

INTRODUCTION TO 4GL

4GL:

Here 4GL is stands for fourth generation language.

Some of the examples are VB, SQL etc.

4GL is a grouping of programming languages which are invented to get closer to human language than 3GLs.

Now the question is how to 4GL is closer to human language?

So we will get answer from SQL query. We have to find out the number of students whose names are Jayesh in a table.

Select * form student where name = 'Jayesh';

Its same as humans speaks.

But the question is what is 3GL?

So 3GL is high level languages like JAVA, C++.

Other than 4GL and 3GL there is 1GL,2GL,5GL too.

1GL, include machine language.

2GL, includes assembly language.

5GL, includes neural networks and artificial intelligence.

MCQs on 4GL

Q1. Which of the following 4GLs invented at IBM and subsequently adopted by ANSI and ISO as the standard language for managing structured data?

- a) SQL
- b) PROLOG
- c) C

Q2. Productivity of software engineers is reduced in using a 4GT.

- a) True
- b) False

Q3. 4GL is an example of _____ processing.

- a) Black Box
- b) Functional
- c) Both Black Box & Functional

Q4. Identify a fourth generation language(4GL) from the given below.

- a) FORTRAN
- b) COBOL
- c) Unix shell

Q5. Which 4GL is used for Data manipulation, analysis, and reporting languages

- a) Lisp
- b) SAS
- c) Unix shell
- d) MATLAB

Q6. Which of the following 4GLs is low code/no code development platform

- a) FORTRAN
- b) New Horizon
- c) Progress

Q7. 4GL used for mathematical optimization

- a) MATLAB
- b) Unify Vision
- c) Visual DataFlex

Q8. GUI creator 4GL

- a) Visual DataFlex
- b) LIRC
- c) R

MCQs Answers

Q1. (a)

Q2. (b)

Q3. (c)

Q4. (c)

Q5. (b)

Q6. (c)

Q7. (a)

Q8. (a)

References:

1. Sebesta, "Concept of programming Language", Pearson Edu
2. Louden, "Programming Languages: Principles & Practices" , Cengage Learning
3. Tucker, "Programming Languages: Principles and paradigms " , Tata McGraw -Hill.
4. E Horowitz, "Programming Languages", 2nd Edition, Addison Wesley

Related posts:

1. Sequence Control & Expression | PPL
2. PPL:Named Constants
3. Parse Tree | PPL | Prof. Jayesh Umre
4. Basic elements of Prolog
5. Loops | PPL | Prof. Jayesh Umre
6. Subprograms Parameter passing methods | PPL | Prof. Jayesh Umre
7. Programming Paradigms | PPL | Prof. Jayesh Umre
8. Subprograms Introduction | PPL | Prof. Jayesh Umre
9. Phases of Compiler | PPL | Prof. Jayesh Umre
10. Parse Tree | PPL
11. Influences on Language design | PPL | Prof. Jayesh Umre
12. Fundamentals of Subprograms | PPL | Prof. Jayesh Umre

13. Programming Paradigm
14. Influences on Language Design
15. Language Evaluation Criteria
16. OOP in C++ | PPL
17. OOP in C# | PPL
18. OOP in Java | PPL
19. PPL: Abstraction & Encapsulation
20. PPL: Semaphores
21. PPL: Variable Initialization
22. PPL: Conditional Statements
23. PPL: Array
24. PPL: Strong Typing
25. PPL: Coroutines
26. PPL: Exception Handler in C++
27. PPL: OOP in PHP
28. PPL: Character Data Type
29. PPL: Exceptions
30. PPL: Heap based storage management
31. PPL: Primitive Data Type
32. PPL: Data types
33. Programming Environments | PPL
34. Virtual Machine | PPL
35. PPL: Local referencing environments
36. Generic Subprograms
37. Local referencing environments | PPL | Prof. Jayesh Umre
38. Generic Subprograms | PPL | Prof. Jayesh Umre
39. PPL: Java Threads

40. PPL: Loops
41. PPL: Exception Handling
42. PPL: C# Threads
43. Pointer & Reference Type | PPL
44. Scope and lifetime of variable
45. Design issues for functions
46. Parameter passing methods
47. Fundamentals of sub-programs
48. Subprograms
49. Design issues of subprogram
50. Garbage Collection
51. Issues in Language Translation
52. PPL Previous years solved papers
53. Type Checking | PPL | Prof. Jayesh Umre
54. PPL RGPV May 2018 solved paper discussion| Prof. Jayesh Umre
55. PPL Viva Voce
56. PPL RGPV June 2017 Solved paper | Prof. Jayesh Umre
57. Concurrency
58. Basic elements of Prolog
59. Introduction and overview of Logic programming
60. Application of Logic programming
61. PPL: Influences on Language Design
62. Language Evaluation Criteria PPL
63. PPL: Sequence Control & Expression
64. PPL: Programming Environments
65. PPL: Virtual Machine
66. PPL: Programming Paradigm

- 67. PPL: Pointer & Reference Type
- 68. try-catch block in C++