

## A New Modern Object-Oriented Programming Languages

Prof. Shilpa Shitole, Rohit Maurya, Tanaya Pawar, Siya Randhe

Department of Computer Engineering, MIT Polytechnic, Pune, Maharashtra, India

### ABSTRACT

#### Article Info

Volume 7, Issue 3

Page Number: 683-685

#### Publication Issue :

May-June-2021

#### Article History

Accepted : 12 June 2021

Published : 20 June 2021

Industries evolve. Our thinking changes as well. Programming languages need evolution too. "The thing is that ideas for new features with its ways of thinking will be flourished, and so perfectly designed those languages won't be perfect anymore." Where did logical programming go? "Notice that you can use this paradigm and just provide a set of constraints for a website and expect the website to develop automatically based on them." It is possible to implement that. Likewise, new paradigms will sooner or later be born. It can't be that we've explored everything. "Technologies are born likewise the old way of thinking, which represents by the previous programming languages might not be adequate. This project is an open-source modern object-oriented programming language that aims to bridge the gap between modern expressive programming paradigms like python and strictly typed rigid languages like Java and C#. Our goal is to provide the usefulness of an object-oriented programming language while holding the simplicity of an expressive programming language without having to sacrifice performance.

**Keywords:** Programming Language, Modern Object-Oriented, Compiler, Garbage Collection, Statically Typed

### I. INTRODUCTION

The First thing is that we learn more about programming, Classes and methods are open by default in Java. Language like kotlin made them both final by default because developers were highly overusing inheritance when they shouldn't have been using inheritance more than required.

Java class members were package-private by default modifier was rarely used. "Languages like kotlin don't allow it, but instead class members are public by

default reason being this is the most common converter for them." We change our habits as we learn, so languages should also change with us."

The third thing is that paradigms change. I see inactivity in terms of programming paradigms, but we still have some to introduce into everyday practice.

Where did logical programming go? Can we use this paradigm and provide a set of limitations for a website and expect the website to be developed

automatically based on them. It is possible to implement that. It can't be that we've explored everything.

## II. DISCUSSION

The examination of this topic based on the Program was done to perfection as to don't get any difficulty further while making it. Required Coding Languages was well studied and execute properly. Throughout the study of used SLDC approach model was the waterfall model as the procedure was in a linear and sequential approach. The use of C++ accurately and effectively.

## III. LITERATURE SURVEY:

Expression tasks for novice programmers: Turning the attention to objectivity, reliability and validity:- This paper presents the importance of understanding and transfer of mathematical expressions in a programming language essentialness for studying engineering.

General-purpose multi-paradigm programming languages: an enabling technology for constructing complex systems: This paper extends Multi-paradigm programming languages and how they have been envisioned as a vehicle for constructing large and complex heterogeneous systems, such as a stock market exchange or a telecommunications network. General-purpose multi-paradigm languages, as opposed to hybrid multi-paradigm languages, embody several prevalent programming paradigms without being motivated by a single problem.

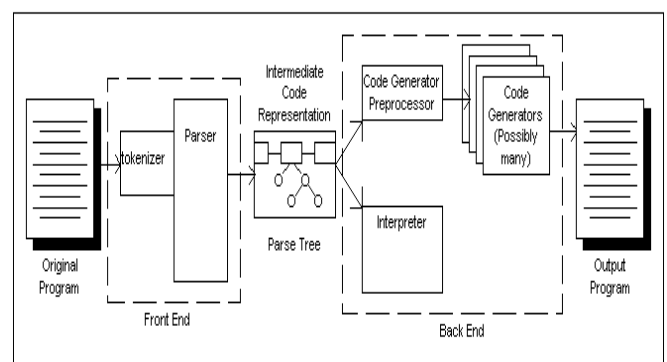
## IV. FEATURES

1. Feasible Design/User Friendly
2. Updated Usage of Technology.
3. Open Source.

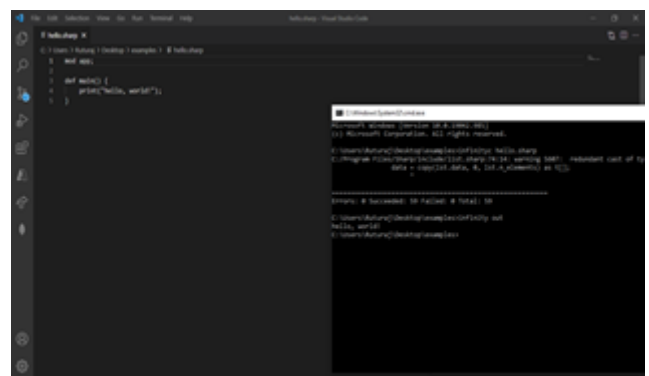
4. Installations are minimal.
5. Language Feature Virtual Machine.
6. multi-Thread system/Inter-Thread communication.
7. Super-classes/garbage collector.
8. Custom JNI/JIT compiler
9. Custom memory allocation

## V. SYSTEM ARCHITECTURE

Our project is an open source a modern object-oriented programming language that aims to compress the gap between modern expressive programming language paradigms and strictly typed rigid languages. Our Aim is to provide the usefulness of an object oriented programming language, while holding the simplicity of an expressive programming language without having to sacrifice performance.

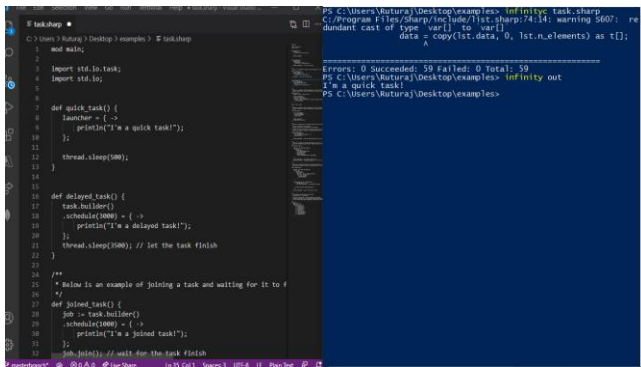


## VI. RESULT/OUTPUT



Output-1: Execution Using CMD

Here we have implemented a simple hello world program using our programming Language



Output-2: Execution Using Powershell

Output 2 show implementation and execution of multi-tasking library .

## VII. SOFTWARE SPECIFICATION:

- C++
- C shell
- CMake
- Assembly

## VIII. CONCLUSION

In future, we suggest using this programming language in manufacturing and design applications, as it allows people to reduce the effort involved. For instance, it can be used while designing blueprints and flowcharts. Sharp makes it possible for the designers and engineers to produce these flowcharts and blueprints accurately..

Because of maintenance cost of this technology is very low compared to the current using method that is economically this method became cheap and also results are very accurate.

## IX. ACKNOWLEDGMENT

It is our privilege to precise our sincerest regards to our project guide, Prof. Mrs.S.N.Shitole for her valuable inputs, able guidance ,encouragement, wholehearted cooperation and constructive criticism throughout the duration of project. We deeply

express our sincere thanks to our Principle Dr.R.S. Kale and Head of Computer Engineering Department Prof. J.G.Mante(Khurpade).We take this chance to thank all our lecture who have directly or indirectly helped our Project.

## X. REFERENCES

- [1]. Compilers: Principles, Techniques, and Tools 2nd Edition by Alfred Aho (Author), Monica Lam (Author), Ravi Sethi (Author), Jeffrey Ullman (Author).
- [2]. <https://ieeexplore.ieee.org/document/7344070>.
- [3]. <https://ieeexplore.ieee.org/document/479354>
- [4]. <https://www.stackoverflow.com>
- [5]. <https://en.wikipedia.org/wiki/Compiler>

### Cite this article as :

Prof. Shilpa Shitole, Rohit Maurya, Tanaya Pawar, Siya Randhe, "A New modern Object-Oriented Programming Languages", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7, Issue 3, pp.683-685, May-June-2021. Available at  
doi : <https://doi.org/10.32628/CSEIT2173148>  
Journal URL : <https://ijsrcseit.com/CSEIT2173148>