPLICATIONS. - The student will be able to:
- 35.01 Identify and define the three logical tiers of a Web application: the presentation layer or view, the control layer, and the application logic layer.
 - 35.02 Describe the benefits of the use of an architectural model in providing structure for a Web application.
 - 35.03 Design and develop an application using an architectural model.
- 36.0 DEMONSTRATE THE ABILITY TO CREATE AND CONSUME XML-BASED WEB SERVICES. - The student will be able to:
- 36.01 Define XML (Extensible Markup Language) and discuss its uses and benefits.
 - 36.02 Write well-formed and valid XML markup.
 - 36.03 Identify rendering and transformation techniques for XML.
 - 36.04 Define the term "Web Service" and describe the use and benefits of XML-based Web Services.
 - 36.05 Write a Web application that consumes an XML-based Web Service.
 - 36.06 Create a basic XML-based Web Service.