Course Name : JSE – CERTIFIED ENTRY LEVEL JAVASCRIPT PROGRAMMER
Duration : 3 Days (Physical Classroom / Virtual Live Instructor)
Skill Level : Beginner

COURSE DESCRIPTION:
Embark on a transformative journey with our Official JavaScript Certification Training, leading to the prestigious JSE – Certified Entry-Level JavaScript Programmer certification. This comprehensive program immerses you in the intricacies of JavaScript, covering essential aspects such as syntax, variables, operators, control flow mechanisms, functions, and the intricacies of the data type system. Delve into universal programming concepts and grasp the role of a programmer in the software development process. Cultivate algorithmic thinking for efficient problem-solving and become adept at applying industry-best coding practices. Hone your troubleshooting and debugging skills, gaining proficiency in identifying and resolving common programming errors.

The JSE – Certified Entry-Level JavaScript Programmer certification is a professional credential that goes beyond theoretical knowledge. It demonstrates your practical understanding of the JavaScript language's core syntax and semantics. After completing this course, the candidate will have to write the official JSE-40-01 exam. The complete exam information is available https://js.institute/jse-certification here.

WHAT WILL YOU LEARN?
Acquire an in-depth understanding of JavaScript's core syntax and semantics, gaining proficiency in variables, operators, control flow mechanisms, and functions. Uncover the nuances of the JavaScript data type system, developing skills in exception handling, troubleshooting, and debugging. Cultivate algorithmic thinking to efficiently analyze problems, and apply industry-best coding practices for creating clean, efficient code in professional software development. Throughout the program, you'll gain hands-on experience using essential elements of the JavaScript language, tools, and resources to design, develop, and refactor simple JavaScript programs. Moreover, the certification ensures you are well-prepared to write the online exam, validating your expertise and commitment to excellence.

PREREQUISITE:
Beginners. No programming experience is required.

METHODOLOGY:
This program will be conducted with interactive lectures, PowerPoint presentations, discussions, and practical exercises. This course can be conducted as instructor-led (ILT) or virtual instructor-led training (VILT).

JOB SCOPE:
Upon completion of this course, candidates may pursue the following career paths:

- JavaScript Developer
- Front-End Developer
- UI/UX Developer
- Full-Stack Developer

EXAM BLOCK#1: INTRODUCTION TO JAVASCRIPT AND COMPUTER PROGRAMMING

- Welcome
- Understand the fundamental programming concepts, such as: interpreting and the interpreter, compilation and the compiler, and client-side vs. server-side programming;
- Have a basic knowledge of how to set up and use a basic programming environment (online or local)
- Gain skills allowing them to run their first JavaScript program on the client side (both as an element embedded in the HTML page and directly in the browser console).

EXAM BLOCK #2: EXAM BLOCK #2: VARIABLES, DATA TYPES, & TYPE CASTING

- Have the knowledge and skills to work with variables, i.e. naming, declaring, initializing, and modifying their values;
- Understand concepts such as scope, code blocks, shadowing, and hoisting;
- Know the basic properties of primitive data types such as boolean, number, bigint, undefined, and null, and be able to use them;
- Be familiar with the basic properties of the primitive data type string, including string literals – single or double quotes, escape character, string interpolation, basic properties, and methods;
- Know the basic properties of complex data types such as Array and Object (treated as a record) and be able to use them in practice.

EXAM BLOCK #3: OPERATORS AND USER INTERACTION

- know what operators are and how we classify them (by type of operands, by number of operands, etc.)
- be able to use assignment, arithmetic, logical, and comparison operators in practice;
- have an understanding of the operation of the conditional operator and the typeof, instanceof, and delete operators;
- understand what the precedence and associativity of basic operators are and be able to influence it by means of bracket grouping;
- be able to perform basic two-way communication with the program user using the alert, confirm, and prompt dialog boxes.

EXAM BLOCK #4: CONTROL FLOW - CONDITIONAL EXECUTION AND LOOPS

- Be able to force conditional execution of a group of statements (make decisions and branch the flow) using if-else and switch commands;
- Be able to force a group of statements to repeat in a loop using the for, while, and do-while commands, using both dependent and independent conditions on the number of iterations;
- Understand and be able to use loop-specific break and continue instructions;
- Be able to use the for-in statement to iterate over the properties of an object;
- Be able to use the for-of statement to walk through the elements of an array.

EXAM BLOCK #5: FUNCTIONS

- Be able to declare and call functions;
- Know how to pass call arguments to a function and return the result of its operation from it;
- understand the concept of a local variable and the effect of shadowing variables with the same names within a function;
- Know that a function in JS is a first-class member and be able to take advantage of this by declaring functions using function expression and passing functions as arguments to calls of other functions;
- Understand the concept of recursion in the context of functions and be able to solve simple programming problems by using it;
- Have a basic understanding of the callback function and be able to use it asynchronously in conjunction with the setTimeout and setInterval methods;
- Have a clear understanding of arrow function notation and be able to write functions alternatively as a regular declaration, a function expression, and an arrow function.

EXAM BLOCK #6: ERRORS, EXCEPTIONS, DEBUGGING, AND TROUBLESHOOTING

- understand the differences between syntactic, semantic, and logical errors;
- understand the concept of an exception and distinguish between the basic exceptions generated by JS when an error occurs: SyntaxError, ReferenceError, TypeError, RangeError;
- have the ability to handle exceptions using the try-catch-finally statement;
- be able to generate their own exceptions using the throw statement;
- have the skills to use the debugger for basic analysis of their own code, including: step-by-step execution, viewing and modifying variables, and measuring code execution time.

CONCLUSION

- QA
- Useful References and Books
- Feedback