UPCISS

UPCISS

Computer Language

- What is Computer Language ?
- Types of Language ?
- What is Programming Language ?
- Difference Between POP and OOP ?
- Assembler, Compiler & Interpreter ?

YouTube UPCISS UPCISS Subscribe

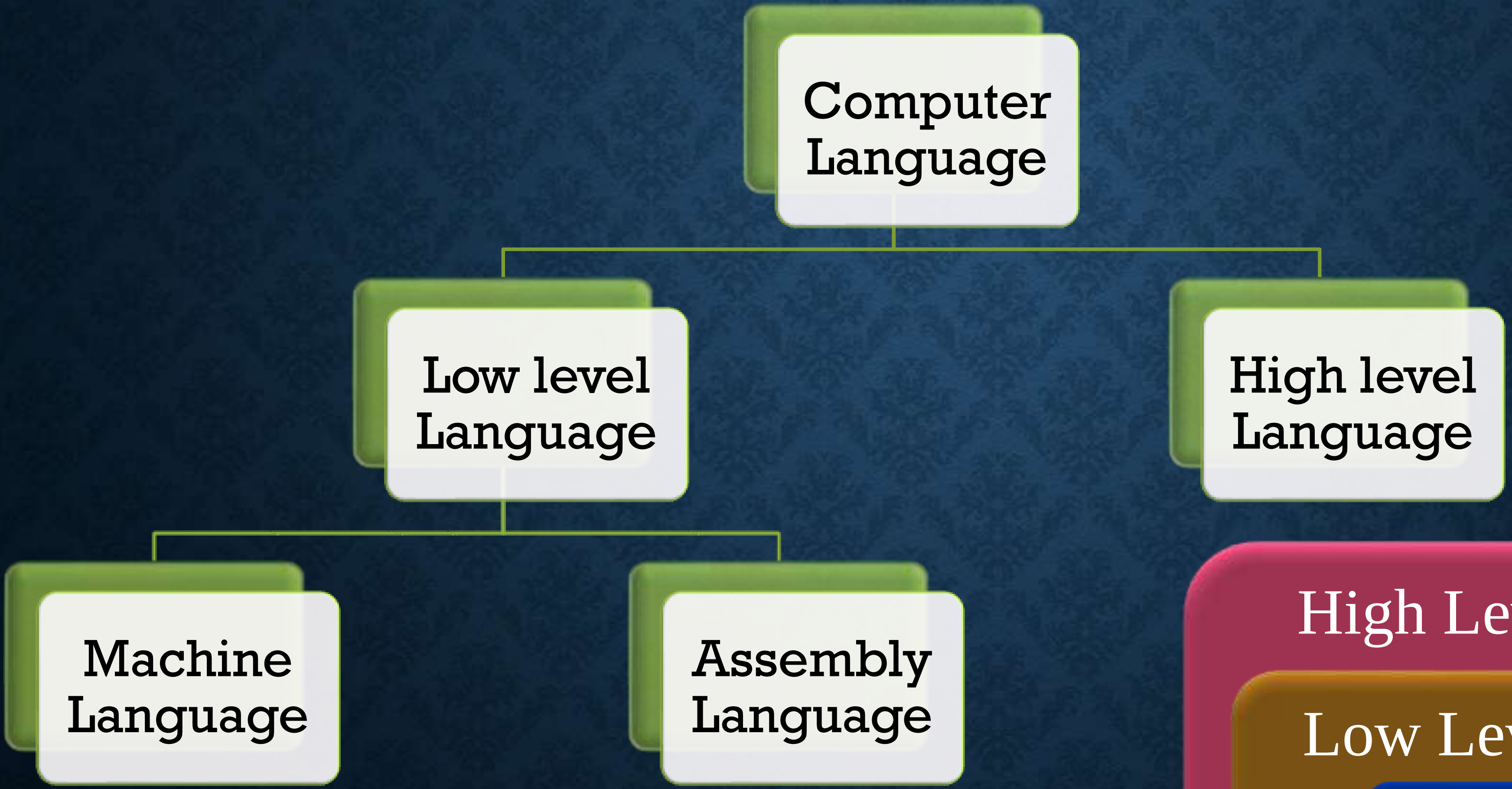
What is Computer Language

UPCISS

Language :- जिस के माध्यम से हम अपने विचार व्यक्त करते है। शब्दों के ऐसे समूह को भाषा कहा जाता है। जिसका कोई अर्थ होता है। कम्प्यूटर से कार्य कराने के लिये जिस भाषा में अपने विचारों को व्यक्त करते है वह यूजर भाषा होती है। और कम्प्यूटर जिस भाषा को समझता है वह मशीन भाषा या बाईनरी (0–1) कहलाती है। यूजर भाषा को मशीन भाषा में बदलने के लिये एक सिस्टम साफ्टवेयर की आवश्यकता होती है। जिसे Translator कहा जाता है।

Types of Language

UPCISS

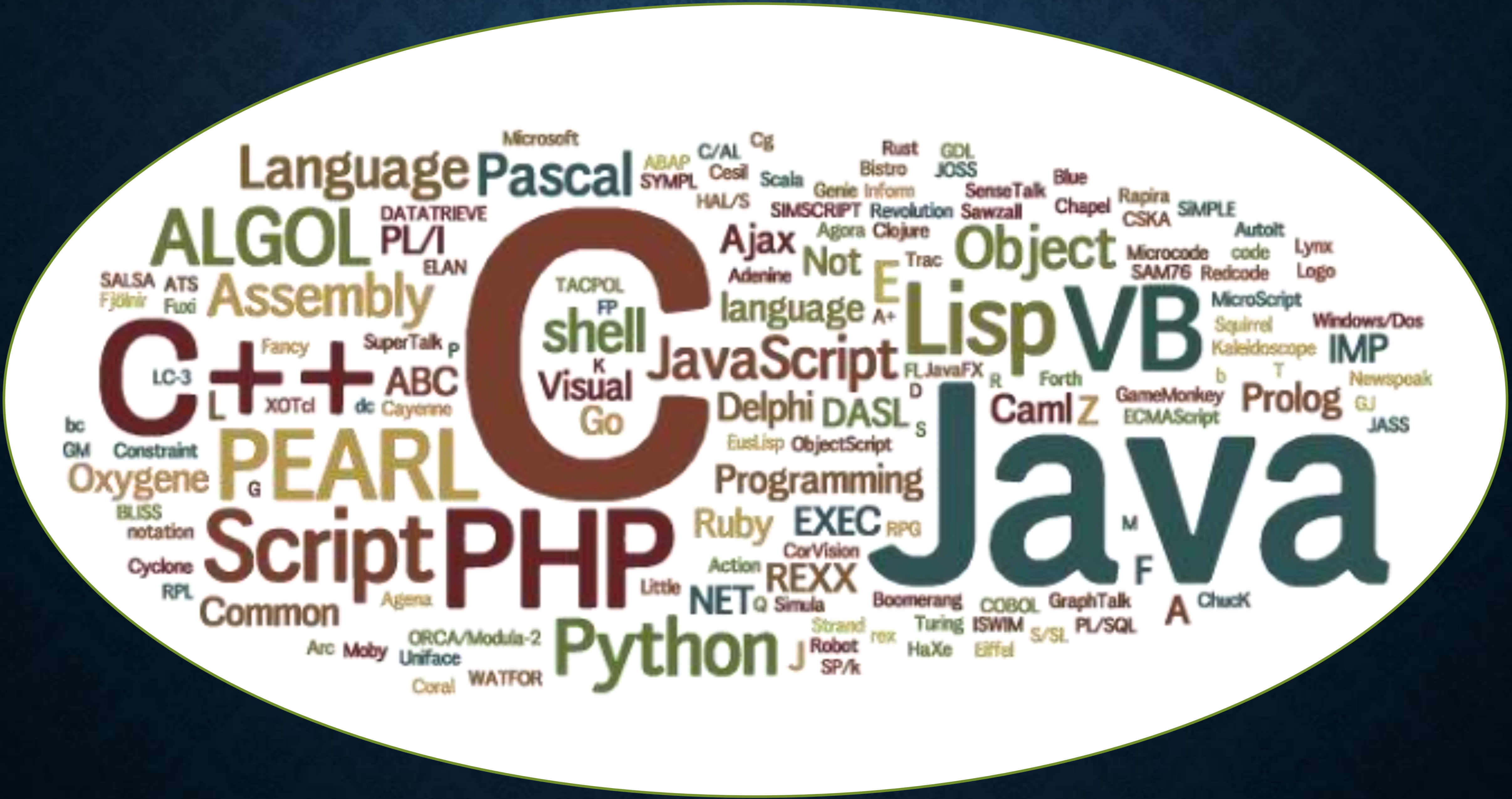


What is Programming Language

UPCISS

Programming Language मतलब एक ऐसी लैंग्वेज जिसे बनाया गया है डिजिटल मषीन से कार्य कराने के लिए। सरल शब्दों में कहें तो प्रोग्रामिंग लैंग्वेज का इस्तेमाल **Software and Web application** को डेवलप करने के लिए किया जाता है। उन **Software program** के द्वारा पहले से ही तय किये कार्य कर सकते हैं।

इस समय लगभग 2,500 प्रोग्रामिंग भाषाएं मौजूद हैं।


Microsoft
Language
Pascal
ABAP
SYMPL
C/AL
Cg
Cesil
HAL/S
Scala
Rust
Bistro
GDL
JOSS
Genie
Inform
SenseTalk
Blue
DATATRIEVE
SIMSCRIPT
Revolution
Sawzall
Chapel
Rapira
CSKA
SIMPLE
ALGOL
PL/I
Ajax
Agora
Clojure
Object
AutoIt
Microcode
code
Lynx
ELAN
Not
Trac
Adenine
E
SAM76
Redcode
Logo
SALSA
ATS
Assembly
TACPOL
Fjölnir
Fuxi
C
shell
FP
language
Lisp
VB
MicroScript
Squirrel
Windows/Dos
A+
Fancy
SuperTalk
P
Kaleidoscope
IMP
C++
JavaScript
LC-3
ABC
K
Visual
FL
JavaFX
R
Forth
b
T
Newspeak
L
XOTcl
dc
Cayenne
D
Delphi
DASL
Caml
Z
GameMonkey
ECMAScript
Prolog
GJ
bc
Go
S
JASS
GM
Constraint
EusLisp
ObjectScript
Oxygene
PEARL
Programming
Java
G
BLISS
Ruby
EXEC
RPG
M
notation
Script
PHP
CorVision
Cyclone
Action
REXX
F
Little
RPL
NET
Q
Simula
Boomerang
COBOL
GraphTalk
ChucK
A
Common
Agena
Python
Strand
rex
Turing
ISWIM
S/SL
PL/SQL
Arc
Moby
ORCA/Modula-2
Robot
Uniface
J
SP/k
HaXe
Eiffel
Coral
WATFOR

Machine Level Language

Machine Level Language:- यह वह भाषा होती है, जो कम्प्यूटर समझता है। इसे बाइनरी भाषा कहा जाता है। इसका प्रयोग प्रथम पीढ़ी के कम्प्यूटरस में किया गया था । इसमें लिखे गये प्रोग्राम तीव्र गति से रन होते है। क्यों कि इस पर सीधे प्रोसेंसिंग की जाती है। इसमें कोई भी Translator साफ्टवेयर की आवश्यकता नहीं होती थी। इसका का आउटपुट भी इसी भाषा में आता है। इसमें प्रोग्रामिंग करना कठिन होता है। यह मशीन पर आधारित होती है।

Assembly Language

Assembly Language :- Machine language की कमीयों को दूर करने के लिये आसेम्बली भाषा का विकास किया गया । इसमें इनपुट के लिए बाईनरी भाषा के स्थान पर mnemonic code का प्रयोग किया गया था। जिनको याद रखना आसान था । इन कोड को माशीन भाषा मे बदलने के लिये आसेम्बलर का प्रयोग किया जाता था जो एक सिस्टम Software है। इस भाषा में प्रोग्रामिंग करना मषीन भाषा की अपेक्षा सरल होता है। इसका प्रयोग द्वितीय पीढ़ी के कम्प्यूटर में किया गया था। इनके प्रोग्राम को पहले मशीन भाषा में बदला जाता था। जिससे इनकी गति मशीन भाषा से कम होती है। Code :- HLT, ADD , CLA , SUB, STA etc..

High Level Language

UPCISS

High Level Language:- Assembly Language की कमीयों को दूर करने के लिये हाई लेविल भाषा का विकास किया गया । इसमें mnemonic code के स्थान पर कम्प्यूटर को English भाषा में निर्देश दिये जाने लगे, जिस से प्रोग्राम को समझना एवं लिखना आसान हो गया है। यह भाषा मषीन पर आधारित नही होती है। इसमें प्रोग्राम को मषीन भाषा में बदलने के लिये compiler and Interpreter का प्रयोग किया जाता है। आजकल कम्प्यूटर में प्रोग्रामिंग करने के लिये इसी भाषा का प्रयोग किया जा रहा है। इस भाषा को कार्य के आधार पर दो वर्गो में बांटा गया है।

1. Procedure-oriented Programming.
2. Object-oriented programming.

Procedure-oriented Programming

UPCISS

POP :- POP में किसी programming problem के हल को क्रमबद्ध कार्य करने जैसे– reading, processing और writing के रूप में देखा जाता है और इन कार्यो को पूरा करने के लिए बहुत सारे functions लिखे जाते हैं। POP की एक बड़ी कमी यह है कि यह समस्या से संबंधित entities का model तैयार नहीं कर पाता है। POP की दूसरी गंभीर समस्या यह है कि इसमें कई सारे data items को global declare किया जाता है, इससे हमारा डेटा असुरक्षित हो जाता है क्यों कि इसे किसी भी functions के द्वारा अनावश्यक रूप से परिवर्तित किया जा सकता है।

Example - C, FORTRAN, VB, Pascal.

Object-oriented Programming

UPCISS

OOP :- OOP में समस्या को हल करने के लिए पहले हम समस्या से संबंधित entities का model तैयार करते है जिन्हें object कहते हैं। object, संबंधित data व functins के समूह होते हैं। वास्तविक दुनिया में प्रत्येक entity के कुछ न कुछ data व functions होते हैं। एक object के अंतर्गत आने वाले data को केवल उसी object के functions के द्वारा ही access किया जा सकता है, किन्तु एक object के अंतर्गत आने वाले functions को दूसरे object के functions के द्वारा भी access किया जा सकता है। इस प्रकार OOP में data बाहरी object से छिपा हुआ व सुरक्षित रहता है।

Example - C++, JAVA, PYTHON, VB.NET, C#.NET.

Difference Between OOP and POP

UPCISS

POP	OOP
इसमें बड़े प्रोग्राम छोटे–छोटे भागों में विभाजित रहते हैं जिन्हें function कहा जाता है।	इसमें बड़े प्रोग्राम छोटे–छोटे भागों में विभाजित रहते हैं जिन्हें object कहा जाता है।
प्रोग्राम बनाते समय function पर ज्यादा ध्यान दिया जाता है।	प्रोग्राम बनाते समय data पर ज्यादा ध्यान दिया जाता है।
ज्यादातर function global data का प्रयोग करते हैं।	data structure ही objects की विशेषता होती है।
कोई भी function data को प्राप्त करके परिवर्तित कर सकता है।	केवल class के function ही data को प्राप्त करके परिवर्तित कर सकते हैं।
प्रोग्राम डिजाईन में top down approach का प्रयोग किया जाता है।	प्रोग्राम डिजाईन में bottom up approach का प्रयोग किया जाता है।

Translator

UPCISS

1- Assembler 2- Compiler 3- Interpreter

Assembler :- Assembler , एक System Software है। जो assembly Language Program को Machine Language में बदलता है। और Machine Language को Assembly language में बदलता है।

Assembly Language → Assembler → Machine -01

Source Code   Translator   Object Code

Compiler & Interpreter

UPCISS

Compiler और Interpreter दोनो ही system Software है, जो High level language program को Machine level Language में Translator करने का काम करते हैं। compiler पूरे program को एक साथ Translate करता है, जब कि Interpreter Line by Line Translate करता है।

HLL Code	→	C & I	→	Machine -01
Source Code		Translator		Object Code

Source code & Object code

UPCISS

Programmer
Source Code

```c
#include <stdio.h>

int main()
{
    int a,b,s;
    printf("Enter a Two number ");
    scanf("%d %d",&a,&b);
    s=a+b;
    printf("Sum of = %d",s);
}
```

Compiler

Machine
Object Code

```
101101110111000
100011101110010
101101001001001
101101110111000
100011101110010
101101001001001
101101110111000
100011101110010
101101001001001
```

Some High Level Language

- Java
- Python
- HTML
- JavaScript
- C#
- C++
- PHP
- Ruby
- Perl
- C - Some consider it a “mid-level” language

UPCISS

Share

Subscribe

Thank you for watching

YouTube UPCISS Subscribe