# Real Time College

Course: Configuration Management - Ansible

Duration: 30 Hours

Hands-On-Training: 50%

Real Time Group is a multi-disciplinary dynamic and innovative Real-Time O.S. and Embedded Software Solutions Center, established in 2007.

Providing Bare-Metal and Embedded Linux solutions, professional services and consulting, end-to-end flexible system infrastructure, outsourcing, integration and training services for Hardware, Software and RT-OS \ Embedded Systems.

The company is divided into the following three Divisions:

Real Time Group		
College	**Development**	**Out-sourcing**
rt-ed.co.il	rt-dev.com	rt-hr.co.il
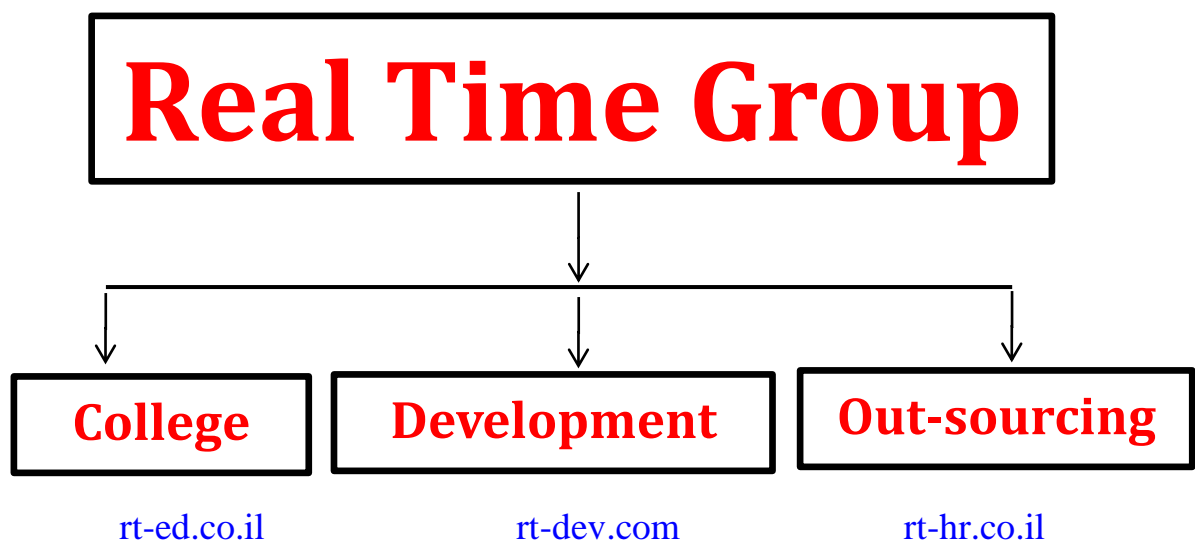
Training Division:

Professional Training Services for Hardware, Software, RT-OS and Embedded systems industries.

We provide the knowledge and experience needed to enable professional engineers to Develop, Integrate and QA Hardware, Software and Networking Projects.

In order to insure experience, all courses are practical – hands-on-training.
The latest Development, QA and Automation equipment which are adopted by the industry are used.

All students are supplied with Development-Boards for home-work and course projects.

Course Overview:

Configuration management is a process for maintaining the configuration of systems throughout their life cycle. A system under configuration management control can have deployed artifacts, configuration files, system packages, user configurations, and services, all defined in source control. This ensures a repeatable process, which is the foundation of automation.

Ansible is an example of a configuration management solution. Mastering Ansible is a step-by-step journey of learning Ansible for configuration management and orchestration.

The course is designed as a journey through configuring a realistic application stack from the ground up. Instead of going page-by-page through the Ansible documentation, topics are ordered to align with the growing complexity of our application as we build it up and refactor it.

The goal is to have a workflow where all of the configuration and troubleshooting is done through ansible playbooks that can be committed to a repository and improved over time.

Who should attend:

- This Course is intended for testers, programmers, DevOps or IT professionals who would like to learn and to understand what Ansible is, how it works, how to deploy a software, service or web application using Ansible and why DevOps use Ansible.

Prerequisite:

- Understanding of Software Development Life Cycle.
- Knowledge about Linux is mandatory.
- Familiarity with Source control utilities.
- Basic Testing concepts are an advantage.

Ansible Course Outline

1. **Introduction to Configuration Management**
   a. What is Configuration Management?
   b. What is Cloud Configuration Management?
   c. What are Configuration Management Tools?
   d. What are the best Configuration Management Tools?

2. **Introduction to Ansible**
   a. What is Ansible?
   b. How Ansible works?
   c. What are Ansible Modules
   d. Main Terminology for Ansible

3. **Installing Ansible - Preparing Your Environment**
   a. Checking your machine requirements
   b. Installing Ansible
   c. Checking installed Ansible version
   d. Configuring Ansible on your Environment

4. **Introduction to YAML files**
   a. What is YAML file?
   b. YAML Syntax
   c. Lists and dictionaries presented in YAML files
   d. YAML files examples
   e. Exercise to write a YAML file

5. **Getting Started with Ansible**
   a. Running first Ansible command - ping
   b. The concept of the inventory file and how Ansible uses it
   c. The default Ansible inventory file format and location
   d. Configuring a local inventory file with hostnames and groups
   e. Configuring a local config file to use the new inventory file by default
   f. Configuring remote hosts by updating SSH public keys
   g. Pinging all remote hosts using Ansible

6. **Building your own inventory file**
   a. Different inventory file formats
   b. INI format for inventory files
   c. YAML format for inventory files
   d. Groups in inventory files
   e. Default Groups
   f. Multiple Groups
   g. Nested Groups
   h. Ranges of Hosts
   i. Variables Assigning
   j. Inventory Aliases
   k. Inventory Examples

7. **Ansible Ad-hoc Commands**
   a. What is ad-hoc commands?
   b. Why to use ad-hoc commands?
   c. How to list files and folders remotely?
   d. To connect as a different user on remote hosts
   e. To reboot remote hosts

8. **Ansible Tasks**
   a. Learn about the basic building block of Ansible: the task
   b. Understand the structure of tasks: modules and arguments
   c. Group tasks for a common set of hosts together into plays

9. **Ansible Playbooks**
   a. What is Ansible Playbook?
   b. Packages: apt
   c. Install packages with the apt module
   d. Set package installed state using one of: present, latest, absent
   e. Ensure the apt cache is updated before package installation
   f. Packages: become
   g. Packages: with_items
   h. Manage running service state with the service module
   i. Services: apache2_module, handlers, notify
   j. Files: copy

10. **Ansible Variables**
    a. What are variables in Ansible?
    b. Why to use variables in Ansible?
    c. Using variables in playbooks
    d. Installing a list of packages using variables

11. **Ansible Roles**
    a. What is Ansible Role?
    b. Why to use roles in Ansible?
    c. Introduction to ansible-galaxy
    d. Creating a role for Apache installation
    e. Creating a role for MySql installation
    f. Creating a role for PHP installation
    g. How to deploy a website using ansible roles?