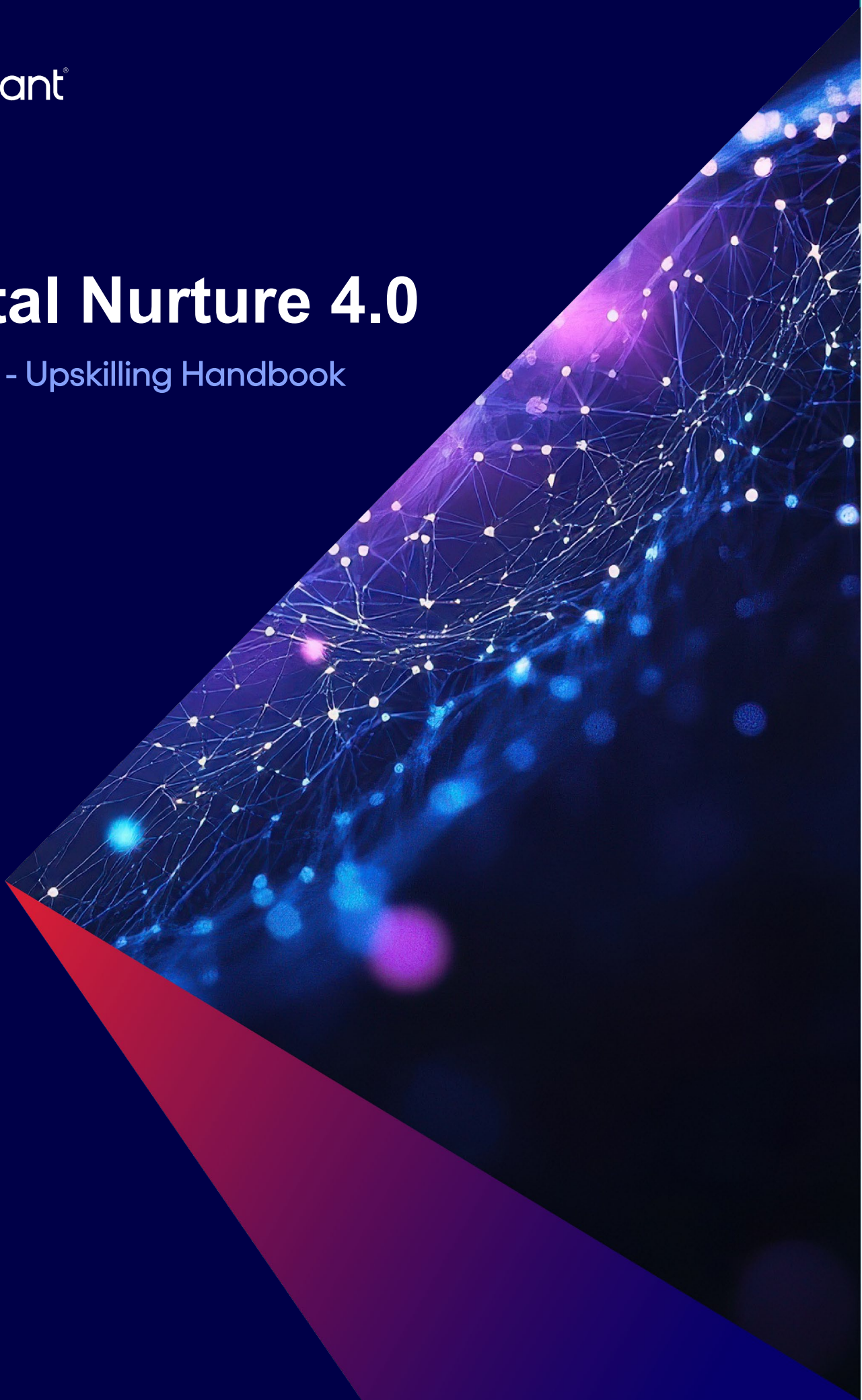
cognizant

Digital Nurture 4.0

Java FSE - Upskilling Handbook



Program Highlights

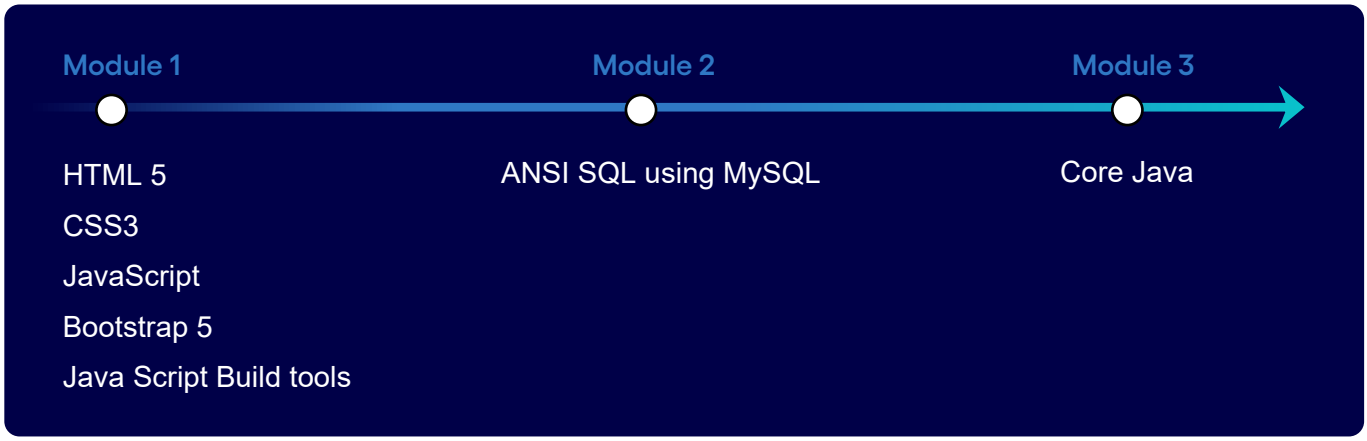
- The Upskilling learning program has 3 Modules to be completed. Go through the recommended self-learning resources and practice the exercises to excel in the recommended Java FSE modules.

- Upskilling covers the below mentioned skills.

  HTML 5 | CSS3 | JavaScript | Bootstrap 5 | JavaScript Build tools | ANSI SQL using MySQL | Core Java

Recommended Program Sequence

The learning journey contains the following modules, followed by an Upskilling Final Assessment.

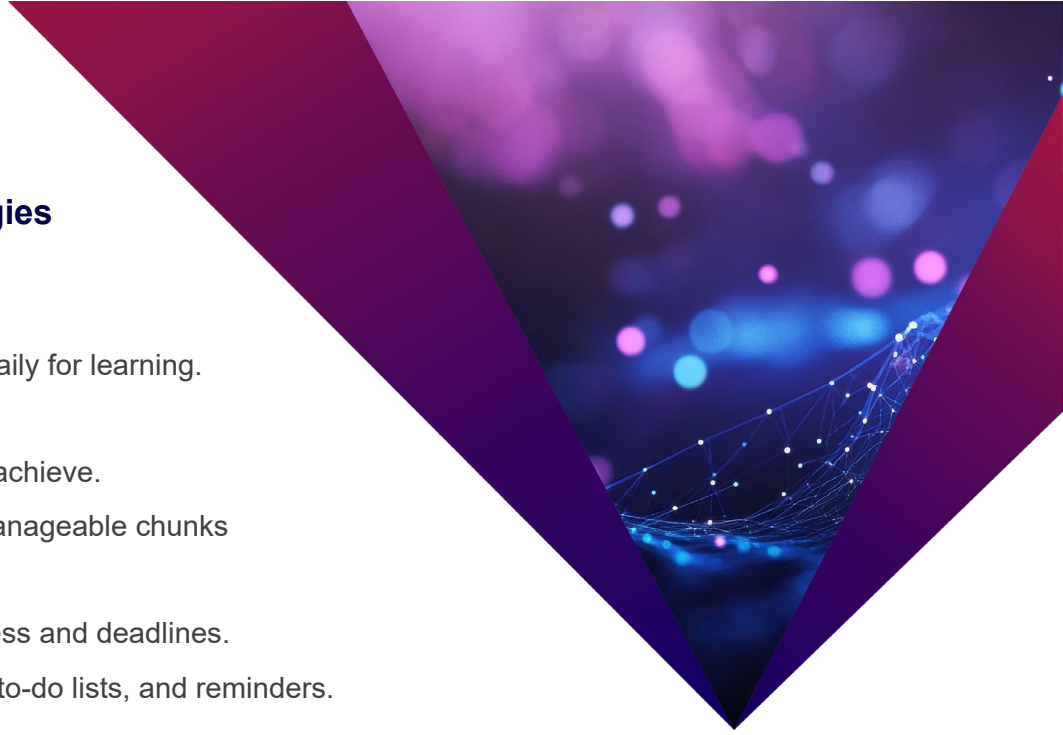


Learning Approach

Self-paced learning through open-source learning reference links

- Please refer to the learning reference links provided to learn and understand the recommended concepts.
- We expect you to dedicate 1-2 hours of focused attention daily and schedule longer sessions on weekends for your learning modules.
- Balance your learning with regular academic commitments.

Disclaimer: Cognizant does not claim ownership or responsibility for the content or any issues with the links provided, as they are merely references available on the internet. Candidates are free to leverage additional sources beyond what has been provided to enhance their skill capabilities for upskilling.



Effective Learning Strategies

1. Create a Study Schedule

- Dedicate specific times daily for learning.

2. Set Clear Goals

- Define what you want to achieve.
- Break down tasks into manageable chunks

3. Stay Organized

- Keep track of your progress and deadlines.
- Use tools like calendars, to-do lists, and reminders.

4. Active Learning

- Engage with the material through hands-on exercises.
- Practice coding and solving problems regularly.
- Ensure to complete all exercises.

Online Code Editor

Online Code Editors	URL
HTML5, CSS3, JavaScript	https://www.w3schools.com/tryit/tryit.asp?filename=tryhtml_hello
	https://onecompiler.com/html
	https://jsfiddle.net/
Bootstrap5	https://www.w3schools.com/bootstrap5/bootstrap_editor.php
SQL	https://www.mycompiler.io/new/sql
	https://www.jdoodle.com/execute-sql-online/
	https://www.tutorialspoint.com/execute_sql_online.php
	https://www.w3schools.com/sql/trysql.asp?filename=trysql_asc
	https://onecompiler.com/mysql
Core Java	https://www.tutorialspoint.com/compile_java_online.php
	https://www.onlinegdb.com/online_java_compiler
	https://www.jdoodle.com/online-java-compiler-ide/
	https://www.w3schools.com/java/java_compiler.asp
• Set up Integrated Development Environments (IDEs) like VS Code, IntelliJ IDEA, Spring Tool Suite, or Eclipse. • Install necessary tools and libraries as per course requirements.	

Exercise Instructions

In this learning program, you will be required to complete exercises designed to reinforce the concepts learned. These exercises are hosted on a public GitHub repository and must be downloaded and solved. Follow the instructions below to ensure a smooth and productive exercise workflow:

1. **Access the Exercises:**
    - Exercises will be made available in our public GitHub repository.
    - The repository URL is https://github.com/trinity2040/Digital-Nurture-4.0
    - Navigate to the repository and locate the folder named Java FSE -> Upskilling. Inside it, you will find a set of exercises.

2. **Download the Exercises:**
    - Download the files for the exercises by clicking the download button on the GitHub repository page.

3. **Solving the Exercises:**
    - Solve the problem statements provided in the downloaded files.
    - Ensure that you understand the problem requirements and apply the concepts learned.
    - Take your time to think through the solutions and code them accurately.

4. **Self-Evaluation:**
    - After completing the exercises, evaluate your solutions based on the problem criteria.
    - Compare your approach with any hints or solutions provided if available.
    - Reflect on any mistakes or areas where you can improve.

5. **Submitting Solutions:**
    - Firstly, organize your solutions and keep them in a folder.
    - Create a **public repository** in your personal GitHub account, upload your solution folder and share the URL with the POC **on demand**.

6. **Additional Support:**
    - If you encounter difficulties or have questions about the exercises, seek help from peers.
    - Utilize the resources and links provided in this handbook for further assistance.

Module 1

HTML5

Overview:

This module provides a comprehensive foundation in HTML5, the core markup language for web development. Learners will explore the essential concepts, elements, and best practices of HTML5 while gaining practical hands-on experience using modern development tools such as Visual Studio Code and Google Chrome Developer Tools. The module also introduces critical HTML5 APIs like Web Storage, Geolocation, and client-side database handling, enabling learners to create dynamic, interactive, and user-friendly web applications.

Through real-world examples, exercises, and exploration of browser features, participants will not only learn to structure web content effectively but also enhance the user experience with interactivity and persistence.

Learning Objectives:

By the end of this module, learners will be able to:

- Understand and Set Up the HTML5 Environment
- Apply HTML5 Tags for Content Structuring
- Build Navigation and References in Web Pages
- Handle Events and Interactions
- Explore Advanced HTML5 Form Features
- Implement Web Storage Solutions
- Understand Client-Side Databases
- Utilize HTML5 Geolocation API

Self-Learning (Open-source links):

HTML 5		
Topics	**Sub-Topics**	**Learning Reference Links**
Introduction	Need and Benefits of HTML, Setup, Browser, BOM and DOM, DOCTYPE, Character Encoding, <script> , <link>, HTML5 Document, Comments	https://www.w3schools.com/html/html_intro.asp
Getting Started	Visual Studio Code features, Google Chrome Developer tools, Inspect document	https://learn.microsoft.com/en-us/microsoft-edge/visual-studio-code/microsoft-edge-devtools-extension/open-devtools-and-embedded-browser

Elements & Attributes	Formatting Tags, List, Table, Form & Input Tags, Images, Styles, placeholder, inline and block elements, id vs class attributes	https://www.w3schools.com/tags/
		https://www.geeksforgeeks.org/html-forms/
		https://www.geeksforgeeks.org/difference-between-id-and-class-attributes-in-html/
		https://www.w3schools.com/html/html_blocks.asp
Navigation	Navigation tags, hyper link, reference to intermediate section	https://www.w3schools.com/tags/tag_nav.asp
		https://www.htmldog.com/guides/html/intermediate/sectioning/
Events	onblur, onchange, onclick, form related events, load events, key events, mouse events, ondblclick, onbeforeonload, oncanplay	https://www.w3schools.com/jsref/dom_obj_event.asp
Web Forms 2.0	<input> element in HTML5, <output>, placeholder attribute, autofocus, required	https://www.tutorialspoint.com/html5/html5_web_forms2.htm
Web Storage	Session storage, local storage and Delete web storage	https://www.w3schools.com/html/html5_webstorage.asp
Web SQL Database	openDatabase, transaction, transaction	https://www.geeksforgeeks.org/what-is-web-sql/
Geo location	Geolocation Methods, Location Properties, Handling Errors, Position Options	https://www.geeksforgeeks.org/html-geolocation/

Check Your Understanding:

HTML 5 Quiz ▶

CSS3

Overview:

This module introduces learners to the fundamental concepts of CSS3, a cornerstone technology for designing visually engaging, responsive, and user-friendly websites. Starting from understanding the need and benefits of CSS, learners will explore how to set up and apply CSS to HTML documents effectively. The module covers basic syntax, commenting, and the three primary ways of including CSS: inline, embedded, and external. Building on this foundation, the course dives into CSS3 selectors, enabling learners to target and style HTML elements efficiently. It then covers essential properties for managing color, backgrounds, fonts, text formatting, hyperlinks, lists, and tables.

The module also introduces the critical concept of the CSS Box Model, explaining margins, padding, borders, outlines, and the difference between visibility and display properties. Learners will further explore layout enhancements like multiple columns.

Finally, the module culminates with an introduction to Media Queries and Responsive Web Design (RWD) — crucial for creating adaptable websites that look great on devices of all sizes.

Learning Objectives:

By the end of this module, learners will be able to:

- Explain the need for CSS and identify its key benefits in web development.
- Set up CSS correctly and apply CSS using inline styles, embedded styles, and external stylesheets.
- Understand and apply CSS syntax rules and use comments effectively to enhance code readability.
- Utilize different types of CSS selectors (universal, element type, ID, class, grouping) to target HTML elements.
- Style web page elements using properties related to color, backgrounds, fonts, text, links, lists, and tables.
- Explain and apply the CSS Box Model, including margin, padding, border, and outline properties.
- Differentiate between the visibility and display properties and use them appropriately.
- Implement multiple column layouts using CSS3 properties.
- Create responsive designs using Media Queries to adapt websites for different screen sizes and devices.
- Demonstrate fundamental Responsive Web Design (RWD) principles to ensure website accessibility and usability across devices.

Self-Learning (Open-source links):

CSS3		
Topics	**Sub-Topics**	**Learning Reference Links**
Introduction	Need and Benefits of CSS, Setup, CSS Syntax, CSS Comments, Including CSS in HTML Documents(Inline styles, Embedded styles, External style sheets)	https://www.w3schools.com/html/html_css.asp
		https://www.w3schools.com/css/css_howto.asp
		https://www.geeksforgeeks.org/types-of-css-cascading-style-sheet/
Selectors	CSS3 Selectors - Universal Selector, Element Type Selector, Id Selectors, Class Selectors, Grouping Selectors	https://www.geeksforgeeks.org/css-selectors/
Styling		https://www.w3schools.com/css/css_colors.asp
		https://www.w3schools.com/css/css_background.asp

	CSS Color, CSS Background, CSS Fonts, CSS Text, CSS Links, CSS Lists, CSS Tables	https://www.w3schools.com/css/css_font.asp
		https://www.w3schools.com/css/css_link.asp
		https://www.w3schools.com/css/css_list.asp
		https://www.w3schools.com/css/css_table.asp
Box Model	CSS Box Model, Margin, padding, border, Outline, Visibility vs. Display, Multiple Columns	https://w3schools.com/css/css_boxmodel.asp
		https://www.geeksforgeeks.org/what-is-the-difference-between-visibilityhidden-and-displaynone/
Advanced	Media Queries, RWD	https://www.w3schools.com/css/css3_mediaqueries.asp
		https://www.w3schools.com/css/css_rwd_intro.asp

Check Your Understanding:

CSS3 Quiz ▶

JavaScript

Overview:

This module introduces learners to the core concepts and modern practices of JavaScript, one of the most essential languages for web development. Starting from setting up a development environment, learners will explore JavaScript syntax, data handling, and control structures. The module then progresses into functions, objects, and the Document Object Model (DOM) for dynamic web interactions.

Advanced topics such as asynchronous programming, modern ES6+ features, and working with forms and APIs are covered to equip learners with practical skills for real-world applications. The module concludes with an overview of debugging techniques, testing strategies, and an introduction to popular JavaScript frameworks and libraries, including using jQuery for DOM manipulation.

Learning Objectives:

By the end of this module, learners will be able to:

- Understand the role and setup of JavaScript in modern web development environments.
- Apply fundamental programming concepts using JavaScript syntax, data types, operators, and control structures.
- Develop modular and maintainable code through the use of functions, scope management, and closures.
- Create dynamic web applications by manipulating objects, handling events, and interacting with the DOM.
- Manage asynchronous operations effectively using callbacks, Promises, and async/await patterns.
- Leverage modern JavaScript features such as ES6+ syntax enhancements and modular programming techniques.

- Work with forms and external data sources by implementing form handling and API communication using Fetch API.
- Debug and test JavaScript applications to ensure code reliability and performance.
- Explore and utilize JavaScript libraries and frameworks to accelerate web development and improve productivity.

Self-Learning (Open-source links):

JavaScript		
Topics	**Sub-Topics**	**Learning Reference Links**
Introduction to JavaScript	What is JavaScript?, Setting Up JavaScript Development Environment	https://www.geeksforgeeks.org/introduction-to-javascript/?ref=lbp
JavaScript Basics	Syntax and Statements, Data Types, Operators	https://www.geeksforgeeks.org/javascript-basic-syntax/?ref=lbp
		https://www.geeksforgeeks.org/variables-datatypes-javascript/?ref=lbp
		https://www.geeksforgeeks.org/javascript-operators/?ref=lbp
Control Flow	Conditional Statements, Loops, Error Handling	https://www.geeksforgeeks.org/control-statements-in-javascript/
Functions and Scope	Function Basics, Scope and Closures, Higher-Order Functions	https://www.w3schools.com/js/js_functions.asp
		https://www.geeksforgeeks.org/difference-between-scope-and-closures-in-javascript/
		https://www.geeksforgeeks.org/javascript-higher-order-functions/
Objects and the Document Object Model (DOM)	Objects in JavaScript, JavaScript Prototypes, The DOM, Event Handling	https://www.geeksforgeeks.org/dom-document-object-model/
		https://www.w3schools.com/js/js_events.asp
Arrays and Array Methods	Array Basics, Array Methods	https://w3schools.com/js/js_arrays.asp
		https://www.w3schools.com/js/js_array_methods.asp

Asynchronous JavaScript	Understanding Asynchronous Programming, Callbacks and Callback Hell, Promises, Async/Await Syntax	https://developer.mozilla.org/en-US/docs/Learn_web_development/Extensions/Async_JS/Introducing
		https://geeksforgeeks.org/javascript-promise/
		https://www.geeksforgeeks.org/what-to-understand-callback-and-callback-hell-in-javascript/
JavaScript ES6+ Features	Let and Const, Template Literals, Destructuring, Rest and Spread Operators, Modules, Default Parameters	https://www.w3schools.com/js/js_es6.asp
JavaScript in Web Development	Working with Forms, AJAX and Fetch API	https://www.infoworld.com/article/2169473/using-javascript-and-forms.html
		https://www.w3schools.com/jsref/api_fetch.asp
		https://www.w3schools.com/js/js_ajax_intro.asp
Debugging and Testing JavaScript	Debugging Tools, Testing JavaScript Code	https://www.w3schools.com/js/js_debugging.asp
		https://learn.microsoft.com/en-us/visualstudio/javascript/unit-testing-javascript-with-visual-studio?view=vs-2022&tabs=jest
Introduction to JavaScript Frameworks and Libraries	Overview of Popular JavaScript Frameworks, Using jQuery for DOM Manipulation	https://www.geeksforgeeks.org/most-popular-javascript-frameworks-for-web-development/
		https://www.w3schools.com/jquery/jquery_intro.asp
		https://www.tutorialspoint.com/jquery/jquery-dom.htm

Check Your Understanding:

JavaScript Quiz ▶

Bootstrap 5

Overview:

This module provides a comprehensive introduction to Bootstrap 5, the world's most popular front-end framework for building responsive, mobile-first websites. You will start by understanding the core structure and setup of Bootstrap, followed by mastering the responsive grid system and flex utilities. The module will guide you through working with essential components like typography, forms, buttons, navbars, cards, and media objects. You'll also explore utility classes that make rapid UI development efficient, including spacing, colors, visibility, and positioning. Finally, the module covers using Bootstrap Icons, enhancing interactivity with JavaScript plugins, and customizing Bootstrap using Sass for project-specific styling needs.

Learning Objectives:

By the end of this module, learners will be able to:

- Set up a new project using Bootstrap 5 and understand its folder structure and core files.
- Use Bootstrap's Grid System to create flexible, responsive layouts.
- Apply column classes, alignment, and reordering techniques for different screen sizes.
- Utilize responsive flexbox utilities for efficient layout management.
- Style text effectively using Bootstrap Typography utilities.
- Build and customize forms, buttons, navbars, cards, and media objects.
- Manage spacing, colors, backgrounds, borders, shadows, and rounded corners using Bootstrap utility classes.
- Control element visibility, display properties, and positioning for responsive design.
- Integrate and use Bootstrap Icons in web applications.
- Utilize Bootstrap's JavaScript plugins to add dynamic behaviors such as modals, dropdowns, and carousels.
- Understand the basics of customizing Bootstrap using Sass variables and mixins to align with project-specific design requirements.

Self-Learning (Open-source links):

BootStrap 5		
Topics	**Sub-Topics**	**Learning Reference Links**
Introduction to Bootstrap 5	Overview of Bootstrap Framework, Setting Up Bootstrap 5, Bootstrap Structure and Files	https://www.w3schools.com/bootstrap5/bootstrap_get_started.php
Bootstrap Grid System	Fundamentals of Responsive Grid Layout, Column Layouts and Grid Classes, Alignment and Reordering in Grid, Responsive Flexbox Utilities	https://www.w3schools.com/bootstrap/bootstrap_grid_system.asp
Bootstrap Components	Typography, Forms, Buttons, Navbars and Navigation, Cards and Media Objects	https://www.geeksforgeeks.org/explain-the-components-of-bootstrap/

Bootstrap Utilities and Helpers	Spacing Utilities, Colors and Backgrounds, Display and Visibility, Borders, Shadows, and Rounded Corners, Positioning Utilities	https://www.w3schools.com/bootstrap5/bootstrap_utilities.php
Advanced Bootstrap 5 Features	Icons with Bootstrap Icons, Bootstrap 5 JavaScript Plugins, Customization with Sass	https://www.geeksforgeeks.org/how-to-add-icons-in-project-using-bootstrap/
		https://getbootstrap.com/docs/5.0/getting-started/javascript/
		https://www.freecodecamp.org/news/how-to-customize-bootstrap-with-sass/

Check Your Understanding:

Bootstrap 5 Quiz ▶

JavaScript Build Tools (Babel,Webpack)

Overview:

This module provides a comprehensive introduction to two critical tools for modern web development: Babel and Webpack.

You will begin by exploring Babel, a JavaScript compiler that enables you to write code using the latest JavaScript features while ensuring compatibility across different browsers and environments. You'll learn about key concepts like transpiling and polyfills, how to set up Babel with various configuration options, and how to enhance its functionality using plugins and presets. You'll also integrate Babel with build tools like Webpack and optimize the performance of your builds.

The second part of the module dives into Webpack, a powerful module bundler. You'll understand its role in managing dependencies and optimizing assets for deployment. Core concepts like entry and output points, loaders, and plugins will be covered in depth. You'll learn how to configure Webpack for different project needs, integrate it with Babel and TypeScript, and optimize it for better performance using techniques like caching, minification, and bundle analysis.

Learning Objectives:

By the end of this module, learners will be able to:

- Explain what Babel is and why it is essential for modern JavaScript development.
- Describe the core concepts of Babel, including transpiling and polyfills.
- Set up and configure Babel using .babelrc and babel.config.js files.
- Utilize plugins and presets (e.g., @babel/preset-env) to customize Babel's behavior.
- Integrate Babel with Webpack to create a seamless build process.

- Apply performance optimization techniques in Babel such as caching and minification.
- Define what Webpack is and describe its purpose in the web development workflow.
- Understand Webpack's core concepts, including entry and output points.
- Install and configure Webpack using webpack.config.js.
- Manage modules and dependencies effectively using Webpack.
- Explain the role of loaders in Webpack and utilize common loaders like babel-loader and css-loader.
- Use plugins such as HtmlWebpackPlugin and CleanWebpackPlugin to extend Webpack's functionality.
- Optimize Webpack builds through caching, minification, and bundle analysis.
- Integrate Webpack with other tools, specifically Babel and TypeScript, for advanced project setups.

Self-Learning (Open-source links):

JavaScript Build Tools (Babel, Webpack)		
Topics	**Sub-Topics**	**Learning Reference Links**
Babel	Introduction to Babel: What is Babel?, Why use Babel?; **Core Concepts:** Transpiling, Polyfills; **Setup and Configuration:** Installing Babel, Configuring .babelrc and babel.config.js**; Plugins and Presets:** Using Plugins, Commonly Used Presets (e.g., @babel/preset-env); **Integration with Build Tools:** Using Babel with Webpack**; Performance Optimization:** Caching, Minification;	https://www.geeksforgeeks.org/what-is-babel/ https://babeljs.io/docs/configuration https://www.robinwieruch.de/webpack-babel-setup-tutorial/
Webpack	Introduction to Webpack: What is Webpack?, Why use Webpack?; **Core Concepts:** Entry and Output; **Setup and Configuration:** Installing Webpack, Configuring webpack.config.js, Modules and Dependencies; **Loaders:** What are Loaders?, Common Loaders (e.g., babel-loader, css-loader); **Plugins:**What are Plugins?, Common Plugins (e.g., HtmlWebpackPlugin, CleanWebpackPlugin); **Performance Optimization:** Caching, Minification, Bundle Analysis; **Integration with Other Tools:** Using Webpack with Babel, Webpack with TypeScript;	https://www.freecodecamp.org/news/an-intro-to-webpack-what-it-is-and-how-to-use-it-8304ecdc3c60/ https://webpack.js.org/guides/typescript/

Check Your Understanding:

Quiz on Babel with Webpack ▶

Module 2

ANSI SQL using MySQL

Overview:

This module provides a comprehensive introduction to ANSI SQL standards with practical application using MySQL. It covers the importance of SQL standardization, basic to advanced querying techniques, data manipulation operations, and schema management. Learners will start by understanding the foundation of SQL and gradually move into querying data with SELECT statements, applying filters, sorting results, using aggregate functions, and performing table joins. The module further explores working with subqueries, manipulating data within tables, and managing database schema objects. It also focuses on ensuring performance and integrity through indexes and constraints. Upon completion, learners will have a strong foundational knowledge to work with relational databases using SQL confidently and efficiently.

Learning Objectives:

By the end of this module, learners will be able to:

- Understand the role and significance of ANSI SQL and MySQL in relational database management.
- Retrieve and manipulate data from relational databases using standard SQL queries.
- Apply filtering, sorting, and aggregation techniques to manage query outputs effectively.
- Implement advanced querying techniques using joins and subqueries to combine and analyze complex data sets.
- Perform data manipulation operations including inserting, updating, and deleting records.
- Design and manage database structures by creating, altering, and dropping tables.
- Enhance database performance and enforce data integrity using indexes and constraints.

Self-Learning (Open-source links):

ANSI SQL for MySQL		
Topics	**Sub-Topics**	**Learning Reference Links**
Introduction to ANSI SQL and MySQL	Overview of ANSI SQL, Introduction to MySQL, Importance of Standard SQL	https://www.dwbi.org/pages/30/introduction-to-ansi-sql
		https://www.w3schools.com/mysql/mysql_intro.asp
Data Retrieval with SELECT Statement	Basic SELECT syntax, Retrieving specific columns, Filtering data with WHERE clause, Sorting results with ORDER BY	https://www.geeksforgeeks.org/mysql-select-statement/

Filtering and Sorting Data	Using logical operators in WHERE clause, Combining conditions with AND, OR, Sorting data using multiple columns	https://www.sqlshack.com/learn-mysql-sorting-and-filtering-data-in-a-table/
Aggregate Functions and Grouping	Introduction to aggregate functions (SUM, AVG, COUNT, etc.), GROUP BY clause for grouping results, HAVING clause for filtering grouped data	https://www.geeksforgeeks.org/mysql-aggregate-function/ https://www.w3schools.com/mysql/mysql_groupby.asp
Joins and Subqueries	INNER JOIN, LEFT JOIN, RIGHT JOIN, Self-joins and cross joins, Subqueries in SELECT, WHERE, and FROM clauses, Correlated Subqueries	https://www.geeksforgeeks.org/mysql-join-1/ https://www.geeksforgeeks.org/mysql-subquery/
Data Modification with INSERT, UPDATE, DELETE	Inserting data into tables, Updating existing records, Deleting records from tables	https://www.geeksforgeeks.org/sql-ddl-dql-dml-dcl-tcl-commands/
Creating and Modifying Tables	Creating tables with CREATE TABLE, Modifying table structure with ALTER TABLE, Deleting tables with DROP TABLE	https://www.geeksforgeeks.org/sql-create-table/?ref=lbp https://www.geeksforgeeks.org/sql-alter-add-drop-modify/?ref=lbp
Indexes and Constraints	Creating indexes for better query performance, Defining primary and foreign key constraints, Ensuring data integrity with UNIQUE and CHECK constraints	https://www.w3schools.com/mysql/mysql_constraints.asp https://www.w3schools.com/mysql/mysql_create_index.asp

Check Your Understanding:

MySQL Quiz ▶

Module 3

Core Java

Overview:

This module provides a comprehensive journey through the essential topics in Core Java programming. Beginning with the setup of the Java development environment and basics of writing programs, it progresses through key programming concepts like data types, operators, control flow, and methods. It further builds expertise in object-oriented principles, exception handling, collections, functional programming with lambdas and streams, file handling, multithreading, and debugging techniques.

Learners are introduced to advanced concepts such as modular programming, networking, reactive programming, reverse engineering, new Java 17 and 21 features, and database interaction using JDBC. By the end of the module, learners will have a solid foundation for building scalable, efficient, and maintainable Java applications.

Learning Objectives:

By the end of this module, learners will be able to:

- Set up and work with the Java development environment (JDK, JRE, JVM) and build basic Java programs.
- Understand and apply fundamental Java programming constructs including variables, data types, operators, control flow, loops, and methods.
- Develop structured, efficient, and well-documented Java applications following best coding practices.
- Implement object-oriented programming principles such as encapsulation, inheritance, polymorphism, abstraction, and interfaces.
- Handle exceptions and manage error scenarios effectively using Java's exception handling mechanisms.
- Work with collections, streams, and functional programming features (lambdas, method references, optional handling) for data processing and transformation.
- Perform file operations and manage input/output streams for data persistence and retrieval.
- Develop multithreaded and concurrent applications using Java's concurrency utilities.
- Debug Java applications using professional tools like IntelliJ IDEA to analyze and improve code quality and performance.
- Apply reactive programming principles and develop non-blocking applications using Project Reactor.
- Leverage modular programming techniques to structure and manage larger Java projects.
- Implement basic networking concepts and build Java applications that communicate over TCP, UDP, and HTTP.
- Understand reverse engineering concepts including bytecode analysis, reflection, and code security techniques.
- Utilize new features from Java 17 and Java 21 to write modern and optimized Java applications.
- Connect to databases using JDBC, execute SQL queries, and manage transactions within Java programs.

Self-Learning (Open-source links):

Core Java		
Topics	**Sub-Topics**	**Learning Reference Links**
Introduction to Java	Overview of Java Platform and Editions, Installation and Setup of Java Development Environment (JDK 21), Introduction to Java Development Kit (JDK), Java Runtime Environment (JRE), and JVM, Writing, Compiling, and Running Java Programs	https://www.geeksforgeeks.org/introduction-to-java/
Basic Syntax	Structure of a Java Program, Writing and Running a Simple Java Program, Comments in Java	https://www.geeksforgeeks.org/java-basic-syntax/?ref=lbp
Data Types and Variables	Primitive Data Types, Reference Data Types, Variable Declaration and Initialization, Type Casting	https://www.geeksforgeeks.org/data-types-in-java/?ref=lbp https://www.geeksforgeeks.org/variables-in-java/?ref=lbp
Operators and Expressions	Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Operator Precedence	https://www.geeksforgeeks.org/operators-in-java/?ref=lbp https://www.programiz.com/java-programming/expressions-statements-blocks
Control Flow Statements	Conditional Statements (if, else if, else, switch), Looping Statements (for, while, do-while), Break and Continue Statements	https://www.geeksforgeeks.org/decision-making-javaif-else-switch-break-continue-jump/ https://www.geeksforgeeks.org/loops-in-java/?ref=lbp
Methods	Defining and Calling Methods, Method Parameters and Return Types, Recursion	https://www.geeksforgeeks.org/methods-in-java/
Arrays and Strings	One-Dimensional and Multi-Dimensional Arrays, String Class and String Methods, StringBuilder and StringBuffer	https://www.w3schools.com/java/java_arrays.asp https://www.w3schools.com/java/java_strings.asp
Object-Oriented Programming in Java	Classes and Objects, Encapsulation, Inheritance, Polymorphism, Abstraction and Interfaces	https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/

Exception Handling	Introduction to Exceptions, Handling Exceptions, Custom Exceptions	https://www.geeksforgeeks.org/exceptions-in-java/
Java Collections Framework	Introduction to Collections, List Interface and Implementations, Set Interface and Implementations, Map Interface and Implementations, Queue Interface and Implementations, Stream API	https://www.geeksforgeeks.org/collections-in-java-2/
Functional Programming in Java	Lambda Expressions, Functional Interfaces, Method References, Optional Class, Streams and Parallel Streams	https://www.geeksforgeeks.org/functional-programming-in-java-with-examples/
Java I/O and File Handling	Java I/O Streams, File Handling with java.nio.file Package, Serialization and Deserialization	https://www.geeksforgeeks.org/file-handling-in-java/
Multithreading and Concurrency	Introduction to Multithreading, Thread Lifecycle and Thread Control, Synchronization, Concurrency Utilities (java.util.concurrent package)	https://www.geeksforgeeks.org/multithreading-in-java/ https://www.geeksforgeeks.org/java-util-concurrent-package/
Application Debugging Using IntelliJ IDEA	What is Debugging?, Importance of Debugging, Overview of Debugging Tools, Setting Up IntelliJ IDEA, Debugging Environment Overview, Setting Breakpoints, Running the Debugger, Inspecting Variables and Expressions, Using Watches, Exception Breakpoints, Thread Debugging, Memory and Performance Profiling	https://www.jetbrains.com/help/idea/debugging-your-first-java-application.html
Introduction to Reactive Programming	Reactive Programming Fundamentals, Reactive Streams and Backpressure, Project Reactor Basics, Concurrency in Reactive Programming	https://www.baeldung.com/cs/reactive-programming
Working with Java Modules	Introduction to Modular Programming, Creating and Using Modules, Modularity in Java 17	https://www.geeksforgeeks.org/java-modules/
Java Networking	Basics of Networking in Java, TCP and UDP Communication, HTTP Client	https://www.geeksforgeeks.org/java-networking/

Reverse Engineering Concepts	Introduction to Reverse Engineering, Decompilation in Java, Java Bytecode Analysis, Reflection and Introspection in Java, Debugging and Code Analysis Techniques, Code Obfuscation and Deobfuscation, Working with Legacy Code and APIs, Software Security through Reverse Engineering	https://www.geeksforgeeks.org/software-engineering-reverse-engineering/
Language-Specific Features - v17, v21	**v17** - Sealed Classes and Interfaces, Pattern Matching for instanceof, Text Blocks, Records, Switch Expression Enhancements, Hidden Classes**; v21 -** String Templates, Sequenced Collections, Pattern Matching for switch and Record Patterns, Virtual Threads	https://mydeveloperplanet.com/2023/11/01/whats-new-between-java-17-and-java-21/
Introduction to Java Database Connectivity (JDBC)	JDBC Overview, Executing SQL Queries, Handling Transactions in JDBC	https://www.geeksforgeeks.org/introduction-to-jdbc/

Check Your Understanding:

Java Quiz ▶

What Next

Congratulations on successfully completing the DN 4.0 Upskilling Learning Program!

As you have now finished this important phase of your learning, you will be taking a **Skill Based Assessment (SBA)** to certify your skills. **This assessment will cover all the skills and topics you have learned during Upskilling Learning Program**, ensuring you have a comprehensive understanding of the material.

We wish you the best of luck for your assessment and look forward to seeing you apply your newly acquired knowledge and skills. Good luck!