

## LOOPS

### MCQs on Loops

Q1. Depending on the place of control statement in loop, it can be classified as what?

- (A)Entry-controlled
- (B)Exit-controlled
- (C)Both (A) and (B)

Q2. Which of the following options show the example of entry controlled loop?

- (A)For loop
- (B)While loop
- (C)Both (A) and (B)

Q3. While loop is example of which loop?

- (A)Entry-controlled loop
- (B)Exit-controlled loop
- (C)Process-controlled loop

Q4. For loop is example of what?

- (A)Exit-controlled loop
- (B)Entry-controlled loop
- (C)Processing controlled loop

Q5. Do-while loop is which kind of loop?

- (A)Entry-controlled
- (B)Exit-controlled
- (C)Entrance-controlled loop
- (D)Both (A) and (B)

Q6. Under which of the following condition, while loop is to be used?

- (A)When the number of iteration cannot be pre-determined
- (B)When loop terminating condition is to be tested before entering the loop
- (C)None of these

Q7. What is done, when break statement comes in loop?

- (A)To terminate loop instantly
- (B)Through program control executes the next statement following the loop
- (C)Repetition of loop

### MCQs Answers

- Q1. (C)
- Q2. (C)
- Q3. (A)
- Q4. (B)
- Q5. (B)
- Q6. (C)
- Q7. (B)

### 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: Introduction to 4GL
22. PPL: Variable Initialization
23. PPL: Conditional Statements
24. PPL: Array
25. PPL: Strong Typing
26. PPL: Coroutines
27. PPL: Exception Handler in C++
28. PPL: OOP in PHP
29. PPL: Character Data Type
30. PPL: Exceptions
31. PPL: Heap based storage management
32. PPL: Primitive Data Type
33. PPL: Data types
34. Programming Environments | PPL
35. Virtual Machine | PPL
36. PPL: Local referencing environments
37. Generic Subprograms

38. Local referencing environments | PPL | Prof. Jayesh Umre
39. Generic Subprograms | PPL | Prof. Jayesh Umre
40. PPL: Java Threads
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++