

Tosa Skills Framework

JavaScript

Overview

Introduction to Tosa Skills Framework	3
Tosa® (Test on Software Applications).....	4
Tosa Skills Framework Objective.....	4
Unique Tosa Scoring.....	4
Domains and subdomains	5
About the JavaScript Certification.....	5
Tosa Sequence for Progressive Skills Development	7
Level 1 – Beginner User	8
Overview	9
Level 2 – Basic User.....	10
Language and syntax.....	11
Data structures and objects	11
Candidates understand and use null and undefined types to materialize the absence of data. They can copy objects partially or modify them partially without having to recreate them.	11
Packages and API.....	11
Applied JavaScript	11
Overview	12
Level 3 – Productive User.....	13
Language and syntax.....	14
Data structures and objects	14
Packages and API.....	14
Applied JavaScript	14
Overview	15
Level 4 - Advanced User.....	16
Language and syntax.....	17
Data structures and objects	17
Packages and API.....	17
Applied JavaScript	17
Overview	18
Level 5 – Expert User	19
Language and syntax.....	20
The Expert-User candidates can perform a bit-by-bit operation with the appropriate operator. They know the precision of floating-point numbers, master a function context switching and they display expert competency in detecting errors during the execution of a program and resolving them.	20
Business application: For example, for a full-stack developer profile, these skills enable him performing numerical conversions and calculations to exploit the speed of bitwise operations	

compared to their traditional “Math” and “parseInt” equivalents and execute a function in accordance with the writing of a program (e.g., multiple uses of a "callback" function).	20
Data structures and objects	20
Packages and API.....	20
Expert-Users can create a program that detects a sub-expression after the execution of a regex. They can listen and react to custom events coming from the DOM (Document Object Model) and they have the mastery of the creation of asynchronous functions with different returned data in case of success and error.....	20
Applied JavaScript	20
Overview	21

Introduction to Tosa Skills Framework

For Tosa Assessment and Certification

Tosa® (Test on Software Applications)

Tosa assessments and certifications will determine and validate a candidate’s proficiency and skill level in software development used in a professional environment. Tosa assessments are designed to validate the professional JavaScript skills of individuals (students, trainees, employees, or jobseekers) in supporting their employment, professional, or academic objectives.

Tosa Assessments employ the Adaptive Testing methodology, which creates a personalized testing experience adapted to a candidate’s skill level for a selected software application. The score is based on the Item Response Theory using a 3-parameter logistic model, similar to the GMAT scoring method. Adaptive-based testing selects questions that challenge candidates to the limit of their knowledge and abilities.

Tosa Skills Framework Objective

This Tosa framework provides an overview of the subject areas being assessed during the Tosa Assessment and Certification exams. The Tosa JavaScript validates candidate proficiency in JavaScript programming using a score on a scale from 1-1000 for the Certification Assessment, and a score divided into 5 levels from “Beginner” to “Expert” for the Diagnostic Assessment.

The objective of this document is to present an overview of the technical skills associated with each of the 4 main JavaScript domains within each proficiency level. This information will also support educators and trainers in tailoring their training program to achieve desired proficiency levels.

Unique Tosa Scoring

The Tosa assessments and certifications are based on a unique score, divided into 5 levels.

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

Tosa® levels	Corresponding Tosa® score	Certification status & documents issued
Expert	876 - 1000	Certification earned - diploma & Credly digital badge issued
Advanced	726 – 875	Certification earned - diploma & Credly digital badge issued
Productive	551 – 725	Certification earned - diploma & Credly digital badge issued
Basic	351 – 550	Certification failed - certificate of completion issued
Beginner	1 – 350	Certification failed - certificate of completion issued

Domains and subdomains

Language and syntax	<ul style="list-style-type: none"> ■ Knowing the main characteristics and specificities of the language ■ Knowing basic functionalities and syntax ■ Managing the inputs, outputs and errors of a program
Data structures and objects	<ul style="list-style-type: none"> ■ Creating and manipulating primitive objects, embedded objects or more complex custom objects ■ Understanding and manipulating the properties of an object ■ Understanding and manipulating the prototype of objects
Packages and API	<ul style="list-style-type: none"> ■ Creating shareable and reusable features ■ Retrieving data from a server ■ Creating and manipulating asynchronous functions ■ Managing errors and successes during the execution of a program
Applied JavaScript	<ul style="list-style-type: none"> ■ Writing a program that handles different types of data in an optimized way ■ Understanding and finding errors within a code and correcting them ■ Searching for information in the documentation ■ Reading and understanding a stack trace to deduce the origin of a bug

About the JavaScript Certification

The Tosa JavaScript certification relies on a database of around 160 questions. It is composed of 20 questions and lasts one hour. The algorithm adapts to each answer of the candidates to adjust the difficulty level of the questions until they reach the exact definition of the candidates' level by calculating the limit of their skills.

Since the test is adaptive, the series of questions that each candidate gets is unique for each test. This uniqueness 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 class, in 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 associated to a proficiency level on a five-level scale. Candidates who score between 1 and 550 points don't earn the certification. They will not receive a diploma but a certificate of completion. Candidates who score 551 points or above earn the certification. They will receive a diploma by email within five (5) business days and are eligible to a Credly digital badge.

There is no requirement to be eligible to take the exam, but our recommendations to be well prepared on exam day are:

- Take at least one Tosa JavaScript adaptive assessment to estimate your level and get familiar with the test format
- Use free practice tests on our website for training
- Follow e-learning or training courses (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 as skill levels evolve or decline over time, depending on the use of the software. New software and software versions are released every year, and skills must be updated. We cannot legitimately certify a digital skills level for more than three years. Limiting the certification validity reinforces the need for life-long learning and professional development.

Tosa certifications can be retaken when it is expired. Earners willing to improve their score and level can also retake the exam at any time.

Tosa Sequence for Progressive Skills Development

Students and professionals can tailor their certification journey with Tosa through vertical progression, demonstrating increasing proficiency in JavaScript. Starting at a productive level, users can advance to advanced and expert levels as their skills grow. This clear path encourages continuous improvement and validates each stage of their development. Tosa's structure makes it easy to track progress and showcase evolving expertise.

Level 1 – Beginner User

Between 1 and 350 points

The Beginner Proficiency is set for a score from 1 to 350, which is the lowest Tosa score category. Achievement of Beginner score defines little or limited knowledge of JavaScript, including the language basic functions and features, highlighting the inability to use it in a professional environment.

Overview

Domains	Skills Assessed
Language and syntax	<ul style="list-style-type: none"> ❖ Creating and using a variable ❖ Commenting on a line of code ❖ Using basic arithmetic operators ❖ Displaying the content of a variable in the console
Data structures and objects	<ul style="list-style-type: none"> ❖ Recognizing and manipulating global objects ❖ Recognizing, creating, assigning, and reading contents of objects ❖ Recognizing, creating, assigning, and reading contents of arrays
Packages and API	<ul style="list-style-type: none"> ❖ Finding an element in the DOM
Applied JavaScript	

Level 2 – Basic User

Between 351 and 550 points

Prior to the acquisition of the skills of the Basic level, the candidate will have mastered the skills of the Beginner level.

Language and syntax

At this level, candidates can read and modify external documents, write comments in JavaScript to comment one's code and use flow control features to automate conditional and/or repetitive data processing. They are also able to use simple and reusable functions.

Business application: For example, for a JavaScript Developer profile, these skills enable automating the processing of data from external documents, recognizing typical JavaScript use cases and setting up data extraction processes from a known and stable source.

Data structures and objects

Candidates understand and use null and undefined types to materialize the absence of data. They can copy objects partially or modify them partially without having to recreate them.

Business application: For example, for a JavaScript Developer profile, these skills enable identifying the absence of data (verification of the filling of fields in a form) and creating programs which data can be modified: for example, enabling users to modify their personal data.

Packages and API

Candidates know the concepts related to modules in order to include the functionalities of a module in a program. They use functions from APIs such as Math or JSON and they can react to DOM events.

Business application: For example, for a JavaScript Developer profile, these skills enable creating a program reusing elementary functions and objects created by another team member or a program that performs advanced mathematical operations and produces structured output files.

Applied JavaScript

Candidates can produce code respecting given specifications and they can find the origin of a simple problem in a program and correcting it.

Business application: For example, for a JavaScript Developer profile, these skills enable implementing a technical specification required by a manager or correcting an existing problem in a given program.

Overview

Domains	Skills Assessed
Language and syntax	<ul style="list-style-type: none">Using const and let statementsCreating multi-line commentsUsing Boolean arithmetic operatorsCreating a workflow with conditional decisionsCreating simple loopsCreating your own functionReading user input from the console
Data structures and objects	<ul style="list-style-type: none">Recognizing undefined and null typesApplying a type conversion on a primitiveUsing basic functions on stringsCreating complex properties and assigning themUsing basic functions on arrays
Packages and API	<ul style="list-style-type: none">Importing a module from a fileUsing basic Math API functionsRecognizing and writing a JSON documentHandling basic DOM events
Applied JavaScript	<ul style="list-style-type: none">Iterating through an array and returning a modified versionDetecting a conditional error in a simple loop

Level 3 – Productive User

Between 551 and 725 points

Prior to the acquisition of the skills of the Productive level, the candidate will have mastered the skills of the Basic level.

Language and syntax

At this level, candidates can write technical documentation of code (JSDoc). They master the input-output tools, in order to read and modify external documents and arrow functions in order to favor a more concise syntax compared to traditional functions.

Business application: For example, for a JavaScript developer profile, these skills enable creating a library of functions along with technical documentation generated on the basis of JSDoc comments or creating more succinct code, and easily maintaining a readable code base.

Data structures and objects

At this level, candidates can create any type of string, regardless of its content and they can identify the type of a variable at runtime and processing it appropriately.

Business application: For example, for an analyst profile, these skills enable displaying any type of message without crashing the program and managing data in different forms and to processing them according to their specificities.

Packages and API

Candidates can create a simple module to manage shareable and reusable functionalities and they master JavaScript asynchronicity. They can make HTTP calls to remote servers.

Business application: For example, for a web developer profile, these skills enable grouping previously created functions and objects in a module for consistent reuse and creating a light web client connecting to a remote server to control and retrieve external data.

Applied JavaScript

Candidates can search online documentation for a program.

Business application: For example, for a web developer profile, these skills enable learning how to use basic technologies of an existing project.

Overview

Domains	Skills Assessed
Language and syntax	<ul style="list-style-type: none"> ✎ Using an assignment operator ✎ Creating standard JSDoc documentation ✎ Handling a strict equality comparison ✎ Configuring complex conditional decisions in a workflow ✎ Mastering flow control instructions associated with loops ✎ Creating arrow functions ✎ Managing properties with undefined values
Data structures and objects	<ul style="list-style-type: none"> ✎ Identifying types associated with variables at runtime ✎ Escaping any character within a string ✎ Creating a new object reference ✎ Performing complex operations on arrays
Packages and API	<ul style="list-style-type: none"> ✎ Creating a module and exporting functions ✎ Serializing and deserializing JSON ✎ Understanding asynchronous mechanisms ✎ Managing the success and failure cases of an asynchronous operation ✎ Adding an element in the DOM ✎ Retrieving data from a server
Applied JavaScript	<ul style="list-style-type: none"> ✎ Iterating through an array according to a given characteristic ✎ Finding a simple error in a given piece of code ✎ Finding the documentation associated with a simple JavaScript library

Level 4 - Advanced User

Between 726 and 875 points

Prior to the acquisition of the skills of the Advanced level, the candidate will have mastered the skills of the Productive level.

Language and syntax

Candidates can create a program for complete data processing, use the advanced features of flow controls and arithmetic operations. They master abstract and strict comparisons of primitive types.

Business application: For example, for a Webmaster profile, these skills enable aggregating available data from an existing extraction and writing comparisons with an adapted comparison type, as well as reducing bugs related to variable comparison.

Data structures and objects

Candidates master template literals, and they generate strings according to a predefined format in a simple and readable way. They are able to choose appropriate data structures.

Business application: For example, for a Webmaster profile, these skills enable displaying messages whose content differs according to the state of the program and enable improving the performance of a program.

Packages and API

Candidates know how to rename module imports and they master DOM manipulation (adding, removing, replacing elements).

Business application: For example, for a Webmaster profile, these skills enable using external modules with the same function names without conflicts and creating a dynamic web page whose content evolves according to external or internal events. (Ex: Creation of an instant messaging page).

Applied JavaScript

The Advanced-User candidates demonstrate the ability to read and understand a stack-trace in order to deduce the origin of a bug.

Business application: For example, for a Webmaster profile, these skills enable reproducing a bug and then analyzing the error message in order to deduce its origin and solving it.

Overview

Domains	Skills Assessed
Language and syntax	<ul style="list-style-type: none"> 🔧 Managing an abstract equality comparison 🔧 Mastering arithmetic operations for different types 🔧 Mastering the configuration of conditional decisions in a workflow
Data structures and objects	<ul style="list-style-type: none"> 🔧 Creating template literals 🔧 Manipulating the properties of an object 🔧 Creating an array from several other arrays 🔧 Identifying collections
Packages and API	<ul style="list-style-type: none"> 🔧 Renaming imported modules 🔧 Creating a regex that matches a string 🔧 Replacing DOM elements 🔧 Managing multiple asynchronous operations
Applied JavaScript	<ul style="list-style-type: none"> 🔧 Writing code capable of reading DOM data and processing it 🔧 Finding the origin of an error using a browser and knowing only the stack trace of the exception 🔧 Finding and understanding documentation to comprehend the standard and error output of an existing piece of code

Level 5 – Expert User

Between 876 and 1000 points

Prior to the acquisition of the skills of the Expert level, the candidate will have mastered the skills of the Advanced level.

Language and syntax

The Expert-User candidates can perform a bit-by-bit operation with the appropriate operator. They know the precision of floating-point numbers, master a function context switching and they display expert competency in detecting errors during the execution of a program and resolving them.

Business application: For example, for a full-stack developer profile, these skills enable him performing numerical conversions and calculations to exploit the speed of bitwise operations compared to their traditional “Math” and “parseInt” equivalents and execute a function in accordance with the writing of a program (e.g., multiple uses of a "callback" function).

Data structures and objects

The Expert-User candidates master the principles of prototyping and its use. They possess expert knowledge of various native array manipulation functions.

Business application: For example, for a full-stack developer profile, these skills enable transforming data stored in an array in an optimized way into a simple object ready to be displayed.

Packages and API

Expert-Users can create a program that detects a sub-expression after the execution of a regex. They can listen and react to custom events coming from the DOM (Document Object Model) and they have the mastery of the creation of asynchronous functions with different returned data in case of success and error.












Business application: For example, for a full-stack developer profile, these skills enable detecting and returning html tags contained in a string for a specific interpretation and retrieving data from a server to return the data received in case of success, or the error if an anomaly is detected during the processing of the request.

Applied JavaScript

The Expert-User candidates display expert competency in writing a program with advanced data manipulation logic. They understand errors within complex code and they can easily correct them.

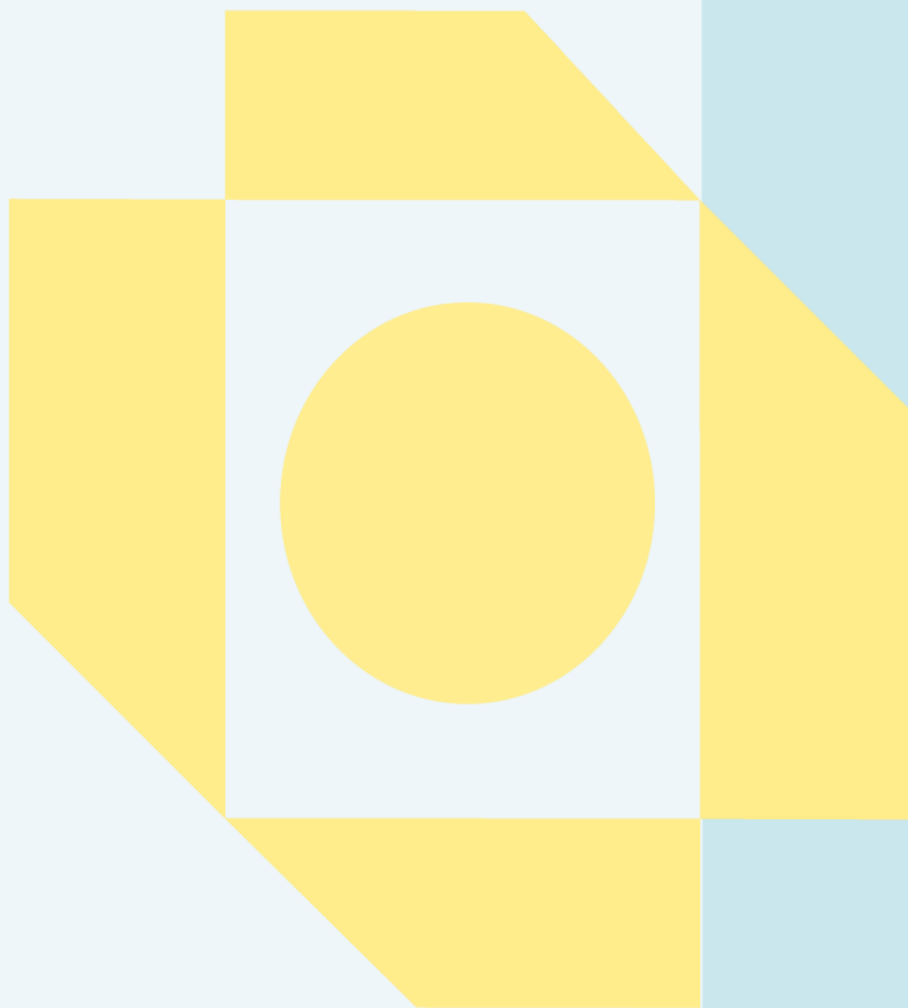
Business application: For example, for a full-stack developer profile, these skills enable sorting data in an optimized way and correcting an error and/or contributing to a code written by a team member.

Overview

Domains	Skills Assessed
Language and syntax	<ul style="list-style-type: none">  Applying bit-to-bit operations  Taking into account the imprecision of a floating-point number  Mastering function linkage  Catching and managing errors
Data structures and objects	<ul style="list-style-type: none">  Understanding the principle of prototype-oriented languages  Mastering array-operations
Packages and API	<ul style="list-style-type: none">  Capturing subexpressions in regex matches  Reacting to complex DOM events  Creating asynchronous functions
Applied JavaScript	<ul style="list-style-type: none">  Implementing complex logic within a function  Finding multiple errors in a complex piece of code



Your skills. Your advantage.



contact@isograd.com

www.tosa.org