



COMPUTER PROGRAMMING (ORACLE DATABASE)

TECHNICAL-PROFESSIONAL

Grade 12

**Three-Term Budget Of Work (BOW)
for Learning Competencies**

**TECHNICAL-PROFESSIONAL
ICT SUPPORT AND COMPUTER PROGRAMMING TECHNOLOGIES
Grade 12**

**Three-Term Budget Of Work
For Learning Competencies**

Please note that each Tech-Pro elective is completed in one term if delivered at Grade 12.

Elective	Computer Programming (Oracle Database)
Prerequisite	None
Performance Standard	The learners execute basic SQL syntax for data manipulation.
Content Standard	The learners demonstrate an understanding of concepts and components of database management systems and use of SQL basic syntax for data manipulation.

Week	Learning Competency	Suggested Activities
1	Discuss fundamental concepts and components of Oracle Database system	Data Detectives: Build the Perfect Database - Analyze real-world information needs and design a simple Oracle-based database system using proper DBMS concepts. - Divide students into groups of 4–5 members. Give each group a scenario (School Library, Online Shop, Hospital Records, Enrollment System). Students identify the needed entities, attributes, and relationships. Create a simple

Week	Learning Competency	Suggested Activities
		schema with tables and keys. Apply basic normalization (1NF to 3NF). Present how Oracle Database can manage the system better than a file system. Explain possible careers involved (DBA, Developer, Data Analyst). ● Oracle Time Machine Challenge ○ Explore the evolution of Oracle Corporation Database and compare database technologies from past to present. ○ Students create a timeline of Oracle Database history and milestones. Assign Oracle versions to groups for research. Compare features of each version. Differentiate RDBMS vs ORDBMS using examples. Discuss how Oracle created business and career opportunities. Present findings through posters or slides.
	● Discuss installation procedure of IDE for Oracle Database	● Oracle Ready, Set, Go! ○ Set up a functional Oracle Database environment by completing the installation process. ○ Check system requirements, choose an SQL

Week	Learning Competency	Suggested Activities
		platform, download the software, install the CLI or GUI tool, then run the SQL executable and show proof of successful setup. • SQL Command Sprint ○ Construct and execute basic SQL syntax using common commands correctly. ○ Open the installed SQL tool, type sample commands such as CREATE, INSERT, SELECT, UPDATE, and DELETE, then run them and record the results.
2	• Construct basic SQL syntax	• SQL Kickstart Challenge ○ Demonstrate understanding of SQL syntax by constructing correct basic SQL statements. ○ Write and run simple SQL commands such as CREATE, INSERT, and SELECT using the proper SQL syntax format. • Command Master Showdown ○ Use basic SQL commands to manage and retrieve data from a database table. ○ Create a sample table, add records, update data, delete selected entries, and display the final output using SQL commands.
	• Apply LIMIT rows syntax	• Filter Frenzy: Find the Right Rows

Week	Learning Competency	Suggested Activities
	retrieved by a query	○ Apply WHERE clauses and conditional operators to retrieve specific records from a database table. ○ Use a sample table and write SQL queries with WHERE, =, >, <, AND, OR, LIKE, BETWEEN, and IN to filter the correct data. ● Top Rows Challenge ○ Use LIMIT rows syntax and operator precedence rules to control and refine query results. ○ Create SQL queries that display only selected rows using LIMIT, combine conditions with logical operators, and compare outputs with and without parentheses
3	● Apply SORT rows syntax retrieved by a query	● Sort It Out: ORDER BY Mission ○ Apply ORDER BY syntax to sort query results in ascending and descending order. ○ Run SQL queries using ORDER BY to sort data in both ASC and DESC formats, then compare the output results. ● Multi-Sort Master Challenge ○ Use sorting techniques including column alias and multiple columns to organize query

Week	Learning Competency	Suggested Activities
		results. ○ Create SQL queries that sort data using column aliases and multiple columns, then observe how order changes based on priority
	● Apply single row functions	● Function Frenzy: Data Transformer Lab ○ Apply single-row SQL functions to manipulate character, number, and date data accurately. ○ Write SQL queries using character functions (UPPER, LOWER, INITCAP), number functions (ROUND, TRUNC, MOD), and date functions to transform sample table data. ● Data Conversion Detective Challenge ○ Use conversion functions to correctly handle and convert different data types in SQL. ○ Create queries using TO_CHAR, TO_NUMBER, and TO_DATE, apply date format models (including RR format), and demonstrate correct implicit and explicit data type conversions.
Suggested Performance Tasks	● Individual Performance Task: Oracle Master Builder: From Setup to Query ○ Apply Oracle Database concepts by installing an IDE, building database structures, and executing SQL queries using filtering, sorting, and functions. ○ executing SQL queries using filtering, sorting, and functions. ○ Discuss fundamental Oracle database concepts, DBMS/RDBMS/ORDBMS differences, and	

Week	Learning Competency	Suggested Activities
	data models, then install and configure an Oracle IDE. They design a simple database schema, apply normalization concepts, and construct basic SQL syntax. They execute queries using WHERE clauses, conditional and logical operators, LIMIT-style filtering, ORDER BY sorting (single and multiple columns), and single-row functions, documenting outputs and explanations. • Group Performance Task: SQL Enterprise Simulation: The Oracle Data Lab Challenge ○ Collaboratively design, implement, and query a database system using Oracle SQL to simulate real-world data management. ○ Groups act as a database solutions team for a company. They analyze a data scenario, design a normalized database schema (entities, attributes, relationships), and set up an Oracle environment. They write and test SQL commands, apply filtering (WHERE, operators), sorting (ORDER BY), and single-row functions to generate business reports. Teams present their system setup, query results, and explain how Oracle DBMS improves data management over traditional file systems.	
Performance Standard	The learners create a database applying multiple row functions, joins, subqueries, and substitution variables to retrieve and display data from multiple tables.	
Content Standard	The learners demonstrate an understanding of multiple row functions, joins, subqueries, and substitution variables to retrieve and display data from multiple tables.	
4	• Apply multiple row functions	• SQL Data Analyzer: Master the Numbers ○ Apply Oracle aggregate functions and GROUP BY techniques to analyze and summarize database data. ○ Students create SQL queries using AVG, SUM,

Week	Learning Competency	Suggested Activities
		MIN, MAX, COUNT, and DISTINCT on a given dataset. They organize data using GROUP BY (single and multiple columns), handle NULL values in group functions, and generate summarized reports that show meaningful data insights. • Business Intelligence Lab: The Oracle Reporting Challenge ○ Collaboratively generate business reports using advanced SQL grouping, filtering, and aggregation techniques. ○ Simulate a business analytics tasked to analyze company data. They apply aggregate functions, create grouped datasets using GROUP BY, refine outputs using HAVING clauses, and demonstrate nesting of group functions to produce decision-ready reports, then present their findings with interpretations.
	• Apply JOIN syntax in retrieving and displaying from multiple tables	• Data Detective: The SQL Aggregation Challenge ○ Apply Oracle aggregate functions and GROUP BY clauses to analyze and summarize database information accurately.

Week	Learning Competency	Suggested Activities
		- Students use a provided dataset to construct SQL queries applying AVG, SUM, MIN, MAX, COUNT, and DISTINCT. They organize results using GROUP BY (single and multiple columns), handle NULL values in group functions, and refine outputs using HAVING clauses and nested group functions to generate meaningful data summaries. Oracle Analytics Team: Business Reporting Simulation - Collaboratively generate structured business reports using advanced SQL aggregation and grouping techniques. - Act as a data analytics team tasked to analyze business records. They design SQL queries using aggregate functions, GROUP BY (including multiple columns), DISTINCT, HAVING, and nested group functions to produce insightful reports, then present findings with interpretations for decision-making.
5	Apply subqueries for complex data retrieval and manipulation	SQL Join Mastery: The Multi-Table Explorer - Apply different SQL JOIN types to retrieve and

Week	Learning Competency	Suggested Activities
		analyze data across multiple related tables. - Students write SQL queries using INNER, LEFT, RIGHT, FULL OUTER, CROSS, NATURAL, and NON-EQUI joins to combine data from multiple tables. They also apply USING and ON clauses, use table aliases, qualify ambiguous column names, and apply AND conditions to refine results for accurate data retrieval. • Database Detective Agency: The Multi-Join Investigation - Collaboratively solve data scenarios using advanced multi-table joins and relational querying techniques. - Act as a database investigation team tasked with solving real-world data problems. They design queries using multi-table joins (including self-joins and three-way joins), apply table aliases, resolve ambiguous fields, and use different JOIN types to generate complete reports, then present findings with query explanations.
	Apply substitution variables in SQL	SQL Prompt Builder: Dynamic Query Challenge

Week	Learning Competency	Suggested Activities
		○ Apply single and double substitution variables to create flexible and interactive SQL queries. ○ Students write SQL statements using single (&) substitution variables for character, date values, column names, expressions, WHERE conditions, ORDER BY clauses, and table names. They also use DEFINE and UNDEFINE commands to manage variables, then test queries with different inputs to observe dynamic results. ● Dynamic Database Simulator: The Interactive Query Lab ○ Collaboratively build reusable SQL queries using repeated substitution variables and SQL formatting tools. ○ Design interactive SQL scripts using double ampersand (&&) substitution variables, applying VERIFY and COLUMN commands to format and control output. They create reusable query templates (including entire SELECT statements), then demonstrate how dynamic inputs change query results in a simulated database system.

Week	Learning Competency	Suggested Activities
Suggested Performance Tasks	• Individual Performance Task: "Oracle Query Mastery: The Dynamic Data Analyzer" ○ Apply aggregate functions, JOINS, subqueries, and substitution variables to build flexible and complex SQL solutions for data analysis. ○ Students design SQL scripts that summarize data using group functions (AVG, SUM, COUNT, MIN, MAX), organize results with GROUP BY, HAVING, and nested aggregates, retrieve multi-table data using various JOIN types (inner, outer, self, cross, etc.), and solve complex queries using single-row and multiple-row subqueries. They enhance all queries using substitution variables (&, &&, DEFINE, UNDEFINE) to make dynamic, reusable SQL statements. • Group Performance Task: Enterprise Data Intelligence Lab: SQL Problem-Solving Challenge ○ Collaboratively design and execute advanced SQL solutions using joins, subqueries, aggregation, and dynamic query variables for real-world business scenarios. ○ Groups act as a data analytics team tasked with solving business cases. They build a relational database query system using multiple row functions with GROUP BY and HAVING, apply various JOIN types (including multi-table and self-joins), construct single-row and multiple-row subqueries (IN, ANY, ALL), and implement substitution variables to create interactive SQL reports. Teams present results with explanations of query logic and business insights.	
Performance Standard	The learners apply DML commands in creating and managing database using constraints, views, database objects, managing user access roles and set operators to combine query results.	
Content Standard	The learners demonstrate an understanding of Data Manipulation Language (DML) commands in creating and managing database using constraints, views, database objects, managing user access roles and set operators to combine query results.	

Week	Learning Competency	Suggested Activities
6	Apply DML commands in manipulating data for table attributes	SQL Data Surgeon: The Table Repair Challenge - Apply DML commands to safely manipulate and correct database records using proper transaction control. - Students perform data operations using INSERT (including NULL and date values), UPDATE, DELETE, ADD, and MERGE on a sample database. They use COMMIT, ROLLBACK, and SAVEPOINT to manage transactions and ensure data accuracy while simulating real-world data correction scenarios. Database Operations Lab: The Transaction Control Mission - Collaboratively execute and manage database changes using DML commands and transaction control techniques. - Simulate a business database team tasked to manage customer or inventory records. They perform INSERT, UPDATE, DELETE, and MERGE operations while using SAVEPOINT, ROLLBACK, and COMMIT to control transactions, ensuring data integrity before presenting the final

Week	Learning Competency	Suggested Activities
		database state and explaining their process.
	• Create tables for managing data	• Database Builder: The Table Architect Challenge ○ Apply table creation and modification commands to design and manage a structured database system. ○ Students create tables using proper naming conventions, then apply ALTER TABLE commands to add, modify, rename, and drop columns. They also manage table structures by viewing schemas, setting unused columns, and adding comments to document database design decisions. • SQL System Admin Lab: The Database Maintenance Mission ○ Collaboratively manage and maintain database tables using advanced SQL structure and optimization techniques. ○ Simulate a database administration team tasked to build and maintain a system. They create multiple tables, apply ALTER operations (rename, truncate, drop), manage indexes, and document table structures with comments,

Week	Learning Competency	Suggested Activities
		ensuring organized and optimized database management before presenting their system setup.
7	• Apply constraint commands for data validation	• Data Guardian: The Constraint Builder Challenge ○ Apply SQL key constraints to enforce data integrity in table creation and management. ○ Students design database tables using NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK, and DEFAULT constraints, ensuring proper relationships and valid data entry rules. They test sample insertions to observe how constraints enforce data accuracy and consistency. • Relational Integrity Lab: The Cascade Control Mission ○ Collaboratively implement and manage cascading constraints to maintain relational database integrity. ○ Simulate a database engineering team tasked to build related tables with foreign key relationships. They apply ON DELETE and ON UPDATE actions (CASCADE, SET NULL, SET

Week	Learning Competency	Suggested Activities
		DEFAULT), test data changes across tables, and demonstrate how cascading rules affect database integrity in real scenarios.
	• Apply VIEW syntax in accessing specific data	• Data Lens Creator: The SQL VIEW Builder Challenge ○ Apply VIEW syntax to create, use, modify, and manage different types of database views for organized data retrieval. ○ Students create SQL VIEWS using simple, complex, and inline queries, then apply modifications and dropping of views. They explore updatable, read-only, and materialized views while testing how each view type displays and organizes data from underlying tables. • Virtual Data Dashboard Lab: The VIEW Management Mission ○ Collaboratively design and manage a structured database reporting system using different types of SQL views. ○ Simulate a data reporting team tasked to build a dashboard system using multiple VIEW types (simple, complex, materialized,

Week	Learning Competency	Suggested Activities
		partitioned). They create, modify, and manage views, then demonstrate how read-only and updatable views support real-world business reporting before presenting their system output.
	• Apply database object functions for storing or referencing data	• Oracle Object Architect: Build the Complete Database System ○ Apply all major database object categories to design and build a fully functional Oracle database system. ○ Students create a complete database structure using tables, views, indexes, stored procedures, functions, triggers, sequences, schemas, synonyms, and user-defined types, demonstrating how each object works together to support data storage, retrieval, automation, and organization in a unified system. • Enterprise Database Simulation: The Oracle System Engineering Lab ○ Collaboratively design and implement a real-world database system using integrated Oracle database objects. ○ Act as a database development team tasked to

Week	Learning Competency	Suggested Activities
		build a complete enterprise system (e.g., school records or inventory system) by assigning roles to design schemas, create tables and indexes, develop views, stored procedures, functions, triggers, sequences, synonyms, and user-defined types, then present how each object contributes to system efficiency and functionality.
8	• Apply database control in managing user access	• Access Control Engineer: The Security Setup Challenge ○ Apply user access management concepts to configure secure database users and privileges. ○ Students create database users, assign system privileges, manage roles and groups, and control access rights by granting and revoking privileges while documenting how each action affects security and system access. • Cyber Security Lab: The Database Access Control Simulation ○ Collaboratively design a secure database environment by implementing user access control, roles, and auditing systems. ○ Simulate a database security team tasked to

Week	Learning Competency	Suggested Activities
		manage user access by creating multiple users, assigning roles and privileges, implementing revoke policies, and setting up auditing and monitoring procedures, then presenting how their system ensures data protection and controlled access.
	• Apply SET operators in data manipulation	• Data Merge Master: The Set Operator Challenge ○ Apply SQL set operators (UNION, UNION ALL, INTERSECT, EXCEPT) to combine and compare data from multiple queries. ○ Students write SQL queries using UNION, UNION ALL, INTERSECT, and EXCEPT to merge and filter results from different datasets, then analyze how each operator changes the output of combined data. • Data Comparison Lab: The SQL Set Operations Simulation ○ Collaboratively solve data comparison problems using SQL set operators to generate meaningful business insights. ○ Act as a data analytics team tasked to compare multiple datasets (e.g., customers, products,

Week	Learning Competency	Suggested Activities
		or enrollments) using UNION, INTERSECT, UNION ALL, and EXCEPT, then present findings showing similarities, differences, and combined results.
	• Execute GROUP BY clause commands for data manipulation	• Data Summarizer Pro: The Advanced Grouping Challenge ○ Apply advanced GROUP BY features to generate structured and meaningful data summaries. ○ Students create SQL queries using GROUP BY with ROLLUP, CUBE, GROUPING SETS, composite columns, and concatenated groupings, then refine outputs using HAVING and group functions to produce multi-level summary reports from a dataset. • Business Analytics Lab: The Oracle Data Insights Simulation ○ Collaboratively generate advanced business reports using enhanced GROUP BY operations and analytical functions. ○ Simulate a data analytics team tasked to analyze business records. They use GROUP BY enhancements (ROLLUP, CUBE, GROUPING

Week	Learning Competency	Suggested Activities
		SETS), apply HAVING filters, and interpret composite and concatenated groupings to present hierarchical and multi-dimensional reports with insights.
Suggested Performance Tasks	• Group Performance Task: ○ Apply complete Oracle database development skills from table creation to advanced querying, security, and reporting. ○ Design a full database system using tables, constraints, DML with transactions, views, database objects, user access control, SET operators, and GROUP BY to generate meaningful reports. • Group Performance Task: "Enterprise Oracle Systems Lab: The Complete Database Solution Challenge" ○ Collaboratively build a secure, functional, and analytical Oracle database system for real-world use. ○ In groups, create and manage a database system applying tables, DML, constraints, views, database objects, security controls, and data analysis tools, then present the final working system.	
Performance Standard	The learners create an application by connecting programming language (PL) to a SQL database, effectively performing Create, Read, Update and Delete (CRUD) operations.	
Content Standard	The learners demonstrate an understanding in connecting programming language (PL) to a SQL database, effectively performing Create, Read, Update and Delete (CRUD) operations.	
9	• Perform connection of PL to	• PL/SQL Code Starter: The Identifier Challenge

Week	Learning Competency	Suggested Activities
	SQL for program application	○ Apply PL/SQL fundamentals by using identifiers, variables, data types, sequences, and executable statements in a basic program. ○ Create a simple PL/SQL block that declares identifiers and variables, uses appropriate data types, generates values with sequences, and executes statements to display or process data. ● Code Logic Lab: The PL/SQL Builder Mission ○ Collaboratively develop a functional PL/SQL program using proper identifiers and programming structures. ○ In groups, design a PL/SQL script that uses variables, data types, sequences, and executable statements to solve a simple database task, then present the program flow and output.
	● Apply SQL syntax to programming .net or Java for storing or retrieval of data	● PL/SQL Query Explorer ○ Apply PL/SQL techniques in retrieving data, handling naming errors, and using cursor attributes. ○ Create a PL/SQL block that uses SELECT statements to retrieve data, applies proper

Week	Learning Competency	Suggested Activities
		naming conventions, identifies common errors, and displays results using SQL cursor attributes. • Oracle Server Mission: Data Control Lab ○ Collaboratively build a PL/SQL script that interacts with the Oracle Server efficiently and accurately. ○ Develop a PL/SQL program using SELECT queries, correct naming conventions, error checking, and cursor attributes, then present how the script retrieves and manages data.
10	• Evaluate bugs using exception handling in programming .net or Java for debugging process	• Error Hunter: The PL/SQL Rescue Challenge” ○ Apply exception handling techniques to identify and manage different PL/SQL errors. ○ Create a PL/SQL block that demonstrates handling predefined, non-predefined, and user-defined exceptions, then display custom messages for each error encountered. • Debug Team Mission: Oracle Error Control Lab ○ Collaboratively design a PL/SQL program that manages runtime errors using effective

Week	Learning Competency	Suggested Activities
		exception handling. ○ In groups, develop a PL/SQL script with scenarios involving server errors and custom exceptions, apply proper handlers, and present how the program prevents failures and improves reliability.
	● Apply store procedures on PL/SQL	● Procedure Pro: The PL/SQL Builder Challenge ○ Apply PL/SQL stored procedure concepts by creating modular subprograms with parameters and execution logic. ○ Create a stored procedure using PL/SQL blocks that demonstrates parameter modes, modular design, and procedure execution, then compare it with an anonymous block. ● Code Factory Lab: The Procedure Development Mission ○ Collaboratively design efficient PL/SQL stored procedures using layered programming techniques. ○ In groups, develop a set of stored procedures for a sample system using modularized design, parameters, and PL/SQL execution flow, then

Week	Learning Competency	Suggested Activities
		present the benefits and procedure structure.
11	• Create the PL/SQL code framework	• PL/SQL Code Kickstart Challenge ○ Apply PL/SQL coding fundamentals using data types, subtypes, collections, and large object handling. ○ Create a PL/SQL program that declares predefined data types, uses custom subtypes, stores data in collections, and performs basic operations on large objects. • Smart Coding Lab: The PL/SQL Data Builder ○ Collaboratively develop a PL/SQL solution using advanced data structures and object handling. ○ Design a PL/SQL script that uses data types, subtypes, collections, and large objects for a sample application, then present how each feature improves data processing.
	• Apply the advanced interface methods on PL/SQL framework	• PL/SQL Beyond Boundaries Challenge ○ Apply advanced interface methods by connecting PL/SQL with external procedures and Java integration. ○ Create a simple PL/SQL example that demonstrates calling an external procedure or Java interface, then explain how it extends

Week	Learning Competency	Suggested Activities
		database functionality. • Tech Fusion Lab: Oracle Integration Mission ○ Collaboratively design a system that uses PL/SQL with external and Java-based interfaces. ○ Develop a concept or prototype showing how PL/SQL can interact with external procedures and Java components, then present the benefits for real-world applications.
	• Apply the advanced interface methods on PL/SQL framework	• Speed Code Challenge: Optimize and Pitch ○ Apply PL/SQL performance tuning techniques and present the value of an optimized solution. ○ Create a PL/SQL program, improve its performance using compiler settings and caching techniques, then prepare a short business pitch explaining its efficiency and benefits. • Startup Tech Pitch: The Smart PL/SQL Solution ○ Collaboratively optimize a PL/SQL system and pitch it as a market-ready business solution. ○ In groups, enhance a PL/SQL application

Week	Learning Competency	Suggested Activities
		through tuning and caching, then present a prototype pitch highlighting performance gains, usability, and business potential.
Suggested Performance Tasks	• Individual Performance Task: “Oracle App Builder: Code to Connection” ○ Apply complete PL/SQL programming competencies by developing an optimized database application connected to Java or .NET. ○ Create a PL/SQL-based application that uses identifiers, variables, data types, SQL retrieval commands, stored procedures, exception handling, collections, advanced interfaces, and performance tuning, then demonstrate the working system. • Group Performance Task: “Enterprise Dev Lab: The Oracle Integration Challenge” ○ Collaboratively design and optimize a real-world application using PL/SQL, external interfaces, and database programming techniques. ○ In groups, develop a connected Java or .NET database system using PL/SQL procedures, data handling, debugging, advanced interfaces, and tuning methods, then present the final application and workflow.	