**TUTORIAL – 1**

1. Compare and contrast declarative and imperative paradigms.

* Paradigm can also termed as method to solve some problem.
* There are two types of paradigms.

Imperitive Programming Paradigms (How to do)

1. Procedural Programming Paradigm.
2. Object Oriented Programming.
3. Parallel Processing Approach.

Declarative Programming Paradigms (What to do)

1. Logic Programming Paradigm.
2. Functional Programming.
3. Database processing Approach.

* Both are divide into three sections.
* Main difference between both programming paradigms are declarative programming paradigms declare the results we want rather how it has be produced.

Ex - # Declarative

small\_nums = [x for x in range(20) if x < 5]

# Imperative

small\_nums = []

for i in range(20):

if i < 5:

small\_nums.append(i)

1. Discuss the difference between procedural programming and functional programming.

* Functional Language allows you to write mathematical function.

Ex – JavaScript, Perl, Haskwell, Scala, Erlang, Lisp, ML, Clojure

* Procedural language performs a series of sequential steps. It has ability to reuse the code.

Ex - C, C++, Java, ColdFusion, Pascal

1. Explain the Lambda calculus and Lambda expressions in functional programming.

* Lambda calculus is a framework developed by Alonzo Church in 1930s to study computations with functions.
* Lambda calculus has three different type of expressions.
* E :: = x (Variables)
* E1.E2  (Function Application) ( E1 – Function Application, E2 – Actual Argument)
* λx.E (Function creation) (x – formal argument, E – Functional body)
* λx.E is Lambda Abstraction.

1. What is meant by “no side-effects” and “referential transparency” in functional programming?

* In programming, a side effect means, when a procedure changes a variable from outside its scope. They are out of language dependency. There are some classes of languages which aim to eliminate side effects (pure functional languages). a side effect is related to state changes. So, Functional programing languages, in that sense, they actually eliminate side effects, since they save no state.
* Referential transparency is an expression, in a program, may be replaced by its value (or anything having the same value) without changing the result of the program.

1. Discuss the key features of Object Oriented Programming.

* OOP – Object Oriented Programming.
* Object Oriented Programming is a programming language model organized around objects and data.
* Objects – Objects are instance of a classes which we can use store data and actions.
* Classes - A class is the abstract definition of the data type, it includes attributes and methods which are defined on the datatype.
* Abstraction - Ability of developing new data types,which contains attributes and methods on those attributes.
* Encapsulation – This concept is used to hide attributes and methods from outside.