TOSA

Your skills. Your advantage.

Tosa SQL

Skills Framework

contact@isograd.com

Table of Contents

Table of Contents 2

Introduction to the Tosa® Skills Framework 3

- Tosa® (Test on Software Applications) 4
- Tosa® Skills Framework Objective 4
- Unique Tosa® Scoring 5
- About the Tosa® SQL certification 6
- Isograd Learning Platform 7
- Tosa® SQL Level Descriptions 8
- Business Applications 9
- Domain 1: Fundamentals and Applications 10
  - Sub-domain: Getting started with SQL 10
- Domain 2: Performance Optimization 11
  - Sub-domain 1: Create and manage indexes 11
  - Sub-domain 2: Optimize queries and analyze execution plans 12
  - Sub-domain 3: Structuring tables and schemas to improve performance 13
- Domain 3: SQL Queries 14
  - Sub-domain 1: Write SQL queries 14
  - Sub-domain 2: Manage and modify data 15
  - Sub-domain 3: Perform joins and aggregations 16
- Domain 4: Data Security 17
  - Sub-domain 1: Manage access security 17
  - Sub-domain 2: Ensure data confidentiality 18
  - Sub-domain 3: Conduct an audit of operations 19
- Domain 5: Database Design 20
  - Sub-domain 1: Structuring databases 20
  - Sub-domain 2: Manage relationships between tables 21
  - Sub-domain 3: Adapt the modeling to the application's requirements 22

Introduction to the Tosa® Skills Framework

For Tosa® Assessment and Certification

Tosa® (Test on Software Applications)

Tosa® assessments and certifications are designed to determine a candidate's proficiency level by evaluating their skills in office software and digital tools commonly used in a professional or educational environment.

These tests are specifically developed to validate the professional competencies of candidates looking to enhance their employability (employees, students, job seekers, and individuals undergoing career transitions).

Tosa® assessments and certifications are adaptive tests, developed using scientific methodologies. The scoring is based on Item Response Theory (IRT). The test algorithm adapts to each candidate's response in real time, adjusting the difficulty level of subsequent questions until it precisely determines the candidate's skill level by calculating the upper limit of their competencies. As a result, the tests provide a detailed and unique diagnosis of each candidate's abilities.

The rigor and reliability of Tosa® tests stem from the combination of a mathematical model for analyzing question difficulty and the relevance of the questions selected for each candidate (IRT).

Tosa® Skills Framework Objective

This skills framework outlines all the skills assessed within the domains and sub-domains of the Tosa® assessment and certification tests for SQL.

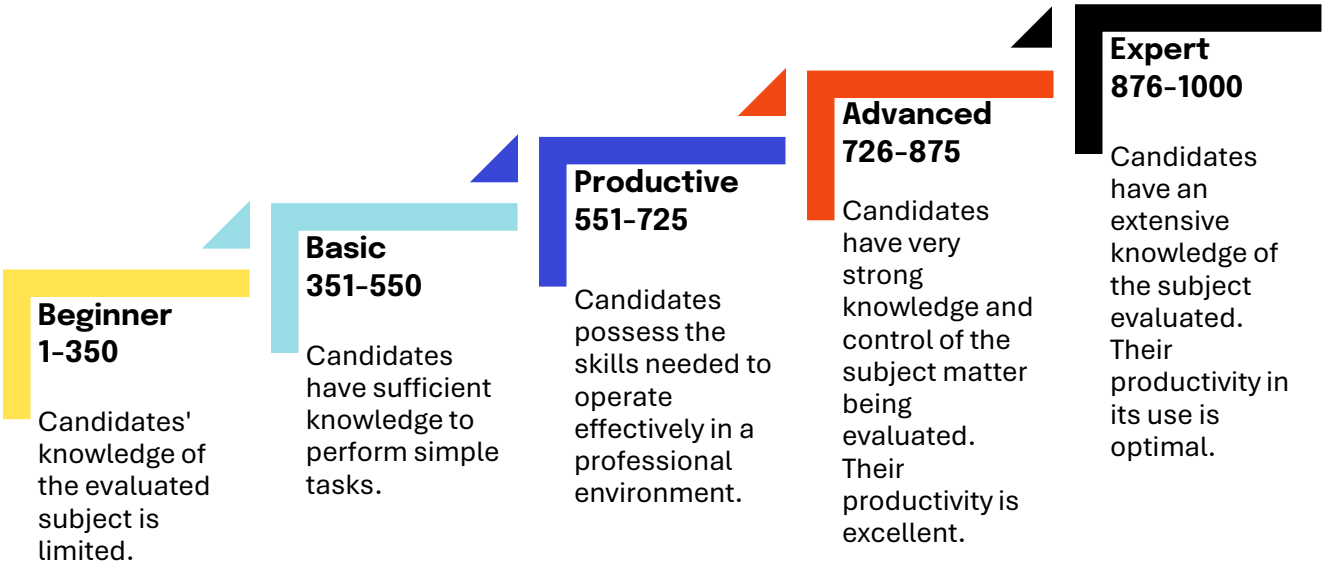
Tosa® assessment and certification solutions are designed to determine learners' proficiency levels using a single scoring scale—ranging from 0 to 1000 for certification—and divided into five levels, from "Beginner" to "Expert," for the assessment.

The purpose of this framework is to specify the technical knowledge expected at each level and within each of the four main skill categories of SQL. It is intended to help identify the most appropriate teaching or training programs to match a learner's target score.

Unique Tosa® Scoring

The Tosa® assessments and certifications are based on a unique score, divided into five levels.

- Ranging from 1 to 1000 for the certification.
- Divided into five levels, from Beginner to Expert, for the assessment.



Certification failed	Certification earned	Certification earned	Certification earned	Certification earned
Certificate of completion issued	Diploma issued	Diploma & Credly digital badge issued	Diploma & Credly digital badge issued	Diploma & Credly digital badge issued

About the Tosa® SQL certification

The Tosa SQL Certification relies on a database of around a 100 questions. It is composed of 35 questions from the question database and lasts for 1 hour and 30 minutes. The algorithm adapts to each of the candidate's answers to adjust the difficulty level of the questions until reaching the candidate's skill limit. This ensures a precise and accurate result.

Since the test is adaptive, the series of questions that a candidate gets is unique for each test. This algorithm allows for a more accurate evaluation of the candidate's level. It also limits cheating and the memorization of questions on different passages.

Our platform allows individuals to take the certification in a classroom setting, an approved testing center, or remotely via our integrated asynchronous online proctoring solutions.

Our remote proctoring solutions provide added flexibility for both the administrator and the candidate, allowing the certification exam to be taken anywhere, at any time. The candidate only needs an internet connection and a computer equipped with a working webcam and microphone.

Candidates receive a numeric score out of 1000 points. This score corresponds to one of five proficiency levels. Candidates who score between 1 and 350 points do not earn the certification. They receive a certificate of completion in lieu of a diploma. Candidates who score 351 points or above earn the certification and will receive a diploma by email within five business days. If a candidate scores 551 points or above, they will also be eligible to receive a digital Credly badge.

There are no prerequisites to be eligible to take the exam, but to ensure a candidate is well prepared on exam day, we suggest they:

- Take at least one Tosa SQL adaptive assessment to estimate their level and get familiar with the test format
- Use the free practice tests on our website for training
- Follow an e-learning or training course (average duration per level is between 10 and 15 hours per certification, so around 150 hours total)

Tosa certification diplomas are valid for three years from the date of issue. This three-year period has been set to ensure that our certifications are consistently accurate and relevant, taking into account software version updates as well as the natural evolution of a candidate's skills over time. Limiting the certification period also reflects the need for life-long learning and continual professional development.

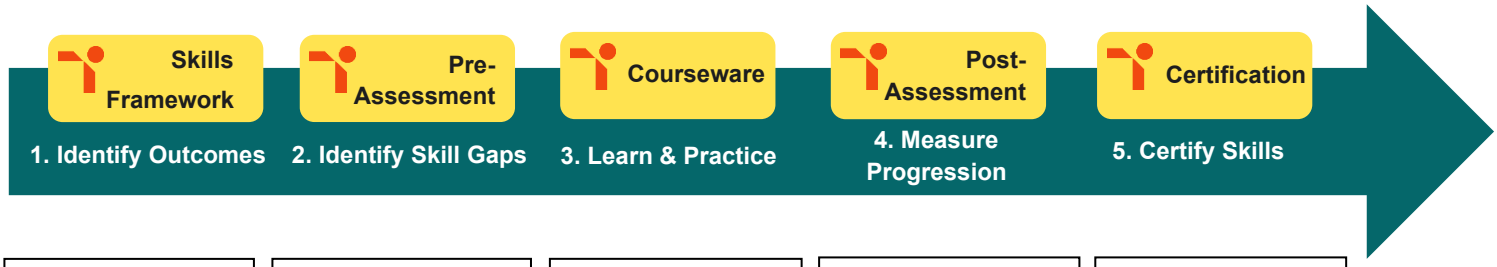
Tosa certifications can be retaken when they expire. Earners willing to improve their score and proficiency level can retake the exam at any time.

Isograd Learning Platform

The Isograd Learning Platform is a multifaceted and adaptable courseware solution designed to help learners prepare for Tosa certification exams. It offers personalized, self-paced, and fully interactive learning experiences, equipping candidates with the essential digital skills that employers seek. These skills span a wide and varied range, applicable to learners just starting their career journey to those wishing to advance on their path.

The browser-based platform supports all learning styles with a wide array of features, including in-application exercises. These in-application capabilities allow learners to experience real-world examples of specific tasks within a given software environment. The platform includes inclusive learning resources, along with extension activities and project-based learning challenges that foster creativity and critical thinking.

All course content on the Isograd Learning Platform is aligned to a Tosa skills framework, which is the foundation of any Tosa certification exam. By aligning course content to a skills framework, learners will be prepared to take the exam once they complete the courseware.



1. Identify Outcomes	2. Identify Skill Gaps	3. Learn & Practice	4. Measure Progression	5. Certify Skills
Using a Tosa® skills framework, instructors integrate Tosa® certifications into their existing content or curriculum and set define outcomes.	Students begin with a formative pre-course assessment to identify their current skills level. Instructors gather data and adjust lesson plans based on students' results.	Instructors match courseware in the Isograd Learning Platform to student's proficiency levels and intended course objectives.	Once the courseware is completed, students take a summative assessment to verify their skills match the intended proficiency level. If skill gaps still exist, instructors can assign additional course content.	When instructors are satisfied with students' progress, certification exams are given.

Tosa® SQL Level Descriptions

At each level, candidates can:

Beginner

- Understand what SQL is, what it stands for, and its role in managing data.
- Identify common use cases and tasks that can be performed with SQL.
- Recognize how SQL interacts with relational databases.
- Interpret and run simple queries to retrieve basic information (e.g., calculating a result from a dataset).

Basic

- Recognize SQL databases, core concepts, and common terminology.
- Write simple queries using conditions, sorting, and aliases.
- Use fundamental clauses such as WHERE, ORDER BY, GROUP BY, and LIMIT.
- Retrieve and combine data using joins and basic subqueries.
- Perform basic data manipulation operations (INSERT, UPDATE).
- Apply aggregation and filtering to analyze grouped data.

Productive

- Query relational databases to retrieve, filter, sort, and join data.
- Create and modify tables, define simple relationships, and manage data (insert, update, delete).
- Use basic transaction controls and indexing to maintain data integrity and performance.
- Manage data access and apply basic security measures.

Advanced

- Design and manage relational databases within application environments.
- Write complex queries using joins, subqueries, aggregations, and grouping to meet business needs.
- Model consistent schemas using keys, normalization, and referential integrity constraints.
- Optimize performance through indexing strategies and execution plan analysis.
- Manage transactions and implement appropriate access control and security mechanisms.

Expert

- Architect and optimize relational databases for complex, high-demand environments.
- Design advanced queries and adapt data models to evolving functional requirements while ensuring scalability and maintainability.
- Diagnose and optimize performance using execution plans, indexing strategies, and data structuring.
- Implement robust security (roles, encryption, auditing) for enterprise-grade environments.
- Troubleshoot and resolve complex issues related to data integrity, performance, and database design.

Business Applications

Achievement of Beginner score defines little or limited knowledge of SQL, including basic concepts such as what SQL is used for and how it interacts with databases, highlighting the candidate's inability to work with data in a professional environment.	*Administrative assistant, customer service representative, office support employee, or any position requiring the use of SQL to retrieve and organize data, perform basic queries, filter and sort information, or interact with structured datasets within existing systems.*	*Operations assistant, project coordinator, junior data analyst, IT support technician, or any role requiring regular use of SQL to query databases, combine data from multiple tables, perform aggregations, and support reporting or day-to-day data operations.*	*Data analyst, database specialist, financial controller, or any position requiring advanced use of SQL for designing and managing relational data structures, writing complex queries, optimizing performance, and supporting data-driven decision-making in a professional environment.*	*Senior data analyst, data engineer, database architect, business intelligence consultant, or any role requiring expert-level SQL proficiency to design, optimize, and maintain large-scale data systems, ensure data integrity and security, and support advanced analytics and strategic decision-making.*

Domain 1: Fundamentals and Applications

Sub-domain: Getting started with SQL

Covers foundational knowledge of SQL as a query language for interacting with relational databases, including its purpose, core use cases, and role in managing and retrieving structured data.

Beginner

- Identify tasks that can be performed with SQL
- Understand how to use SQL
- Know what SQL is used for

Basic

- Identify SQL database
- Understanding basic SQL terminology
- Match roles to their typical SQL usage

Productive

- Differentiate SQL from other programming languages
- Know SQL concepts and real-world cases
- Know the role of SQL in a company

Advanced

- Know database query language fundamentals

Expert

- N/A

Domain 2: Performance Optimization

Sub-domain 1: Create and manage indexes

Covers creating simple or composite indexes on key columns and analyzing their impact to improve query performance and reduce data access times.

Beginner

- N/A

Basic

- N/A

Productive

- Create simple indexes on columns used in WHERE, JOIN, or ORDER BY clauses to improve query performance.
- Check the DBMS's basic statistics to verify the effectiveness of the indexes.

Advanced

- Design indexes tailored to specific use cases (composite indexes, covering indexes) based on the columns used for filtering, joining, and sorting.
- Analyze the impact of indexes on performance, taking into account the cost of data modification operations.

Expert

- Optimize the indexing strategy for a relational database by tailoring indexes to the critical queries identified in the application.
- Adjust or remove redundant indexes to improve the balance between read performance and write overhead.

Sub-domain 2: Optimize queries and analyze execution plans

Covers optimizing queries using aggregate functions and refining complex query structures, while analyzing execution plans to identify and resolve performance bottlenecks.

Beginner

- N/A

Basic

- N/A

Productive

- Optimize simple queries by reducing the volume of data processed using appropriate filters.
- Interpret a basic execution plan to identify costly operations.

Advanced

- Rewrite complex queries by adjusting joins, subqueries, and aggregations to improve their efficiency.
- Analyze detailed execution plans to identify bottlenecks and apply targeted optimizations.

Expert

- Diagnose performance issues in complex queries by analyzing their execution plans and logical structures.
- Adjust query syntax and index usage to optimize response times in an application context.

Sub-domain 3: Structuring tables and schemas to improve performance

Covers adapting the table structure and use views or schema/modeling adjustments to facilitate data access and improve overall processing efficiency.

Beginner

- N/A

Basic

- N/A

Productive

- Create simple views to make it easier to access frequently used data.
- Structure tables consistently to minimize unnecessary redundancy and improve the readability of queries.

Advanced

- Implement modeling adjustments (e.g., partial denormalization, appropriate data type selection) to address performance needs.
- Optimize the use of views to simplify complex queries and improve maintainability.

Expert

- Adapt the structure of tables and views based on critical query patterns to improve overall performance.
- Make modeling decisions considering performance, readability, and maintainability of SQL code.

Domain 3: SQL Queries

Sub-domain 1: Write SQL queries

Covers writing SQL queries using conditions, logical operators, and aggregating functions to extract, filter, and organize data in response to a functional or application-specific need.

Beginner

- Write queries with filtering, aggregation, and calculated results

Basic

- Use aliases and simple expressions to improve query readability and structure results clearly.
- Write queries that extract and present structured information (e.g., listing products with attributes) in a clear and ordered format.

Productive

- Build SQL queries on a single table using basic filter conditions (WHERE) and logical operators (AND, OR) to extract data for basic analysis.
- Use aggregation functions (COUNT, AVG, SUM) to perform simple calculations on datasets.
- Format the results using simple string functions (CONCAT, UPPER) to improve the readability of the extracted information.

Advanced

- Build SQL queries that combine multiple tables using joins (INNER JOIN, LEFT JOIN) to analyze data from different sources and extract information useful for decision-making.
- Apply complex filters based on multiple conditions and advanced search expressions to refine query results.
- Use groupings (GROUP BY, HAVING) and subqueries in SELECT, WHERE, or FROM clauses to produce detailed and relevant results.

Expert

- Design complex SQL queries incorporating multiple joins, correlated subqueries, advanced conditional expressions, and analytical functions (WINDOW FUNCTIONS) to meet business requirements.
- Structure queries using CTEs and advanced coding techniques to improve the readability, maintainability, and reusability of SQL code.
- Optimize query logic by selecting appropriate query strategies without altering the overall database architecture.

Sub-domain 2: Manage and modify data

Covers manipulating data in a database using the SQL INSERT, UPDATE, and DELETE commands, and managing transactions to add, modify, or delete information while maintaining data integrity.

Beginner

- N/A

Basic

- N/A

Productive

- Use SQL data manipulation commands (INSERT, UPDATE, DELETE) to add, modify, or delete records in a table, while adhering to the integrity constraints defined in the database.
- Apply simple conditions using WHERE clauses to precisely target the relevant records and prevent any unintended modifications.

Advanced

- Handle INSERT, UPDATE, and DELETE operations by using subqueries and joins to manipulate data spread across multiple tables and ensure the consistency of updates.
- Use SQL transactions (BEGIN TRANSACTION, COMMIT, ROLLBACK) to group multiple modification operations and ensure data consistency and integrity in the event of an error or interruption.

Expert

- Design complex update operations by combining subqueries, joins, and conditional statements to ensure data consistency across multiple tables.
- Structure transactions to secure critical operations and ensure the validity of updates in complex application contexts.
- Optimize modification logic by selecting appropriate SQL statements (MERGE, conditional processing) to simplify code and improve maintainability.

Sub-domain 3: Perform joins and aggregations

Covers performing joins across multiple tables and using aggregate functions (e.g., SUM, AVG) to combine and analyze data from different sources.

Beginner

- N/A

Basic

- N/A

Productive

- Perform simple joins (INNER JOIN) to combine data from multiple tables using the keys defined in the database.
- Use aggregation functions (SUM, COUNT, AVG, MIN, MAX) in conjunction with GROUP BY to summarize data and produce structured results.
- Apply simple conditions (WHERE, HAVING) to filter the results of joins and aggregations according to specific requirements.

Advanced

- Perform complex joins (INNER, LEFT/RIGHT OUTER, CROSS) across multiple tables by structuring relationships in a way that is consistent with and tailored to the data model.
- Perform advanced aggregations by combining GROUP BY across multiple columns, HAVING filters, and conditional expressions to produce structured analysis.
- Improve the performance of queries combining joins and aggregations by utilizing simple or composite indexes tailored to the columns used in the join and group clauses.

Expert

- Design complex joins that incorporate multiple levels of relationships and correlated subqueries to structure sophisticated analysis.
- Perform advanced aggregations by combining multi-column GROUP BY clauses, conditional expressions, and subqueries to generate summaries tailored to complex business needs.
- Optimize the structure of queries combining joins and aggregations by analyzing their logic and adjusting index usage to improve their efficiency.

Domain 4: Data Security

Sub-domain 1: Manage access security

Covers creating users, assigning roles, and defining access rights to control database usage while applying the principle of least privilege.

Beginner

- N/A

Basic

- N/A

Productive

- Create users and assign basic access rights to databases and tables, applying the principle of least privilege.
- Configure simple authentication policies to secure access to databases.

Advanced

- Manage access rights at various levels (database, schema, table, column) using roles to structure and simplify permission management.
- Implement differentiated access policies (read, write, update) tailored to application requirements.

Expert

- Design a coherent structure of roles and permissions to secure a relational database used by multiple application profiles.
- Adjust and monitors access rights in response to functional changes to maintain a balance between security and ease of operation.

Sub-domain 2: Ensure data confidentiality

Covers implementing encryption and data protection mechanisms for sensitive data to secure stored or exchanged information.

Beginner

- N/A

Basic

- N/A

Productive

- Apply standard encryption mechanisms to protect sensitive data at rest or in transit.
- Implement authorized decryption procedures in accordance with established security policies.

Advanced

- Implement encryption solutions tailored to application requirements and configures the management of associated keys at the database level.
- Control access to sensitive data based on defined roles.

Expert

- Incorporate encryption and access control mechanisms into the database design to protect sensitive data while maintaining query performance.
- Tailor data protection strategies to identified application use cases.

Sub-domain 3: Conduct an audit of operations

Covers configuring logging mechanisms and analyze database operations to identify anomalies and ensure the traceability of operations.

Beginner

- N/A

Basic

- N/A

Productive

- Set up simple audits to track users' key actions (additions, modifications, deletions).
- Review audit reports to identify obvious anomalies.

Advanced

- Implement audit mechanisms for sensitive events and sets up alerts for suspicious activity.
- Analyze audit logs to identify trends or abnormal behavior.

Expert

- Develop an audit strategy tailored to the application’s operations to ensure the traceability of critical operations.
- Use audit results to adjust access rights and improve the database’s operational security.

Domain 5: Database Design

Sub-domain 1: Structuring databases

Covers creating and managing database structures by defining tables, column data types, and constraints to ensure data organization and integrity.

Beginner

- N/A

Basic

- N/A

Productive

- Create tables by defining the appropriate data types and applying basic constraints (PRIMARY KEY, NOT NULL) to structure the data consistently.
- Organize tables in accordance with fundamental data structuring principles, ensuring the uniqueness of entities and the clarity of the schema.

Advanced

- Design relational databases consisting of tables linked by primary and foreign keys, in accordance with normalization principles.
- Implement advanced integrity constraints (UNIQUE, CHECK, DEFAULT) and structures dependencies between tables to ensure the consistency and robustness of the schema.

Expert

- Design complex relational schemas by structuring dependencies between entities and optimizing table organization to facilitate application use.
- Anticipate future functional changes to the system by adapting the database structure to ensure its consistency and maintainability over time.

Sub-domain 2: Manage relationships between tables

Covers defining primary and foreign keys, establish relationships between tables, and applying referential integrity constraints to ensure data consistency and optimize query performance.

Beginner

- N/A

Basic

- N/A

Productive

- Define simple primary and foreign keys to establish relationships between tables and ensure basic referential integrity.
- Apply standard referential integrity constraints to maintain data consistency between related tables.

Advanced

- Design complex relationships between multiple tables and configures referential integrity rules tailored to functional dependencies.
- Optimize relationships between related tables to ensure data consistency while maintaining satisfactory performance during joins.

Expert

- Design and fine-tunes referential integrity mechanisms (cascades, update and delete rules) to ensure data consistency in complex relational structures.
- Make decisions regarding relationship modeling, taking into account the impact on schema readability, query performance, and the maintainability of SQL code.

Sub-domain 3: Adapt the modeling to the application's requirements

Covers designing a structured relational schema by applying normalization principles and anticipating application requirements to ensure the database's performance and maintainability.

Beginner

- N/A

Basic

- N/A

Productive

- Apply the fundamental principles of data modeling (identifying entities, relationships, and cardinalities) to design a simple and consistent schema.
- Use simple views to facilitate access to and interpretation of data in common scenarios.

Advanced

- Implement advanced design techniques, such as partial denormalization, to address identified performance constraints.
- Design dimensional models (star schema, snowflake schema) suitable for decision-making systems.

Expert

- Design advanced relational models by adapting the level of normalization to the application’s constraints and identified performance requirements.
- Structure the model to facilitate the application’s functional evolution while maintaining the consistency and clarity of the relational schema.

TOSA

Your skills. Your advantage.

contact@isograd.com
www.tosa.org