

Perl Interview Questions and Answers (1–50)

1. What is Perl?

Answer: Perl (Practical Extraction and Report Language) is a high-level, general-purpose programming language known for its text processing capabilities.

2. What are the key features of Perl?

Answer: Text manipulation, regular expressions, portability, dynamic typing, CPAN library support.

3. What are scalars in Perl?

Answer: Scalars are single data items. Prefix: \$. Example: `$name = "Harsh";`

4. What are arrays in Perl?

Answer: Ordered lists of scalars. Prefix: @. Example: `@colors = ("red", "blue");`

5. What are hashes in Perl?

Answer: Unordered key-value pairs. Prefix: %. Example: `%ages = ("John" => 25);`

6. How do you declare variables in Perl?

Answer: Using `my` keyword: `my $name = "Harsh";`

7. What is the use of `chomp()`?

Answer: Removes the newline character from the end of strings.

8. What is the difference between `chop()` and `chomp()`?

Answer:

- `chop()` removes the last character
- `chomp()` removes the newline character only

9. What is a subroutine in Perl?

Answer: A reusable block of code defined using `sub` keyword.

10. How do you pass parameters to a subroutine?

Answer: Using the special array `@_`. Example: `$_[0], $_[1]`

11. What are context types in Perl?

Answer:

- **Scalar context** returns a single value
- **List context** returns a list of values

12. What is a regular expression in Perl?

Answer: A pattern used to match strings. Syntax: `/pattern/`

13. What is the use of `=~` operator?

Answer: Tests if a string matches a regex pattern.

14. What is the use of `!~` operator?

Answer: Tests if a string does **not** match a regex pattern.

15. How do you open a file in Perl?

Answer: `open(my $fh, "<", "file.txt") or die "Can't open file";`

16. How do you read a file line by line?

Answer:

```
while (my $line = <$fh>) {  
    print $line;  
}
```

17. How do you write to a file in Perl?

Answer:

```
open(my $fh, ">", "output.txt");  
print $fh "Hello";
```

18. What is CPAN?

Answer: Comprehensive Perl Archive Network – a large collection of Perl modules.

19. What is `undef` in Perl?

Answer: Indicates an undefined value.

20. What are Perl data types?

Answer: Scalar, Array, Hash, Reference.

21. What is a reference in Perl?

Answer: A scalar that holds the location of another variable.

22. How do you create a reference?

Answer: `$ref = \@array;` or `$ref = \%hash;`

23. How do you dereference a reference?

Answer: Using `@{$ref}` or `%{$ref}` depending on the type.

24. What are special variables in Perl?

Answer: Predefined variables like `$_`, `$!`, `$@`, `$.`, etc.

25. What is `$_` in Perl?

Answer: The default variable for loops and regex operations.

26. What is the difference between `use` and `require`?

Answer:

- `use` is evaluated at compile time
- `require` is evaluated at runtime

27. What is `eval` in Perl?

Answer: Used to execute Perl code stored in a string or to catch errors.

28. What is `die` in Perl?

Answer: Prints a message and exits the program if a condition fails.

29. What is a module in Perl?

Answer: A reusable package of Perl code with `.pm` extension.

30. How do you include a module in Perl?

Answer: Using `use ModuleName;`

31. What is object-oriented Perl?

Answer: Perl supports OO with `bless`, `package`, and methods.

32. What is the `bless` function?

Answer: Associates an object with a class.

33. What is inheritance in Perl?

Answer: Achieved using `@ISA` array to define parent classes.

34. What is AUTOLOAD in Perl?

Answer: A method called when an undefined subroutine is invoked.

35. What is `tie` in Perl?

Answer: Binds a variable to a package class that defines customized behavior.

36. How do you handle errors in Perl?

Answer: Using `eval { ... }` and checking `$@` for error messages.

37. What is strict mode in Perl?

Answer: Forces declaration of variables using `use strict;`

38. What is warnings pragma in Perl?

Answer: Helps catch possible issues with `use warnings;`

39. What is the difference between `print` and `say`?

Answer: `say` adds a newline automatically (Perl 5.10+)

40. How do you sort an array in Perl?

Answer: `@sorted = sort @array;`

41. How do you reverse an array?

Answer: `@rev = reverse @array;`

42. How do you find the length of a string?

Answer: `length($str)`

43. How do you split a string into an array?

Answer: `@arr = split(/,/, $string);`

44. How do you join an array into a string?

Answer: `join(",", @array);`

45. What is the difference between `my`, `our`, and `local`?

Answer:

- `my`: lexical scope
- `our`: global variable within a package
- `local`: temporarily changes global value

46. What are Perl one-liners?

Answer: Short programs executed from the command line using `-e`.

47. What is the `map` function?

Answer: Applies a block of code to each element of a list and returns the modified list.

48. What is the `grep` function?

Answer: Filters elements of a list based on a condition.

49. What is `$_` used for in `map` and `grep`?

Answer: It holds the current element being processed.

50. How do you comment in Perl?

Answer: Using `#` for single-line comments.