Working with ApplinX Frameworks

●	Web Application Configuration	Configuring Your Web Application
●	General Application Customization	Customizing the Default Template
		Creating a New Template
		Activating an Application Map from a Menu
		Using ApplinX Repository Folders' Structure to Organize Web Pages
		Controlling the Connection Properties from the Code
		Handling Flickering of Screens
		Waiting for Screens, using Wait Conditions
		Customizing the Host Keys
		Activating the Screen Locker
		Natural UNIX: Integrating a Login Page in the Web Application
		Implementing & Controlling JavaScript Events using the gx_event Object
		Retrieving Data from Fields Outside the Modal Window Currently Displayed
		ApplinX Server Load Balancing
		Retrieving a Unique Device Name per User Name / IP Address from a Database and Applying it to the Current User Session
		Retrieving the Host Printer Device Name from a Database and Setting the ApplinX Printlet to Work with that Device
		Customize ApplinX Framework Session Error Handling
		Customize the Web Application's Error Page
●	Transferring Natural Data to/from the Host	Transferring Data using ActiveX
		Transferring Data without using ActiveX
		Activating a Natural Command
●	Instant Pages Customization	Using a Proportional (Non-Fixed) Font in Web Pages
		Controlling Instant Display Properties
		Code Transformations
		Creating a New Code Transformation
		Applying a New Transformation to all Screens

Working with ApplinX Frameworks

		Applying a New Transformation to a Screen Group
		Manipulating Individual Host Fields
		Positioning Specific Fields
		Formatting Specific Fields
		Replacing a Field’s Text
		Replacing a Field with a Web Element, Adding a Web Element
		Manipulating Host Characters
		Manipulating Host Keys
		Improving Transitions between Screens
•	Emulation Behavior Tasks	Customizing the Background Check for Host Screen Changes
		Enabling the User to Control the Font Size
		Opening Multiple Web Sessions
		Printing a Capture of the Host Screen
		Enabling Sending Dup and FieldMark Characters to the Host
•	Page Customization	Generating a Framework Page for a Screen
		Creating Designed Web Pages
		Using Web Application Controls in Generated Pages
		Partial Page Rendering
		Creating a Button / Hyperlink for Submitting a Host Key
		Creating a Button / Hyperlink for Executing a Navigation Path
		Collect all Modified Page Fields into an ApplinX Request
		Exporting Data to an MS Office Application (Excel, Word)
		Building an External Login Page
		Collecting Data from Multiple Host Screens
		Binding Procedure Outputs to an ApplinX Framework Based Web Page
		Updating Data in Multiple Host Screens
		Activating a Server Side Function from JavaScript
		Mapping Keyboard Keys to User Actions in Individual Pages
		Handling the Screen Locker on the Page Level
		Navigating between Input Fields
		Retrieving Browser Information
		Validating your Data

Working with ApplinX Frameworks

		Handling Web Application Windows using the gx_windows Object
		Working with Cookies
		Working with JavaScript User Exits
		Retrieving HTML Objects using gx_getElement
		Using the Calendar Component in Generated Pages
		Replacing Static Host Confirmation Messages with JavaScript Confirmation Pop-up Box
		Opening an Independent Pop-up Box that doesn't have a Corresponding Host Screen
•	Working with Tables	Creating a Page with a Table
		Adding the Sorting Capability to a Screen-Based Table
		Adding the Sorting Capability to a Procedure-Based Table
		Changing Table Layout for Instant HTML Pages
		Retrieving Values from a Selected Row within a Table
		Customizing the Table's Display
•	Transferring Files (FTP)	FTP Configuration
		Using FTP to Upload Files
		Using FTP to Download Files
•	Printlet Servlet Redirector for ApplinX	
•	Framework Management	Upgrading an Existing JSP Web Application
		Deploying an ApplinX Web Application (JSP)
		Upgrading an Existing .NET Web Application
		Deploying an ApplinX Web Application (.NET)
		Disconnecting the Host Session Correctly
•	Troubleshooting your Framework	Performance Monitoring
		JavaScript Logger Engine
		Investigating the Web Application's Code