

Session Details

Session Build Dynamic Software

Objective Learn best practices on how to build software such that changes are additions and not modifications.

The objective is to learn code construction such that maintenance and feature addition is done by adding new code, even to running application, such that the application can reconfigure itself and reflect the new behavior.

Trainer Sanjay Vyas

Dates November 9-10, 2019

Time 9am-6pm

Facilities

Venue **Left Right Mind**

Ground Floor, Business Bay, Plot No. 84,
Wellesley Road, Near RTO (Sangam Bridge) Pune 411001



<https://goo.gl/maps/JSzUb9CXUzs>

Internet Connectivity **Internet connectivity will be provided at the venue**
Please carry a dongle/hot-spot as a backup

Writing Material **Please carry the following**
A4 size stationery (1-2 40 leaves book/pad)
Flair Sunny 4-in-1 color pen (black, blue, red, green)

Stationery can also be bought for Rs. 50 at venue

Tea/Lunch Tea & simple vegetarian lunch will be served during the session

Parking Parking space is available at nearby walking distance free public parking space.
See Google Map link below. Please arrive early to find a good spot. <https://goo.gl/maps/k285FWwAN4m>

POI Hotels [Click here for Hotels near Center](#)

AirBnb [Click here for AirBnb's nearby](#)

Restaurants [Click here for Restaurants in the area](#)

ATMs [Click here for ATMs in the area](#)

Trainer Profile

Trainer Sanjay Vyas

Bio Sanjay has been in the field of IT since 1984. He has worked with companies like Datamatics, Informatics and Aptech. He co-founded Synergetics in 1993, thru which he trained and architected solutions world over on varied technologies like Unix, C, C++, Windows, COM, .NET, Linux and Java stack.



He has been Microsoft MVP several times, nominated as Microsoft Regional Director in 2016, and was invited by Microsoft to their Head Quarters in Redmond to participate in Software Architect Forum.

Currently he is focusing on JS stack, Python, DS, ML, AI and cloud.

Methodology The session will be conducted using Concept Visualization methodology, where concepts will be drawn as 3D visuals to understand how code works internally.

Contact Telegram: @sanjay_vyas
Email: sanjay.vyas+d2d@gmail.com

Prerequisites

Entry Profile **Basic development experience in programming.**

Participants must be experienced developer in any platform/language
Proficiency in basic programming concepts is a must

- Program structure
- Data Type
- Control flow statement
- Basic OOP concepts

Exit Profile Participants will be able to use concepts learned effectively in their projects.

Hardware Get any laptop with **minimum** 4Gb memory

Software OS Windows 10, Apple Mac or Linux (Ubuntu preferred)

IDE **[C# or C++]**
[Microsoft Visual Studio Community 2017](#)

[Java]
[Eclipse IDE](#)
[Java JDK](#)

Software RDBMS – SQL Server OR MySQL

Day 1

Fundamentals of dynamic languages	Static vs Dynamic languages	30 mins
	Dynamic typing vs Dynamic code	
	Internals of language process maps	
Why software changes	Business rule changes	30 mins
	New features	
	Bug fixes	
Code construct which forces changes	Customer changes his mind too often	
	switch-case	1 hr
	if-else-if-else ladder	
Kill the switch and if-else ladder	sequential conditions which change over time	
	static creation of objects using new	
	Eliminate	4 hrs
Case Study	- value replacement fixed switch	
	- value replacement dynamic switch	
	- value-action switch	
	- value-action if-else ladder	
	- varied condition sequential ifs	
	- multiple switch across modules	
	- dynamic creation of objects	
	Build a dynamic software where future changes are accommodated without modifying existing code	2 hrs

Day 2

Parameters	Mutating set of parameters causing changes	1 hr
	Constructing ParamObj to reduce signature changes	
	Designing polymorphic ParamObj	
Eliminate new	Dynamic ParamObj using maps	
	Hardcoded object creation causes change	2 hr
	Switch to dynamic creation using reflection	
Case Study 2	Use Registry driven dynamic object creation	
	Build Object Factories which are not static	
	Use Dependency Injection	
Case Study 3	Convert a brittle existing application into dynamic software	4 hr
	Ensure changes are not modifications	
	Configure it such that new or existing modules can be changed by dropping a new dll/jar in running apps directory	
	Analyzing an existing dynamic software	1 hr
	Understanding the architectural decisions applied	
	Looking for design patterns applied	

Terms & Conditions

Payment	Payment must be made before training begins.
Refund	No refund for cancel/no-show. Seat may be transferred.
Cancellation	Full refund if cancellation due to issues with venue/facility/trainer.
Other Notes	Check for travel time & parking at venue before training begins.