**Core JAVA**

**fundamentals**

**Oops concepts**

**Collection**

**Exception**

**File handling**

**Thread handling**

**Automation:**

**Selenium –** WebDriver component

Locators, Event handling

testNG Unit testing Framework

Maven

Jenkins:

Git & Github

POM

BDD Framework – Cucumber / Junit

M/C language:- Language which understands by the computer , in the form Binary – 0,1

Assembly Language: (Nonstructural) Low level Language ( understand by the ma/c in the form instructions)

STORE A,10

STORE B,10

ADD A,B

Assembler: which converts assembly lang into m/c Lang.

**Procedural lang.:C (structured ) – compiled lang.**

**HLL--------------------- Compiler ---------------------M/c (0,1)**

**Compiler: compiles the whole code in one go ,and gives / displays the Syntax error**

**HLL-** which I plain English lang. easily understands by Human being.

Procedure: functions

There is no data Security

C++ : oops based lang.

Oops based: classes and Object concept

**Four Pillars/ concepts:**

* Inheritance
* Encapsulation
* Polymorphism
* Abstraction

**Interpreter:** - interpreter interprets the code and find the errors line by line .

**Programming lang:-** compiled based lang.

**Scripting Lang:-**interpreter based lang

**OOPs:**

Class: blueprint, collection of objects:

For eg: map of house, Student

Object: real world entity, tangible, touchable, an object having some properties and its behavior

House building, jagroop

Student- class

**Properties** (variables): Jagroop(object)- name, id, height, weight

**Behavior** – walk, eat (in the form of methods)

Cacluator:class

Object - simple cal.

+,-,keys,=

Properties/ variable:- a=10, b=20,color

Methods: sum(){

C = A+b

}, sub

**Four Pillars/ concepts:**

* Inheritance: - Parent- child relation
* Encapsulation: capsule 💊 binding up the data as a single entity/object
* Polymorphism: Many form, one method will behave differently

Pen/laptop: - main button

Wash the vegs

* Abstraction: hiding the inner implementation

Eg: CPU

**JAVA: programming lang, oops based lang,both compiled and interpreted lang.**

**This is also a platform.**