A STUDY ON EVOLUTION OF GENERAL PURPOSE PROGRAMMING LANGUAGE

Haraprasad Naik

Assistant Professor, Computer Science Department,Utkal University,Vani Vihar,
Bhubaneswar, Odisha 751004, India
hnaik.cs@utkaluniversity.ac.in

Dinesh Patel

Student, Computer Science Department,Utkal University,Vani Vihar,
Bhubaneswar, Odisha 751004, India
dineshpatelbgh@gmail.com

Ajay Kumar Barla

Student, Computer Science Department,Utkal University,Vani Vihar,
Bhubaneswar, Odisha 751004, India
aajaykumarbarla@gmail.com

Subhassish Pradhan

Student, Computer Science Department,Utkal University,Vani Vihar,
Bhubaneswar, Odisha 751004, India
Subhasishp26@gmail.com

Abstract:

This research paper analyzes the evolution of general-purpose programming language over time. The programming language gains popularity based on the number of engineers worldwide, the number of courses available worldwide, etc. There are many programming Languages and this research paper shows that every programming language has its significance This paper shows the common difference between the programming Languages. In this paper the advance features like locales, switch-case with different notation, MVC architecture and LAMBDA expression are discussed.

***Keywords*: Programming Language; MVC Architecture; Locales; Lambda expression.**

1.Introduction

As we know that to communicate or interact with any person. We need a specific language that the person can understand and able to respond easily.

Similarly to communicate with the computer we need a language that a computer can able to understand and respond to our instructions.

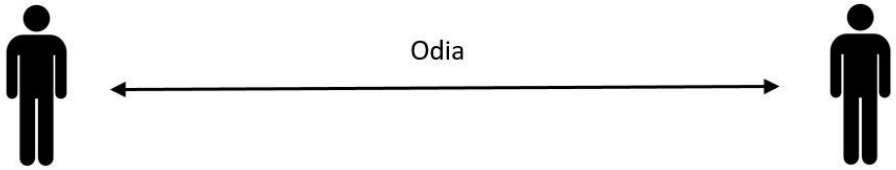

Fig 1. Communication between two person

For example, In the above figure person A is interacting with person B in odia. Which is known to person B and vice-versa.

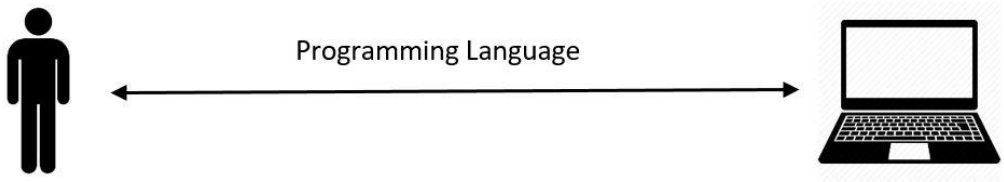

Fig 2. Communication between a person and computer

Similarly in the above figure the programmer is giving instruction by using any programming language which is supported by the respective computer system.

1.1 Language

It is a medium through which a person can communicate with another person or group of person belong to the same geographical area/community/nation.Language plays an important role in exchanging ideas and opinions with each other.[1]

Fig 3. Communication between two person using different languages

1.2 Programming language

A programming language is a language through which the programmer can communicate with the computer[2].Programming language is a set of instruction or command through which a computer understand what task is to be perform.[2]

1.2.1 Characteristic of programming language

- A programming language must be understandable.
- A programming language must be portable, that means for example a C++ program should produce the same result irrespective of CPU, Operating system or C++ compiler.[2]
- A programming language must be memory efficient.
- The library of any programming language must be well documented and organized.
- The source code of any programming language must be easily converted into machine code.

1.2.2 Types of Programming language

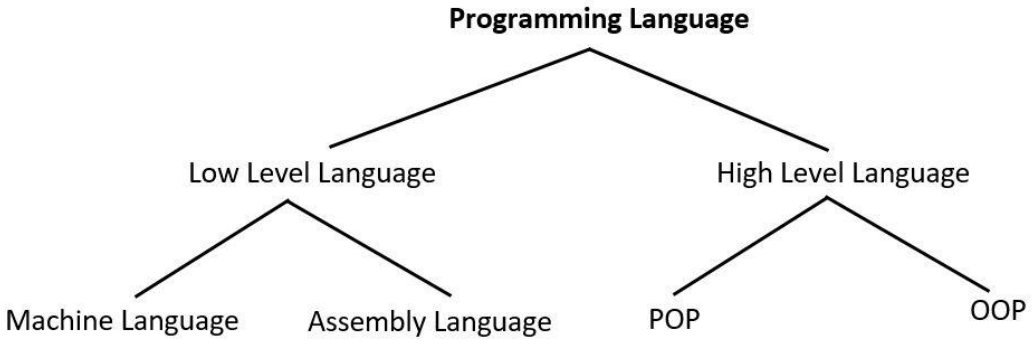

Fig 4.Types of programming language

1.2.2.1 Low Level language

- A low level language is a language that is under-stable by the machine.
- A low level language is not under-stable by human.
- It is native language for computer system.
- It is further devided into two types.
  - Machine language
  - Assembly language

1.2.2.1.1 Machine language

- Machine language is a series of 0's and 1's which is under-stable by computer and it can run on the CPU directly.
- Example : we represent 5 as 101 in machine language.

1.2.2.1.2 Assembly language

- Assembly language is a language that is under-stable only by human and not by computer system.
- It replace the 0's and 1's with the mnemonic code for corresponding machine language.
- It translate the high level language into machine level language.
- It consist of some command like MOV,ADD,TMP etc.

Eg.

```
MOV a, 10
MOV b, 20
ADD a,b
MOV c,a
End
```

1.2.2.2 High level language

A programmer communicate with the computer with the help of language known as high level language basically it is a set of instruction return in any specific language like Java, C++, python etc. to perform any task.

It is further divided into two types.

- Procedural Oriented Programming language.
- Object Oriented programming language.

1.2.2.2.1 Procedural oriented programming language

- It is a set of instruction function variable statement that must be executed in a proper order sequence to perform a specific task.
- Example of pressural language are BASIC, C, FORTRIN etc.
- it is commonly used to solve scientific and engineering problem.

1.2.2.2.2 Object Oriented Programming Language

It is a programming language which uses the concept of object and class to represent the real world entity.

High Level language	Low Level language
It is under-stable by user. It is not memory efficient. It is a portable language.	It is under-stable by computer. It is memory efficient. It is not a portable language.

Table 1. Difference between high level and low level language

1.3 Object oriented programming concept in real world

The following four features are the main property of object oriented programming language.

- Inheritance
- Abstraction
- Polymorphism
- Encapsulation

1.3.1 Inheritance

- It refer to the acquiring the property of parent class by the child class.
- As in family the son or daughter inherits all the properties like skin tone, ears, noses etc. from their parents.

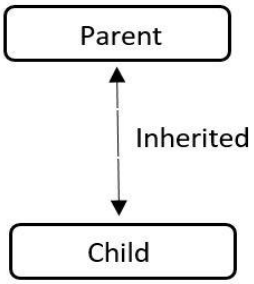

Fig 5. Concept of Inheritance

- Likely in programming concept the child class inherit the properties of parent class by which the courts can be reuse.

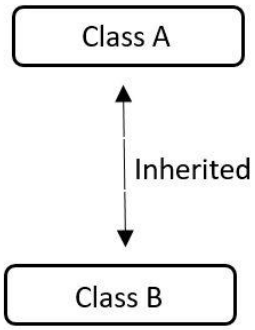

Fig 6. Concept of Inheritance

Here is a parent class and b is a child class which inherit the properties of class A.

.

```
public class Parent {
        void accept(int a,int b)
        {
                int a=7;
                int b=8;
        }
}
class Child extends Parent{
        void add()
        {
                int c=a+b;
                System.out.println("The result is :"+c);
        }
}
```

1.3.2 Abstraction :-

- It refer to showing a line necessary details in hiding the background details.
- For example while we start the engine of the car we only insert the key and turn the key will need not to know about how the engine work behind it.
- Similarly in coffee machine when we need coffee we just need to press the button then the coffee will pour into the glass we need not to know about how the coffee will made behind the machine.

1.3.3 Polymorphism :-

- It means many forms here one function can be modified and used in many way.
- For example in real world a person is an employee in office, a customer in shopping mall, a passenger in bus or train or a son/daughter at home.
- Let us take an example of a policeman, a policeman when goes to the police station he plays the role of a **police** does his duty and after his duty he goes to his home and at home he has a role of **son**, **father**, **husband** to play and the same person when goes to the shopping mall plays a role of a **customer**, when the same person travels in train or bus his role is of a **passenger**.
- There are two type of polymorphism
    - Method overloading
    - Method overriding.

1.3.3.1 Method overloading :-

- When the polymorphism is done within the same class then that type of polymorphism is called method overloading.
- In method overloading inheritance is not necessary.
- In method overloading the structure of the method should be changed.
  Eg.

```
Class A
        {
                void add(int a,int b)
                {
                        int c=a+b;
                }
                void add(float a,float b)
                {
                        float d=a+b;
                }
        }
```

1.3.3.2 Method overriding:-

- When the polymorphism is done within two different classes then that type of polymorphism is called method overriding.
- In method overriding inheritance is necessary.
- In method overriding the structure of the method never change.
  Eg.

```
Class A
{
        void add(int a,int b)
        {
                int c=a+b;
        }
}
Class B extends A
{
        void add(float a,float b)
        {
                float d=a+b;
        }
}
```

1.3.4 Encapsulation :-

- It is a property that finds a data and function into single unit.
- Agent real world college bag consist of pain bottle notebooks books extra we can say that the college bag and capsulate all these things together.
- Similarly in programming concept class encapsulate different method in variable together.

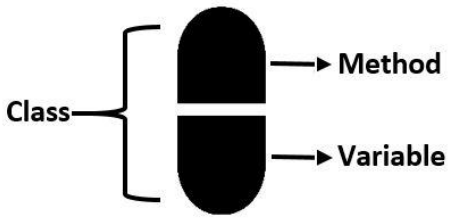

Fig 7. Concept of encapsulation

1.4 History of OOPs concept

- The object oriented programming was first appear in MIT (Massachusetts institute of technology) in late 1950s and early 1960s and the term object oriented was wined by Alan Kay in 1967.
- The first programming language which was design as object oriented watch simula in 1965. The concept of classes, inheritance and virtual methods.
- After simula programming language “smalltalk” was developed by Alan Kay, Dan Ingalls, Adele Goldberg and other at Xerox PARC 1970s.
- The smalltalk was more object oriented than Simula.

1.4.1 Programming language which uses the OOPs concept -

Java	Ruby
Python	Kotlin
C++	PHP
JavaScript	VB

2. Literature review

Sl.No	Author	Publication Year	Publication	Description
01	Janhwi Goyal, Archi Singla, Archee Gupta, Malvika	2020	Why there are So Many Programming Languages?	Understand the importance of a programming language. Understand how a language works. Understand that a computer is a machine like computer. A programming language is set of instructions which tell computer to perform a specific operation for a specific task. The programming languages were invented back in 90's and most of them are still popular among the masses. Programming language have a major role in development of internet which has made the life of humans simpler. Machine language has a set of numeric codes which helps computer to execute any operation directly. It is very simple as it contains a lot of library functions, data types etc. It can be executed on different machines with some changes.

02	Philip Norlin ,Valenti Wannesian	2018	Study of different programming language	A survey was conducted to find out where developers want languages to go in the future, and how developers determine what language is suited for their projects. A programming language is a collection of instructions used to create various kinds of outputs from a computer. They are often divided into two major categories, low level and high level. Low level language are platform dependent meaning they can run on the same hardware and configuration. High level language is human readable form such as Java, and C. Modern languages have improved in the sense that they do not force the programmer to be very specific with what they want to achieve, and leaves most of the hard work for the compilers to deal with. Automatic memory management allows the programmer. to allocate necessary data without needing to explicitly state when. the resources should be returned back to the operating system, decreasing the chances memory leaks.
03	Baishakhi Ray, Daryl Posnett, Vladimir Filkov, Premkumar Devanbu	2014	A Large Scale Study of Programming Languages and Code Quality in GitHub.	Language design has a modest but modest effect on software quality. We also report that functional languages are somewhat better than procedural languages. We have presented a large scale study of language type and use, as it relates to software quality. The data indicates functional languages are better than procedural languages. It suggests that strong typing is better than weak typing. We have examined the interactions of language, domain, and defect type. That the defect proneness of languages in general is not associated with software domains. Languages are more related to individual bug categories than bugs overall.
04	Ghazala Shafi Sheikh, Noman Islam	2016	A qualitative study of major programming languages: teaching programming	A comparative analysis of contemporary programming languages for teaching computer

			languages to computer science students	science to computer science students is carried out. It is based on a careful examination of current curriculum and market demands and provides recommendations on the selection of a programming language. Java is the best general purpose programming languages to be used for teaching computer science concepts. The most widely used for development for JavaScript is Web Storm which is available with some licensing fee. An updated analysis of major programming languages of recent time has been performed.
05	Matt Sherman	2015	Why Are There So Many Programming Languages?	Is there a single language in the world that isn't a language that we need to be able to communicate with our clients? A programming language is a tool for humans to express ideas to computers. I started with FORTRAN, a little COBOL, a LOT of assembler, UCSD Pascal, a few months of ADA, C and C++ for a while. I have been using Java and JavaScript since. Programming languages are tools, and we choose different tools for different jobs. Stack Overflow chose C# mostly because that's what our founders knew. A good ecosystem – Ruby has a great one, for example – can make the individual developer more successful.

3. Evolution of Programming Language :-

3.1 Generation of programming language :-

On the basis of performance of programming language and their robustness they can generated into different generation.[7]

- First Generation Language (1GL)
- Second Generation Language (2GL)
- Third Generation Language (3GL)
- Fourth Generation Language (4GL)
- Fifth Generation Language (5GL)
- Sixth Generation Language (6GL)

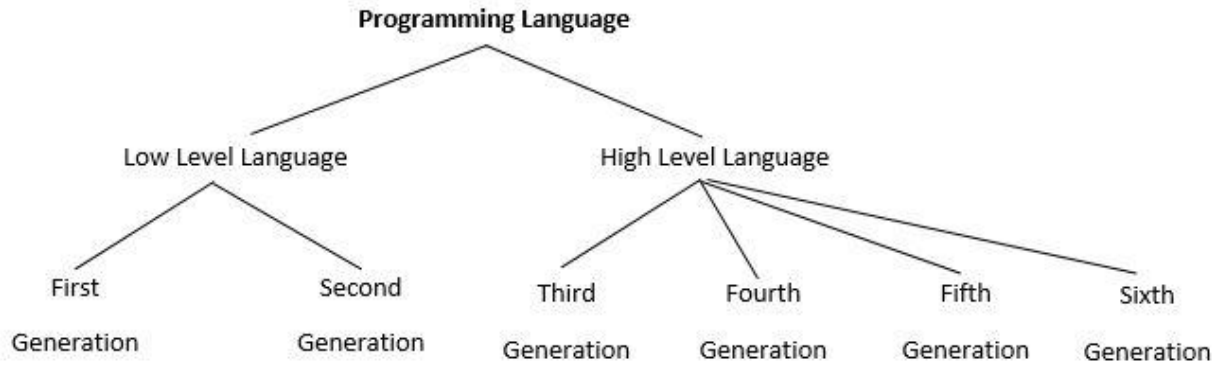

Fig 8. Types of programming language

3.1.1 First Generation Language :-

The first generation language is also known as the machine language or 1G language.The 1st generation language are machine independent.It is directly written in binary quotes i.e 0 &1.In first generation language no translator are required as it is written in binary language.As it is written binary language it is very difficult to understand so this was the major problem faced by the developers and for this reason only the Second generation language came into existence.[7]

3.2.2 Second Generation Language :-

Second generation language or 2G language are also known as assembly language.And the assembly language contain the codes that a human or programmer can understand.The instruction code in 2G languages consist of a meaningful abbreviation of mnemonic.The main advantage of the 2G language is,as it contains the human understandable codes so the programmers can easily understand the codes.hence the programmers can easily understand the code so the modifications and the corrections becomes easy for them.

The main drawback of 2nd generation languages is that an assembler is required to execute the code.And these languages are machine dependent which means for the code written in one machine cannot be executed on another machine.

3.2.3 Third Generation Language :-

Third Generation language is a language in which instruction are written in English like sentence that a human can understand easily.It offer more abstraction to hide and necessary details as compared to the previous generation.FORTRAN, COBOL, ALGOL users the first 3G languages that were introduced in late 1950s.The most popular 3G language used now-a-days are C, C++, Java, BASIC and Pascal.

The 3rd generation language supports structure programming and the learning and understanding of The 3G language is very easy.As 3G language is a machine dependent so for different machine a different compiler is required.

3.2.4 Fourth Generation Language :-

The fourth generation language at the non procedural language,with the help 4G language we can use and access the database i.e we can store, retrieve, update, delete the data in database,
Eg. SQL , Focus, Fox-pro are the 4G language.

4G language is very easy to learn and understand and it requires very less time to execute the instruction.Compared to the previous languages it requires more memory and it is less flexible and less portable.

3.2.5 Fifth Generation Language :-

The 5G language are built based on the concept of artificial intelligence.5G language you just the concept of solving the problems by developing the application.The 5G languages use AI to make decision which reduces the efforts of the programmers.E.g. PROLOG, LISP are 5G language.

The major drawback of the 5G languages are they are very complex to write and understand,and the resources which are required are very expensive.

3.2.6 Sixth generation language :-

6G Language is a very high programming language,it offers extreme level of abstraction from hardware.It offers extreme level of abstraction from hardware.

Programming language based on visual development is a 6G language and hence no codes are required as it is based on the visual development. 6G languages are very expensive to implement.

4. Common and difference between programming language :-

Not every programming languages are equally similar some programming languages are very different and some are very similar.The programming languages like C,C++, Java,python are used for the application development.The programming languages like JavaScript and PHP etc used for the web development and programming.

Application Development	Web Development
• C • C++ • Java • Python	• PHP • javaScript • HTML

Table 2. Difference between application development and web development

The programming languages have the similarities based on the nature of execution that is compiled and interpreted.

The programming languages like C, C++, C#, COBOL etc are compiled programming languages and the programming languages like JavaScript, python, BASIC etc are interpreted language.

Compiled language	Interpreted language
• C • C++ • Go • COBOL	• Python • PHP • Ruby • JavaScript

Table 3. Difference between compiled language and interpreted language

4.1 Compile

The process in which the HLL(High level language) is converted into law level language is known as compile.To compile any program written by programmer need a compiler interpreter which convert high level language into low level language.The main purpose of a compilation is to convert the human readable code into a format that the computer can understand and execute.

4.2 Compiler

The software program which convert a program written in high level language(HLL) to a low level language(LLL) is called as compiler.In other words it is a program that translate the source code into machine code or byte code which is machine independent.

4.3 Interpreter

It is also a program that translates the high level language into machine code line by line, translating each line into machine code as it is encountered.When a program run or execute using an interpreter if any error encounters at any line of code it will stop the program and report the error or mistake to the user or programmer.

Example :

Let us consider a scenario where a programmer or software developer has written the 1521 line of code (LoC) and the program is being compiled using a compiler, after compilation over it will be generated byte code or machine code which can be directly executed by the computer. The generated byte code can be run on any computer and there is no need for original source code anymore.

If the program written by the programmer has an error statement at the line of 1405 and the program is executed by using the interpreter then up to 1405 lines of code(LoC) it is executed smoothly after encountering an error interpreter immediately stops the program and reports the error or mistake to the user or programmer.

Here are some examples of programming languages and whether there associated with the compiler or an interpreter.

Languages commonly associated with compilers:-

- C
- C++
- Java
- Haskell
- Rust
- Go
- Swift

Language commonly associated with the interpreter:-

- Python
- Ruby
- JavaScript
- PHP

Some programming languages like c and c++ can be implemented using both compiler and interpreter.

4.4 History of compiler and interpreter

In order to know the history of compiler and interpreter we need to go back to the early days of Computer programming. In the year 1950s and 1960s, programmer road program in machine language which was very complex and difficult to understand. To write programs in those days programmers should have knowledge of multiple machine languages to work and multiple computers.

In the late 1950s, John backup and his team at IBM developed the first compiler for the FORTRAN programming language. Which made it possible to write the program in high level language which can be compiled to low level language easily.

In the 1970s, with the development of C programming language compiler became the most popular and widely used. C compiler is used to develop operating system and software program and C programming language become very popular all the time.

First interpreter was developed in the year 1958s by John Mc Carthy for the programming language LISP. In the year 1970s, the BASIC programming language became popular and many computer came with the BASIC interpreter built-in.Now it is both compiler and interpreter widely used to translate the high level language into machine level language.

4.5 Control Statement

A control statement is a programming language statement that controls the flow of execution in a program. It allows you to specify the order in which statements are executed, based on certain conditions or loops.

Control statements are an essential part of programming that allow developers to control the flow of execution in their programs. These statements include conditional statements, loops, and branching statements. Conditional statements, such as if-else statements, allow developers to execute different blocks of code based on a particular condition being met. Loops, such as for loops and while loops, allow developers to repeat a block of code multiple times. Branching statements, such as break and continue, allow developers to alter the normal flow of execution in their programs.

Control statements are important because they allow developers to create more dynamic and flexible programs. By using the control statement the line of codes can be reduced to some extent. Without control statements, programs would be limited in their ability to respond to different conditions or to repeat tasks efficiently. By using control statements, developers can create programs that are more responsive to user input, more efficient in their execution, and more adaptable to changing conditions. Overall, control statements are an essential tool for any developer looking to create effective and efficient programs.

Some common types of control statements include:

Conditional statements - these allow you to execute a certain block of code based on whether a certain condition is true or false. For example, "if" statements, "else" statements, and "switch" statements.

Looping statements - these allow you to execute a certain block of code repeatedly, based on certain conditions. For example, "for" loops, "while" loops, and "do-while" loops.

Jump statements - these allow you to transfer control to a different part of the program. For example, "break" statements, "continue" statements, and "goto" statements.

Control statements are essential in programming, as they enable you to make decisions and automate the tasks, which makes your code more efficient and easier to manage.

Which control statement are supported by which programming language are given in table below.

	C	C++	Java	Python	JavaScript	Ruby	C#	PHP
if	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
If-else	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
switch	Yes	Yes	Yes	No	Yes	No	Yes	Yes
for	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
while	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Do-while	Yes	Yes	Yes	No	Yes	No	Yes	Yes
break	Yes	Yes	Yes	Yes	No	No	Yes	Yes
continue	Yes	Yes	Yes	Yes	No	Next	Yes	Yes
goto	Yes	Yes	No	Yes	No	No	Yes	Yes
pass	No	No	No	Yes	No	No	No	No
until	No	No	No	No	No	Yes	No	No

Table 4. Control statement of different programming language

5. Comparison of programming languages using features

5.1 Method overloading

It is a feature where two more more function have the same name but different parameters.

5.2 Multiple inheritance

Multiple inheritance is a feature where one child class inherits the property of the two or more parent class.

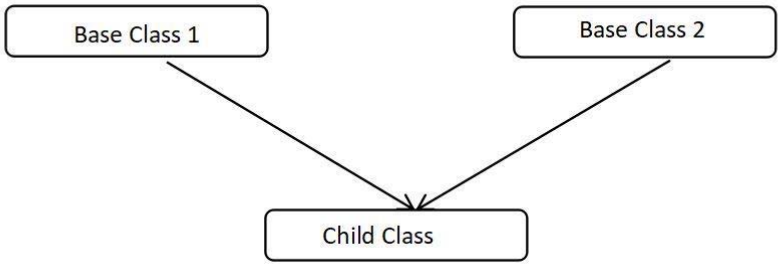

Fig 9. Concept of multiple inheritance

5.3 Operator overloading

It is a compile time polymorphism.where the operator is overloaded to use it with a user defined data type.

c++, python supports operator overloading.

5.4 for each Loop

It is a looping statement which is used to iterate over the container like arrays, vector etc ,without initialization and increment/ decrement.

Syntax :-

```
for ( data type variable-name : container type)
{
operation using variable name
}
```

It is supported by c++, java, PHP.

5.5 structure

Structures are used to store the different attribute of a single entity. For example a student have his name, address, roll number etc ,so these all information can be stored by creating a structures.

Syntax for creating structures:

```
Struct structure-name
{
data_type member 1;
data_type member 2;
data_type member n;
};
```

C,c++ have "structure".

5.6 Pointers

Pointer is a variable which is used to store the memory address of another variable. The pointer variable can be of different data type.

e.g

```
int a=7;
int * d= &a;
```

C, C++ supports the pointer concept.

Features of different programming language :-

Features	C	C++	Java	Python	PHP
Method Overloading	No	Yes	Yes	No	No
Multiple Inheritance	No	Yes	No	Yes	No
Operator Overloading	No	Yes	No	Yes	No
Foreach Loop	No	Yes	Yes	Yes	Yes
Structure	Yes	Yes	No	No	No
Pointers	Yes	Yes	No	No	No

Table 5. Features of different programming language

6. Analysis and Discussion :

6.1 Locales :

It used to represent a geographical,cultural or political or cultural region to display the required text or numbers or any operation in a required language,I.g an English text can be displayed in Chinese text.

As different region,states,countries have the different languages for.I.g Odia language is used in Odisha, Marathi is used in Maharashtra etc.So,to use an application in different religions we need to display the required texts,numbers,operations in the required languages and This is done by using Locales.

A locale is typically identified using a communication of language and country/region codes.such as “en_US” for English in the united states,are “hi_IN” for Hindi in India.

Creating a Locale:

There are several different ways to create a Locale object.

Builder:

Using Locale.Builder you can construct a Locale object that conforms to BCP 47 syntax.

Constructors:

The Locale class provides three constructors:

Locale(String language)

Locale(String language, String country)

Locale(String language, String country, String variant)

These constructors allow you to create a Locale object with language, country and variant, but you cannot specify script or extensions.

Factory Methods:

The method for Language Tag(java.lang.String) creates a Locale object for a well-formed BCP 47 language tag.

6.2 Lambda expression

Lambda expression is a formal system in mathematical logic that has been widely used as a theoretical foundation for functional programming languages.[11] It provides a mathematical framework for expressing computations in terms of anonymous functions, also known as lambda functions. Many modern programming languages, such as Lisp, Haskell, and Python, have adopted concepts from lambda expression to enable functional programming paradigms.

In programming languages, lambda expression is typically used to represent and manipulate functions as first-class citizens, which means that functions can be passed as arguments to other functions, returned as values from other functions, and stored in variables. This allows for higher-order functions, which are functions that take other functions as arguments or return them as values, and can be used to implement powerful and flexible programming patterns.

Note that while Java supports lambda expressions and functional programming concepts, it is primarily an object-oriented programming language and not a purely functional programming language like Haskell or Lisp. Lambda expressions in Java are limited in their expressiveness compared to full-fledged lambda expression, but they can still be used to implement functional programming concepts in Java code.[11]

Although C and C++ is best or powerful language but is does not have the support for lamda expression which is a demerit of these programming languages.

SYNTAX:

Parameter->expression

If we have more than one parameter we can write

(parameter1,parameter2 ->expression

6.3 Switch Case with Different Notation:

A switch statement is a branching statement in which the value of the expression is compared with the values of each case and if the match is found the following statement is executed. The switch statement is a fall-though which means all the statements are executed after the first match is found until the break statement is encountered.

The switch statement in C and C++ only works with the integer or character constant case values as in java the switch statement works with the byte,short,int,long and some wrapper types like Byte,Short,Int and Long.

Syntax:

```
Switch(expression)
        {
                Case value1:
    //code to be executed;
                Break;
                Case value2:
    //code to be executed;
                Break;
                .
                .
                Default:
    // code to be executed when none of the cases are
matched;
        }
```

In the above syntax we have to write the break statement after every case otherwise it will face fall-through i.e if the match is found all the cases are executed until the break statement is encountered. So to avoid writing the break statement repeatedly there is another way to write the switch statement.

The syntax is given below:

```
Case label_1,case label_2,…….,label _n ->expression
```

6.4 MVC architecture:

MVC architecture stands for Model View Controller architecture.It is a design pattern in a web development field .It is a way to organize the code,it specifies that an application must consist of data model,presentation information and control information.

diagram:

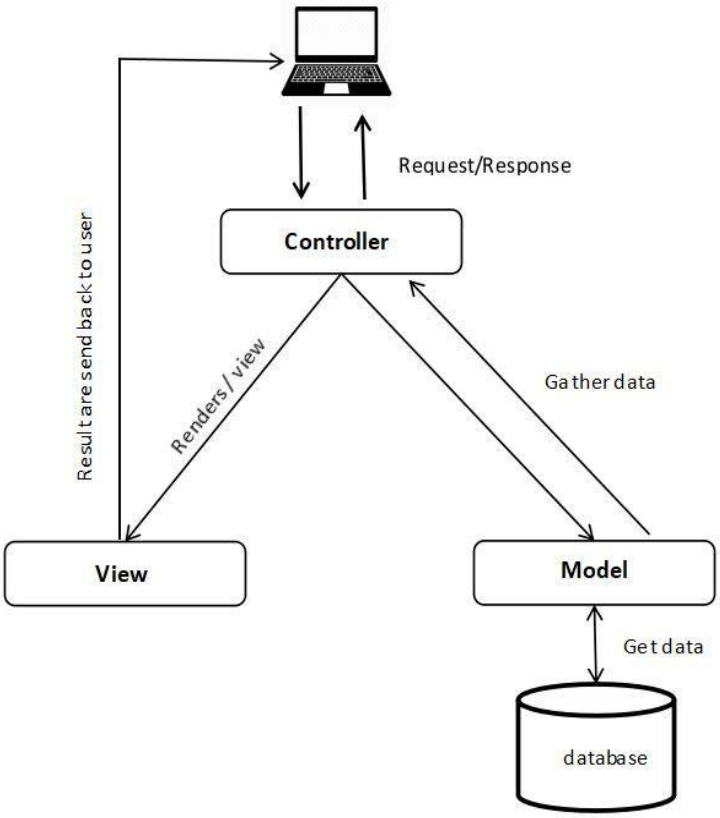

Fig 10. MVC Architecture

The model layer in the MVC architecture acts a data layer for the application.The model object fetch and store the model state in the database.[9] View layer in MVC architecture represents the data in the model and the controller layer is between model and view layer is gathers all the data that has been requested.[9]

The client browser first send the request to the controller on the server side,the controller then the data which is being requested then the data which is being retrieved by the controller is transferred to the view layer and then the view sends the result again back to the browser.

The significance of the MVC architecture is that the user interface and the business logic are separated and it is also very easy to maintain as the different components of this MVC architecture are maintained individually.

The car driving mechanism is an example of the MVC model. As the car mechanism consists of the view i.e gear,clutch,steering etc, the controller i.e engine and the model i.e the fuel tank.[9] So when someone drives

the car it only interacts with the user interface but at the background the controller is taking the fuel from the fuel tank.

The disadvantage of this model is it is lil bit complex and it has new layers of abstraction.It is difficult to read and reuse this MVC model architecture.[9]

6.5 **TIOBE community index:**

The popularity of the programming language is not stable.if the programming language is under top 10 today then tomorrow it may slide down to top 50 as everyday a new programming language is developed but it is very difficult for it to chart in the top 100.[10] It takes at least 10 years to chart in the top 100,but there are the languages which are under top 100 and which are younger than 10 years. Some of them are #rust(17) #swift(14) #crystal(48).

The oldest programming language FORTRAN is again in top 20 according the TIOBE index because of the increasing demand of computational power.[10]

This TIOBE programming community index shows the popularity of the programming language and it is updated once in a month.The TIOBE index is not about the best programming languages or the numbers of lines of code in programming language but it is based on the number of engineers worldwide,number of courses available worldwide etc.[10]

The top 10 programming languages according to the TIOBE are given below:

May-22	May-23	Programming Language	Rating
1	1	Python	13.45%
2	2	C	13.35%
3	3	Java	12.22%
4	4	C++	11.96%
5	5	C#	7.43%
6	6	Visual basic	3.84%
7	7	JavaScrpit	2.44%
10	8	PHP	1.59%
9	9	SQL	1.48%
8	10	Assembly Language	1.20%

Table 6. Top 10 programming language

7. Final Remarks

Different programming languages are used for the different purposes. Every programming language has its own significance.

7.1 python

Python is very popular language to work with artificial intelligence, machine learning, data analytic. It is also used in web development.

7.2 javascript

It is a programming language which is used to developed the front end. It makes the website interactive and handles all the alerts, pop-up, events of the web pages.

7.3 Java

It is a very powerful object oriented programming language. It is used to develop desktop and mobile applications. it allow the developers to develop the software in the industry level.

7.4 C/C++

This languages are very low level programming languages used to develop operating system, databases and compilers, real time systems, IOT, embedded system, games etc.

7.5 C#

This programming language is used for developing the project like game development, server side programming, web development, creating web frame, mobile application etc.

7.6 PHP

It is called the scripting language. It is embedded with HTML to add more functionality to the web pages. it is used for web development i.e to make website more dynamic and interactive.

8. Conclusion :

Many programming languages out there is that different problems require different tools to solve them.Each programming language has certain features and characteristics that make it suitable for specific tasks.Let us understand this with the example: If we have any issue In our teeth we will go the dentist,similarly we will go to the cardiologist if we have any problem in heart and we will go to the eye specialist if we have any problem in our eyes.

Let us take another example: suppose we want to cut a tree and we have two tools i.e an axe and a knife then we have to use an axe to cut a tree because for cutting a tree an axe is an appropriate tool,unlikely we have to use a knife to cut an apple because it is an appropriate tool for cutting the apple.

Similarly among different programming languages we have to choose an appropriate programming language for particular task,like we use “JavaScript” for the front end designing and “python” for the beck end designing.we use “java” to develop an application in an enterprise level. “C” language is used for developing system software. “c++” is used in developing operating systems and applications.

So for different purposes an appropriate programming language is used that is the reason we have so many programming languages as each programming language has its own significance.

Funding

No funding is provided for the preparation of manuscripts.

Conflicts of interest

The authors have no conflicts of interest to declare.

Reference

[1] Sitti Rabiah, “Language as a tool for communication and cultural reality dis-closer”, November,2012
[2] Janhwi Goyal, Archi Singla, Archee Gupta, Malvika, “Why there are so many programming language”, Internatioal Journal of research in engineering science and management, volume-3, january-2020
[3] B.Ray, D.posnett, V.Filkov, P.Devanbu, “A large scale study of programming language and code quality in github”, 2017
[4] G.S. Sheikh, Noman Islam, “A qualitative study of major programming language : teaching programming language to computer science student”, International Journal of information and communication technology, 2016.
[5] https://www.freecodecamp.org/news/why-are-there-so-many-programming-languages/
[6] Matt Sherman, “Why are there so many programming language?”,2015
[7] https://en.wikipedia.org/wiki/Programming_language_generations
[8] https://en.wikipedia.org/wiki/History_of_compiler_construction
[9] Robert Eckstein, “Java SE Application design with MVC”, March 2007
[10] https://www.tiobe.com/tiobe-index/
[11] Davood Mazinanian, Ameya Ketkar, Nikolaos Tsantalis, and Danny Dig. 2017. Understanding the use of lambda expressions in Java. Proc. ACM Program. Lang. 1, OOPSLA, Article 85 (October 2017), 31 pages.

Authors Profile :

Haraprasad Naik is Assistant Professor (S-II) in the P.G Dept. Of Computer Science & Application Utkal University, Bhubaneswar, Odisha, India. He has obtained Master in Computer Application from College of Engineering and Technology Bhubaneswar under BPUT(Biju Pattnaik University of Technology) and currently pursuing Ph.D. in Computer Science from Utkal University, Vani Vihar, Bhubaneswar. His current area of interest are Artificial Intelligence, Programming languages, OLTP, Image Processing etc.

Dinesh Patel is the final year student of MSc Computer Science in the department of Computer Science & Application, Utkal University, Bhubaneswar, Odisha, India.He has obtained the Bachelor of Science in Computer Science from Gangadhar Meher University, Sambalpur.He is currently working in the field of computer language domain applicable to artificial intelligence.

Ajay Kumar Barla is the final year student of MSc Computer Science in the department of Computer Science & Application, Utkal University, Bhubaneswar, Odisha,India.He has obtained the Bachelor of Science in Computer Science from Gangadhar Meher University, Sambalpur.He is currently working in the field of computer language domain applicable to artificial intelligence.

Subhashish Pradhan is the final year student of MSc Computer Science in the department of Computer Science & Application, Utkal University, Bhubaneswar, Odisha, India.He has obtained the Bachelor of Science in Computer Science from Khallikote University, Berhampur.He is currently working in the field of computer language domain applicable to artificial intelligence.