# OBJECT ORIENTED PROGRAMMING

# INTRODUCTION

# PARADIGM VS PROGRAMMING VS IDE

**PARADIGM**

- Imperative
- Logic
- Functional
- Object Oriented

**LANGUAGE**

- C
- C++
- C#
- JAVA
- SWIFT

**IDE**

- NetBeans
- Visual Studio
- Android Studio

# PARADIGM

- A programming paradigm is a fundamental **style of computer programming**.
  - Paradigms differ in concepts and abstractions used to represent the elements of program.
  - Different methodologies are more suitable for solving certain kinds of problems or applications domains.

# PARADIGM

- Programming paradigms differ in:
  - The concepts and abstractions used to represent the elements of a program (such as objects, functions, variables, constraints, etc.)
  - The steps that compose a computation (assignation, evaluation, data flow, control flow, etc.).

# PARADIGM

- Types of paradigms:
  - Declarative
    - Logic paradigm → Prolog language
    - Functional paradigm → Lisp language
  - Imperative
    - Procedural paradigm → Fortran, Pascal, C languages
    - Structured paradigm → Pascal, Algol, Ada languages
    - Non-structured paradigm → Assembly, Joss, Focal languages
    - Object Oriented paradigm → Java, C++, C# languages

# PARADIGM

- Declarative:
  - Expresses the logic of a computation, without describing its control Flow.
  - They’re concerned with WHAT you want done, rather than HOW you want it done.
- Imperative:
  - Use statements that change a program’s state
  - Expresses through the use of commands for the computer to perform.
  - Focuses on describing HOW a program operates.

# FUNCTIONAL PARADIGM

**LISP:**

```
> (defun sum-greater-than (x y z) (> (+ x y) z))
> (print (sum-greater-than 1 4 3))
> T
```

# LOGIC PARADIGM

**PROLOG:**

```
cat(tom).
mouse(jerry).
eats(tom,jerry).

1 ?- mouse(tom).
      false.
2 ?- cat(tom).
      true.
```

# LOGIC PARADIGM

**PROLOG:**

```
cat(tom).
mouse(jerry).
eats(tom,jerry).

1 ?- mouse(tom).
     false.
2 ?- eats(jerry,tom).
     false.
```

# IMPERATIVE PARADIGM

**C:**

```
#include <stdio.h>
int main(){
    int iVal;
    printf("Hello World, give me a value ");
    scanf ("%d",&iVal);
    printf("Value %d" , iVal);
    return 0;
}
```

# OBJECT ORIENTED PARADIGM

**JAVA:**

```
public class Main{
	public static void main(String[] args) {
		System.out.println("Hello World");
	}
}
```

# PROGRAMMING LANGUAGE

- Coded language used by programmers to write instructions that a computer can understand, to do what the programmer (or the computer user) wants.
- Is a formal language which comprises a set of instructions used to produce various kinds of output.
- Are used to create programs that implement specific algorithms.

# LANGUAGE

- **A system of** conventional spoken, manual, or **written symbols** by means of which human beings, as members of a social group and participants in its culture, express themselves.
- The functions of language include **communication**, the expression of identity, play, imaginative expression, and emotional release.

# IDE

- **Integrated Development Enviroment**
- It’s a tool
- Is a software application that provides comprehensive facilities to computer programmers for software development.
  - Netbeans
  - Visual Studio
  - Android Studio
  - Eclipse