

# Career & Technical Education | Information Technology

## Database Applications Development

**Subject Code: 145085**

### Outcome & Competency Descriptions

#### Course Description:

Students will use developer strategies to manipulate data, present database systems theory, and develop database applications. Students will learn to import and export data, manipulate table properties, make advanced queries, and run basic SQL forms and reports. Students will develop macros for automating database tasks and building menu-driven applications. Knowledge and skills of data modeling, diagramming, query writing, and design theory will be developed.

#### Strand 1. Business Operations/21st Century Skills

Learners apply principles of economics, business management, marketing, and employability in an entrepreneur, manager, and employee role to the leadership, planning, developing, and analyzing of business enterprises related to the career field.

#### Outcome: 1.1. Employability Skills

Develop career awareness and employability skills (e.g., face-to-face, online) needed for gaining and maintaining employment in diverse business settings.

#### Competencies

- 1.1.1. Identify the knowledge, skills, and abilities necessary to succeed in careers.
- 1.1.2. Identify the scope of career opportunities and the requirements for education, training, certification, licensure, and experience.
- 1.1.4. Describe the role and function of professional organizations, industry associations, and organized labor and use networking techniques to develop and maintain professional relationships.
- 1.1.7. Apply problem-solving and critical-thinking skills to work-related issues when making decisions and formulating solutions.

#### Outcome: 1.2. Leadership and Communications

Process, maintain, evaluate, and disseminate information in a business.  
Develop leadership and team building to promote collaboration.

#### Competencies

- 1.2.5. Communicate information (e.g., directions, ideas, vision, workplace expectations) for an intended audience and purpose.
- 1.2.7. Use problem-solving and consensus-building techniques to draw conclusions and determine next steps.

**Outcome: 1.3.****Business Ethics and Law**

Analyze how professional, ethical, and legal behavior contributes to continuous improvement in organizational performance and regulatory compliance.

**Competencies**

- 1.3.7. Identify the labor laws that affect employment and the consequences of noncompliance for both employee and employer (e.g., harassment, labor, employment, employment interview, testing, minor labor laws, Americans with Disabilities Act, Fair Labor Standards Acts, Equal Employment Opportunity Commission [EEOC]).
- 1.3.8. Verify compliance with computer and intellectual property laws and regulations.

**Outcome: 1.5.****Global Environment**

Evaluate how beliefs, values, attitudes and behaviors influence organizational strategies and goals.

**Competencies**

- 1.5.6. Analyze work tasks for understanding and interpretation from a different cultural perspective.

**Outcome: 1.7.****Entrepreneurship / Entrepreneurs**

Analyze the environment in which a business operates, and the economic factors and opportunities associated with self-employment.

**Competencies**

- 1.7.13. Protect intellectual property and knowledge (e.g., copyright, patent, trademark, trade secrets, processes).

## **Strand 2. IT Fundamentals**

Learners apply fundamental principles of IT, including the history of IT and its impact on society, common industry terms, systems theory, information storage and retrieval, database management, and computer hardware, software, and peripheral device configuration and installation. This base of knowledge and skills may be applied across the career field.

### **Outcome: 2.8 Databases**

Describe the fundamentals of databases.

#### **Competencies**

- 2.8.1. Identify types of databases (e.g. Relational, Object-oriented, NoSQL, Graph, Data Warehouse, Distributed, Open Source, Cloud, Artificial Intelligence).
- 2.8.2. Describe the use and purpose of a database and a Database Management System (DBMS).
- 2.8.3. Compare database structures (e.g., flat file, hierarchical, relational, data lakes, object-oriented, cloud, multi-modal).
- 2.8.4. Describe the elements of a database (e.g., table, record/row, field, relationships, transactions, schema, normalization, keys).
- 2.8.5. Describe the elements of the database front-end that allow users to access, modify, delete, or insert data. (e.g., form, filters, reports)
- 2.8.6. Describe Structured Query Language (SQL).
- 2.8.7. Describe how data can be stored in and extracted from a database.
- 2.8.8. Explain the importance of data integrity and security.
- 2.8.9. Differentiate between a front-end interface and a back-end database.

### **Outcome: 2.14. Artificial Intelligence**

Understand and apply prescribed methods of using Artificial Intelligence.

#### **Competencies**

- 2.14.1. Describe how machine learning and neural networks operate differently than standard decision trees.
- 2.14.2. Analyze how artificial intelligence technology impacts society and the ethical implications of its usage.

### **Strand 3. Information Security**

Learners apply principles of information security to implement and maintain security compliance and network security. Learners select components and mechanisms required for a multilayer defense structure and evaluate and minimize security risks to wired and wireless networks and devices.

#### **Outcome: 3.2. General Security Compliance**

Implement and maintain general security compliance.

#### **Competencies**

3.2.1. Identify and implement data and application security.

## **Strand 5.      Programming and Software Systems**

Learners apply principles of computer programming and software development to develop code; build, test, and debug programs; create finished products; and plan, analyze, design, develop, implement, and support software applications.

### **Outcome: 5.1.      Programming Concepts**

Describe programming concepts.

#### **Competencies**

- 5.1.1. Describe how computer programs and scripts can be used to solve problems (e.g., desktop, mobile, enterprise, AI, cloud).
- 5.1.2. Explain how algorithms and data structures are used in information processing.
- 5.1.3. Model the solution using both graphic tools (e.g., flowcharts, IPO charts, UML, decision trees, logic tables), pseudocode techniques and artificial intelligence.
- 5.1.4. Describe, compare, and contrast the basics of procedural, structured, object-oriented (OO), and event-driven programming.
- 5.1.5. Describe the concepts of data management through programming languages.
- 5.1.6. Analyze the strengths and weaknesses of different languages for solving a specific problem.
- 5.1.7. Compare and contrast the functions and operations of compilers and interpreters.
- 5.1.8. Describe version control and the relevance of documentation.

### **Outcome: 5.2.      Computational and String Operations**

Develop code that performs computational and string operations.

#### **Competencies**

- 5.2.1. Compare and contrast primitive types of numeric and nonnumeric data (e.g., integers, floats, Boolean, strings).
- 5.2.2. Identify the scope of data (e.g., global versus local, variables, constants, arrays).
- 5.2.3. Write code that uses arithmetic operations.

- 5.2.4. Write code that applies string operations (e.g., concatenation, pattern matching, substring).

**Outcome: 5.3. Logical Operations and Control Structures**

Develop code that uses logical operations and control structures.

**Competencies**

- 5.3.1. Explain Boolean logic.
- 5.3.2. Solve a truth table.
- 5.3.3. Write code that uses logical operators (e.g., and, or, not).
- 5.3.4. Write code that uses relational operators and compound conditions.
- 5.3.5. Write code that uses conditional control structures (e.g., if, if-then-else).
- 5.3.6. Write code that uses repetition control structures (e.g., while, for).
- 5.3.7. Write code that uses selection control structures (e.g., case, switch).
- 5.3.8. Write code that uses nested structures and recursion.
- 5.3.9. Write code that creates and calls functions.
- 5.3.10. Code error handling techniques.
- 5.3.11. Write code to access data repositories.
- 5.3.12. Write code to create classes, objects, and methods.

**Outcome: 5.4. Integrated Development Environment**

Build and test a program using an integrated development environment (IDE).

**Competencies**

- 5.4.1. Configure options, preferences, and tools.
- 5.4.2. Write and edit code in the integrated development environment (IDE).
- 5.4.3. Compile or interpret a working program.
- 5.4.4. Define test cases.
- 5.4.5. Test the program using defined test cases.

- 5.4.6. Correct syntax and runtime errors.
- 5.4.7. Debug logic errors.

**Outcome: 5.5.      Programming Conventions**

Develop programs using applications security best practices according to information security policies (e.g., cross-site scripting, Structured Query Language [SQL] injection attack, bounds\_-checking).

**Competencies**

- 5.5.1. Develop programs using data validation techniques.
- 5.5.2. Develop programs that use reuse libraries.
- 5.5.3. Develop programs using operating system calls.
- 5.5.4. Develop programs that call other programs.
- 5.5.5. Use appropriate naming conventions and apply comments.
- 5.5.6. Format output (e.g., desktop, mobile, enterprise, reports, data files).
- 5.5.7. Read inputs (e.g., user input, data file, sensors, databases, APIs).

**Outcome: 5.7.      Configuration Management**

Describe configuration management activities.

**Competencies**

- 5.7.1. Explain version management and interface control.
- 5.7.2. Explain baseline and software lifecycle phases.
- 5.7.3. Analyze the impact of changes.

## **Strand 8. Databases**

Learners apply principles of designing, creating, and maintaining databases, including data storage, retrieval, modeling, manipulation, and formatting; database access, management, and administration; and database hardware and software issues.

### **Outcome: 8.1. Data Modeling**

Develop a data model to describe an application's data.

#### **Competencies**

- 8.1.1. Develop specifications for a database in consultation with the client.
- 8.1.2. Understand levels of data abstraction models (e.g., conceptual, logical, physical, and view).
- 8.1.3. Select the data model(s) based on client specifications (e.g., hierarchical, relational, object-oriented, entity-relationship, document, entity-attribute-value, star, object-relational, multidimensional, graph, multivalued, document).
- 8.1.4. Identify relationships between database entities (e.g., primary key/foreign key, nodes/relationships, key/value).
- 8.1.5. Determine the format required for data storage based upon the data model.
- 8.1.6. Determine constraints based upon the data model (e.g., null, unique, primary key, foreign key, and custom).
- 8.1.7. Normalize or denormalize the data model as appropriate for the application.
- 8.1.8. Generate data modeling documentation (e.g., entity-relationship, workflow, Unified Modeling Language [UML], data dictionary, tree).
- 8.1.9. Verify that the data model matches specifications.

### **Outcome: 8.2. Design and Creation**

Design and create databases.

#### **Competencies**

- 8.2.1. Name database objects with proper naming conventions.
- 8.2.2. Define constraints to satisfy project goals (e.g., primary key, foreign key, index).
- 8.2.3. Implement data integrity, security, encryption and regulatory restrictions (e.g., HIPPA, FERPA).

### **Outcome: 8.3. Data Entry and Access**

Enter and access data in databases.

#### **Competencies**

- 8.3.1. Collect and maintain data in the database (e.g., insert, update, delete).
- 8.3.2. Import large data sets into a database (e.g., bulk command, SQL script, CSV file).
- 8.3.3. Implement data validation (e.g., format check, range check, length check).



## **Outcome: 8.4. Database Management**

Manage databases.

### **Competencies**

- 8.4.1. House database files in an environment appropriate to anticipated user demand.
- 8.4.2. Control user access to data.
- 8.4.3. Log access to the database by user and type of transaction.
- 8.4.4. Backup, verify, and recover data.
- 8.4.5. Optimize a database for best performance (e.g. indexes, query generation, monitoring efficiency)
- 8.4.6. Implement data migration (e.g. different location, environment, file format, application).

## **Outcome: 8.5. Queries and Transactions**

Perform data queries and database transactions.

### **Competencies**

- 8.5.1. Write Structured Query Language (SQL) scripts and stored procedures.
- 8.5.2. Commit and roll back transactions.
- 8.5.3. Retrieve, filter, sort, and parse data.
- 8.5.4. Generate and print forms, reports, and results of queries (e.g., calculated fields, functions).

## **Strand 9. Cybersecurity**

Learners apply principles of Cybersecurity to secure and defend information technology systems, selection and implementation of methods and tools to secure physical and digital assets, manage threats, deploy countermeasures, and establish strategies to protect business information using risk and incident management.

### **Outcome: 9.3. Application Development Security**

Develop and maintain application security.

#### **Competencies**

- 9.3.1. Identify application vulnerabilities (e.g., Cross-site scripting, SQL injection, LDAP injection, XML injection, Directory traversal/command injection, Buffer overflow, Integer overflow, Zero-day, Cookies and attachments, Locally Shared Objects (LSOs), Flash cookies, Malicious add-ons, Session hijacking, Header manipulation, Arbitrary code execution/remote code execution).
- 9.3.3. Implement secure coding concepts (e.g., Error and exception handling, Input validation, Cross-site scripting prevention, Cross-site Request Forgery, (XSRF) prevention, OWASP).
- 9.3.4. Implement secure application configuration (e.g., Application hardening, Application patch management).
- 9.3.5. Discover and mitigate common database vulnerabilities and attacks.
- 9.3.8. Secure communication paths and data flow processes to harden the system.