

COURSE OVERVIEW:

This CVPD Java Training – Using Java to Extend Call Studio and VXML Server Functionality (CVPDJT) is a hands-on programming course teaching participants to extend the functionality of CVP Call Studio v11.6 using the CVP Java API. Participants will learn to write, compile, deploy and maintain custom components for CVP Call Studio and VXML Server in the CVP V11.6 environment. Participants must be familiar with Call Studio and know a small amount of Java or similar programming language.

PREREQUISITES:

The knowledge and skills that a learner must have before attending this course include:

- Working knowledge of CVP Studio
- Basic experience programming with Java or a similar programming language

COURSE OBJECTIVES:

Upon completing this course, the learner will be able to meet these overall objectives:

- Introduction to the Java API, its capabilities and limitations
- Practice creating every different Java component to extend Studio (see below)
- Introduction to Voice Elements and their complexity
- This course is applicable for any version of CVP and Studio.

COURSE OUTLINE:

- Standard Actions and Decision as 'drop in code' to execute functions on VXML Server such as obtaining the IP address for setting the Default Media Server, for accessing substrings, validation of dates and times, performing date arithmetic.
- Configurable Action and Decision elements - creating new elements in Studio. Substring element, String Length element, Read File of name=value pairs, Read an XML file, Date comparison element test if a date is in the past, future, or today.
- Say it Smart plugins - Say it Smarts convert data into a list of audio files and/or text for TTS playback. For example, create a plugin to spell data to the caller. Example, speaking a date in European format.



Using Java to Extend Call Studio & VXML Server Functionality (CVPDJT) 1.0

- Start of Call classes - Create session variables or VoiceXML Properties based on ICM passed data, or database or file. Example, access a database to create root document VXML Properties.
- End of Call classes - execute code at the end of visit to an application, regardless of how the visit ends (hangup, complete, error, application transfer, telephony transfer). Example, determine if call ended with an error, write to a custom log file.
- Start of Application classes - execute application specific code whenever VXML Server is started or the application is updated. This is often used to create customer detail records. Example, create persistent variables.
- Create Persistent Variables - Global variables and Application variables remain in memory between phone calls but can only be accessed through Java code.
- Dynamic Menus - configure Settings and Audio for Studio elements at runtime using Java. This is especially useful for building Dynamic Menus with varying number of options and audio prompts. Example, create a Dynamic Menu based upon caller's account number.
- Extend existing Studio elements to add/modify settings, catch Java exceptions, encapsulate multiple elements