



# Atlantic Technical College

## Database Application Development & Programming Program Syllabus 2017 - 2018



**Instructor Name:** Chandrakasan Iyar  
**Department Name:** Business and Information Technology  
**Office/Classroom Location:** Building 7 Room 176  
**Phone Number:** 754-321-5100 ext. 493-3043  
**Email Address:** Chandrakasan.iyar@browardschools.com

**Instructor Office Hours:**  
**M-F:** 6:30 am to 7:00 am  
 2:00 pm to 3:30 pm by appointment

<b>Student Hours:</b> <b>Monday – Friday</b> 7:00 am – 10:00 am HS 10:15 am – 1:45 pm HS <b>Lunch</b> 11:00 to 11:35 am	<b>Program Name:</b> <i>Database Application Development and Programming – High School</i>		
	<b>OCPs</b> A B C D	<b>Course Number : Course Name</b> <b>OTA0040</b> : Information Technology Assistant <b>CTS0041</b> : Computer Programmer Assistant <b>CTS0044</b> : Computer Programmer <b>CTS0062</b> : Database Programmer	<b>Hours/ Days</b> 150 / M-F 300 / M-F 150 / M-F 600 / M-F

**Technical College Policy/Adult Student Attendance:**

- A student must be withdrawn after being absent for six (6) consecutive days.
- Two (2) additional absences may be allowed under certain circumstances with appropriate documentation.
- Please refer to the Student Handbook for postsecondary students.

<http://www.atlantictechcollege.edu/atc-student-handbook/>

**Magnet High School/Attendance Policy:**

A student who has had at least five unexcused absences, or absences for which the reasons are unknown, within a calendar month, or 10 unexcused absences, or absences for which the reasons are unknown, within a 90-calendar-day period, may be exhibiting a pattern of non-attendance (F.S.1003.26 (1) (b)) and the School Board of Broward County, Policy 5.5.

**Required Book(s):**

- *CIW Database Design Specialist Electronic Student Kit*
- *CIW JavaScript Specialist Electronic Student Kit*
- *Starting Out With Programming Logic and Design 4e (ISBN# 9780133985078)*

**Required Materials/Supplies:**

- *Computer Headset*
- *USB Drive / 8GB*
- *Three-ring binder with index tabs*
- *Pen, Pencil & Notebook*

*All required books & most materials/supplies can be purchased from the school bookstore. Stop by during operational hours for pricing & purchasing information.*

***Books are subject to change. Bookstore has current list.***

<p><b>Grading System:</b></p> <p>A 90 - 100%</p> <p>B 80 - 89%</p> <p>C 70 - 79%</p> <p>D 60 - 69%</p> <p>F 0 - 59%</p> <p>I Incomplete</p>	<p><b>Additional Program Specific Grading Information:</b></p> <p>Weekly Assignments 30%</p> <p>Projects &amp; Labs 30%</p> <p>Exams &amp; Test 30%</p> <p>Work Habits/Cert Prep 10%</p>
<p><b>Course Grading Policy:</b></p> <ul style="list-style-type: none"> <li>Scheduled assignments are due each Monday by 11:59 pm. Late submission of work will affect the assignment grade.</li> <li>Students assume full responsibility for the content and integrity of the academic work they submit. The guiding principle of academic integrity shall be that a student's submitted work, examinations, reports, and projects must be that of the student's own work.</li> <li>Unless otherwise stated by the instructor, external references including books, calculators, notes and/or the Internet may not be utilized during assessments or exams. Blank scratch paper will be permitted during certain assessments.</li> <li>Exams may include an oral or lab/skills component and final exams will be completed in-person during a class/lab session.</li> </ul>	
<p><b>View Your Grades:</b></p> <p><b>Grades can be viewed online by following the directions below:</b></p> <ol style="list-style-type: none"> <li>Go to <a href="https://browardfocus.com">https://browardfocus.com</a> (access FOCUS using Chrome, Firefox or Safari)</li> <li>Student ID which is on your student schedule.</li> <li>Passcode: Student's date of birth formatted as YYYYMMDD. Four digits for the year, two digits for the month and two digits for the day.</li> </ol>	
<p><b>Class Room/ Lab Rules:</b></p> <ul style="list-style-type: none"> <li>✓ No food or beverages in the lab</li> <li>✓ Cell phones on vibrate or turned off</li> <li>✓ Sign in on the attendance sheet</li> <li>✓ Follow all instructions given by ATC staff</li> <li>✓ Come to class prepared to work</li> <li>✓ Call or email instructor when absent</li> </ul>	
<p><b>Industry Credentials:</b></p> <ul style="list-style-type: none"> <li>✓ MTA Database Administration Fundamentals</li> <li>✓ CIW JavaScript Specialist</li> </ul>	<p><b>Outstanding Student Recognition Information:</b></p> <p>A gold seal will be applied to a Program Completion Certificate or an Applied Technology Diploma if the student has earned a 3.5 GPA or higher in their Career and Technical Education (CTE) classes.</p>

**Program Name:** Database Application Development & Programming

**Course Number:** OTA0040

**Course Name:** Information Technology Assistant

**Occupational Completion Point:** A

**Intended Outcomes:** (From FL DOE Curriculum Framework)

- Demonstrate knowledge, skill, and application of information systems to accomplish job objectives and enhance workplace performance.
- Develop an awareness of microprocessors and digital computers.
- Demonstrate an understanding of operating systems.
- Use technology to enhance the effectiveness of communication skills utilizing word processing applications.
- Use technology to enhance communication skills utilizing presentation applications.
- Use technology to enhance the effectiveness of communication utilizing spreadsheet and database applications.
- Use technology to enhance communication skills utilizing electronic mail.
- Investigate individual assessment and job/career exploration and individual career planning that reflect the transition from school to work, lifelong learning, and personal and professional goals.
- Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance.
- Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication.
- Demonstrate competence in page design applicable to the WWW.
- Develop an awareness of emerging technologies.
- Develop awareness of computer languages and software applications.
- Demonstrate comprehension and communication skills.

**Course Number:** CTS0041

**Course Name:** Computer Programmer Assistant

**Occupational Completion Point:** B

**Intended Outcomes:** (From FL DOE Curriculum Framework)

- Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development.
- Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types.
- Distinguish between iterative and non-iterative program control structures.
- Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages.
- Describe the processes, methods, and conventions for software development and maintenance.
- Explain the types, uses, and limitations of testing for ensuring quality control.

- Create a program design document using Unified Modeling Language (UML) or other common design tool.
- Describe information security risks, threats, and strategies associated with software development.
- Design a computer program to meet specific physical, operational, and interaction criteria.
- Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types.
- Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input.
- Create a unit test plan, implement the plan, and report the results of testing.
- Use oral and written communication skills in creating, expressing and interpreting information and ideas.
- Solve problems using critical thinking skills, creativity and innovation.
- Use information technology tools.
- Demonstrate leadership and teamwork skills needed to accomplish team goals and objectives.
- Describe the importance of professional ethics and legal responsibilities.

**Course Number:** CTS0044

**Course Name:** Computer Programmer

**Occupational Completion Point:** C

**Intended Outcomes:** (From FL DOE Curriculum Framework)

- Explain key concepts that distinguish object-oriented programming from procedural programming.
- Create a project plan for an object-oriented programming project that defines requirements, structural design, time estimates, and testing elements.
- Design, document, and create object-oriented computer programs.
- Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results

**Course Number:** CTS0062

**Course Name:** Database Programmer

**Occupational Completion Point:** D

**Intended Outcomes:** (From FL DOE Curriculum Framework)

- Develop an awareness of the changes taking place in the information age and how they fit into an evolving society.
- Develop the "big picture" of database design and how to best organize data according to business rules and/or client needs.
- Develop the process of creating an entity by identifying relationships.
- Formulate and assemble initial entity relationship by expanding on modeling concepts.
- Consider the degree and optionality of relationships of entities.
- Demonstrate proficiency in early construction stages of the data modeling process by using unique identifiers and many-to-many (M:M) relationships for building entity relationship diagrams.
- Demonstrate proficiency in advanced data constructs by analyzing business requirements and diagramming entities and relationships.
- Demonstrate proficiency in designing and adding complexity to an entity-relationship model (ERM).
- Apply complex ERM information by fine-tuning entities and the process for relating them.
- Apply initial database design and normalization by following the set of house rules that determine how items are stored and retrieved.
- Demonstrate proficiency in the technique of normalization by labeling and organizing all items in a database in such a way as to prevent any confusion or mistakes.
- Demonstrate proficiency in table normalization by combining the techniques of an entity relationship model or a top-down, business approach to data with normalization or a bottom-up mathematical approach to data.
- Apply blueprint principles to begin designing a tool for creating a web-based interface access to a database.
- Extend the logical model presentation model by normalizing the data and mapping the management system.
- Apply techniques for building a storage management system by creating a website using templates and wizards.
- Demonstrate design and functionality by constructing a group business presentation.
- Demonstrate comprehension of database modeling competency through group presentation.
- Demonstrate comprehension that the database management software is a system for organizing the storage unit (or database) according to business needs and rules, through data integrity constraints.
- Demonstrate comprehension of aspects of SQL language interface by writing basic SQL statements.
- Demonstrate proficiency working with columns, characters, and rows in SQL.
- Demonstrate proficiency in using SQL comparison operators.
- Demonstrate proficiency in using logical comparisons and precedence rules.
- Demonstrate proficiency using SQL single row functions.
- Demonstrate proficiency displaying data from multiple tables.

- Demonstrate proficiency aggregating data using group functions.
- Demonstrate proficiency utilizing subqueries.
- Demonstrate proficiency producing readable output with SQL language interface, reporting tool, and data manipulation language.
- Demonstrate proficiency creating and managing database objects.
- Demonstrate proficiency altering tables and constraints implementing views.
- Demonstrate mastery of creating and implementing views, synonyms, indexes and other database objects.
- Demonstrate ability to control user access and SQL language interface and reporting tool.
- Demonstrate comprehension of bundling features of SQL.
- Demonstrate comprehension working with composite data types by writing executable script files.
- Describe the differences between SQL and SQL extension languages.
- Create program blocks.
- Use variables in program blocks.
- Recognize lexical units.
- Recognize data types.
- Use scalar data types.
- Use various types of joins.
- Use SQL group functions and subqueries.
- Write executable statements.
- Use nested blocks and variable scope.
- Use good programming practices.
- Write DML statements to manipulate data.
- Retrieve data.
- Manipulate data.
- Use transaction control statements
- Use IF conditional control statements.
- Use CASE conditional control statements.
- Use basic LOOP iterative control statements.
- Use WHILE and FOR loop iterative control statements.
- Use nested loop iterative control statements.
- Use explicit cursors.
- Use explicit cursor attributes.
- Use cursor FOR loops.
- Use cursors with parameters.
- Use cursors for update transactions.
- Use multiple cursors.
- Handle exceptions.
- Trap server exceptions.
- Trap user-defined exceptions.
- Create procedures.
- Use parameters in procedures.
- Pass parameters.
- Create stored functions.

- Use functions in SQL statements.
- Manage procedures and functions.
- Manage object privileges.
- Use invoker's rights.
- Create packages.
- Manage package constructs.
- Use advanced package concepts.
- Manage persistent state of package variables.
- Use vendor-supplied packages.
- Understand dynamic SQL.
- Understand triggers.
- Create DML triggers.
- Create DDL and database event triggers.
- Manage triggers.
- Use large object data types.
- Manage binary types.
- Manage indexes.
- Manage dependencies.
- Program a database application.
- Utilize the basic concepts of database design.
- Utilize SQL and union queries.
- Implement program statements using objects.
- Utilize debugging tools and write error handlers.
- Demonstrate file I/O.
- Create forms and identify all the properties of a form.
- Manipulate data using object models
- Develop custom controls.
- Utilize API functions.
- Demonstrate database replication and implement database replication using programming tools.
- Analyze and implement security options.
- Implement client/server applications.
- Optimize the performance of a database.
- Perform application distribution.
- Test and debug databases.