Real Time College

Course: **Docker**

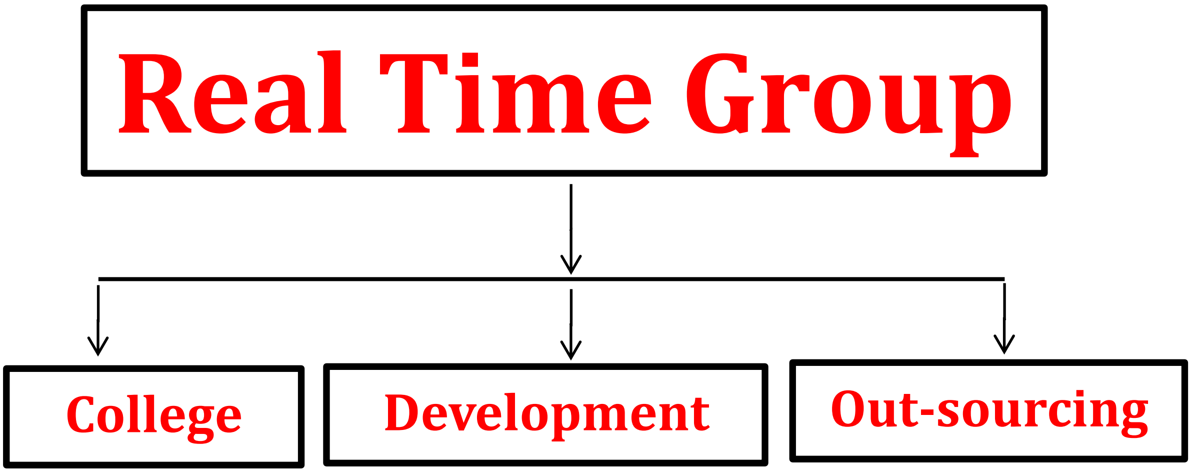
Duration: 30 Hours

Hands-On-Training: 75%

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

Providing Linux solutions, professional services, IT 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 IT, 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:

Docker is a platform that turns out to be a perfect fit for the DevOps ecosystem. It's developed for software companies that are struggling to pace up with the changing technology, business, and customer requirements.

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications.

In this course you will Learn the essentials of Docker. After completing this course, you will be able to build applications in a cloud native way using containerization tool.

Who should attend:

- This Course is intended for testers, programmers or IT professionals who would like to learn how to create applications using Docker containers.

Prerequisite:

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

Docker Course Outline

1. **Introduction to Docker**
   a. What is Docker used for?
   b. What a Docker container is?
   c. Virtualization Architecture
   d. Docker Architecture
   e. How containers differ from virtual machines?
   f. The Benefits of using containers
   g. How Docker utilizes a layered filesystem to create images?
   h. System Requirements

2. **Instating Docker - Preparing Your Environment**
   a. Download Docker
   b. Installing Docker
   c. Configuring the Docker on your Environment
   d. Setting Up Docker

3. **Getting Started with Docker**
   a. Start and stop Docker containers
   b. Display information about containers running on the host system
   c. Download container images
   d. Use the built-in Docker help system

4. **Managing Container Images**
   a. Download a Docker image
   b. View the history of an image
   c. Tag an image
   d. Delete an image
   e. View and clean-up the storage used by images

5. **Building images with Dockerfiles**
   a. Docker Registries
   b. Building Images with Dockerfiles
   c. Exercise Instructions: Build and Push an Image
   d. Exercise Walkthrough: Build and Push an Image

6. **Building a Web Server with Docker**
    a. Managing Ports with Docker
    b. Building a Web Server as a Docker container
    c. Pushing the Web Server Docker image to Docker Registries
    d. Building a Web Site as a Docker container

7. **CI/CD with Docker Integration**
    a. CI/CD overview
    b. Using pipelines for Continuous Delivery
    c. Deployment different docker containerized apps
    d. CI/CD secured
    e. Deployment a Web application with Docker

8. **Advanced Docker usage (Based on Timing Constraints)**
    a. Introduction to Docker Volumes
    b. Managing Docker Volumes
    c. Docker Networking and Dockerizing Applications
    d. Instruction Commands
    e. Docker-Compose
    f. Introduction to YAML files