Tips and Tricks to crack Technical Interview

Dr. Rajesh Kumar Panda

Kareer School

KIIT Deemed to be University

KIIT UNIVERSITY
Established U/S 3 of UGC Act
Bhubaneswar, Orissa, India

Criteria for Selection in Technical PI

- Attitude
- Confidence
- Communication
- Ability to learn
- Problem Solving Ability
- Critical Thinking Ability
- Projects
- Domain Knowledge
- Skill

Technical PI

- Question may be asked from following areas.
  - Fundamentals
    - CS/IT : C,DS,C++,Java, OS, Networking,RDBMS, SE, OOPS etc.
  - Programs/Codings
  - Domain
  - Project/Training/Internship
  - Puzzles
  - Latest Technology

10 tips that will help you face and successfully crack Technical PI

1. Know The Company

This goes without saying! This is what any employer expects from you and they aren't not wrong! Research well about the company you are applying for and know their aim, mission, vision, values and functions. Using this information in your answers will create the right impression in front of your employer.

2. The Introduction

‘Tell me about yourself’ is every employers favorite question

- **Be confident and enthusiastic:** Stand tall, make eye contact, and greet the interviewers with a smile. Show enthusiasm for the opportunity to be there.
- **Start with the basics:** Begin by stating your full name, and if applicable, mention the position you are applying for.
- **Provide a brief background:** Give a concise summary of your education and relevant Internship/Training experience. Mention any degrees or certifications that are relevant to the Internship/Training.

- **Highlight your expertise:** Emphasize your technical skills and expertise, focusing on the areas that are most relevant to the position you are applying for.
- **Showcase your achievements:** Mention any notable projects or accomplishments that demonstrate your abilities and problem-solving skills.
- **Connect with the company:** Briefly explain why you are interested in working for their company specifically, mentioning any particular aspects that appeal to you.
- **Keep it concise:** Aim for a self-introduction that lasts around 1-2 minutes. Be mindful not to go into too much detail, as you will have the opportunity to expand on your experiences later in the interview.

3. What Can You Offer?

Thoroughly know the job title you are applying for, the responsibilities and why you are fit for the job. For every point that you make, try to have an instance where you have used that quality.

4. Act Confident

This can be quite a task considering it’s the first time you are sitting for an interview but you have to pull off this lie. You have to act confident and make the employer feel that you have it all together and perfectly fit the job.

5. Never Lie In Your Resume

Lying can backfire! The person sitting in front of you has ample experience of interviews and can easily know when you are faking it. So keep it real and only write in your CV, what you have actually achieved.

6. Dress Well

No one can stress enough on the fact that in case of an interview, ‘The first impression is the last impression.’ Put on your best clothes (formal of course!) and look neat when going for an interview for the first time or otherwise.

7. Clarity in Communication

There should be clarity in your communication. You should be a good listener first.

8. Be Eager To Learn

It’s okay to not know everything! If you get stuck on some questions and don’t know what to say, politely let the interviewer know. But also make it a point to ask what the answer was once they are done with all the questions. This shows your eagerness to learn, which is often considered useful from an employer point of view.

9. Breathe

More than anything else, it’s the anxiety that ruins most interviews. Just before entering the room, take a few deep breaths and relax your nerves. It always helps!

10. Thank You in the end

Sample Interview Questions

Project

- Summary of the Project
- Why this project?
- Your contribution/role in the project.
- What tools or software have you used ? And why have you used them ?
- Explain by drawing a block diagram.
- Explain the project.
- What you learnt from this project?
- Which labs/Companies did you visit for this project?
- Which books/website did you follow?

- What are the results or conclusions?
- How this project can benefit our company or Society?
- What are the limitations of your project?
- What was the systematic approach that you followed in your project?
- What were the problems faced in completion of this project?
- Software model used in your project.
- Testing Methodology.
- Questions related to technology used.

Sample Interview Questions

C++

- Differences between C and C++.
- Differences bewteen POP and OOP.
- What is wchar_t in C++?
- What is name mangling?
- What is VTABLE?
- What is VPTR?
- What is RTTI?

- What is reinterpret cast?
- What is Composition?
- What is object slicing?
- What is upcasting?
- Can pure virtual function have a body?
- What is the use of explicit keyword?
- What is inline function?
- Difference between inline function and macro.

- Why we can not have a virtual constructor but we can have a virtual destructor?
- What is cout an cin?
- What is persistence?
- What is genericity?
- Which operator has the highest priority in C++?
- Which operators can not be overloaded?

Difference Between POP and OOP

Procedural Oriented Programming	Object Oriented Programming
• In procedural programming, program is divided into small parts called ***functions***.	• In object oriented programming, program is divided into small parts called ***objects***.
• Procedural programming follows ***top down approach***.	• Object oriented programming follows ***bottom up approach***.
• There is no access specifier in procedural programming.	• Object oriented programming have access specifiers like private, public, protected etc.
• Adding new data and function is not easy.	• Adding new data and function is easy.
• Procedural programming does not have any proper way for hiding data so it is ***less secure***.	• Object oriented programming provides data hiding so it is ***more secure***.
• In procedural programming, overloading is not possible.	• Overloading is possible in object oriented programming.
• In procedural programming, function is more important than data.	• In object oriented programming, data is more important than function.
• Procedural programming is based on ***unreal world***.	• Object oriented programming is based on ***real world***.
Examples: C, FORTRAN, Pascal, Basic etc.	Examples: C++, Java, Python, C# etc.

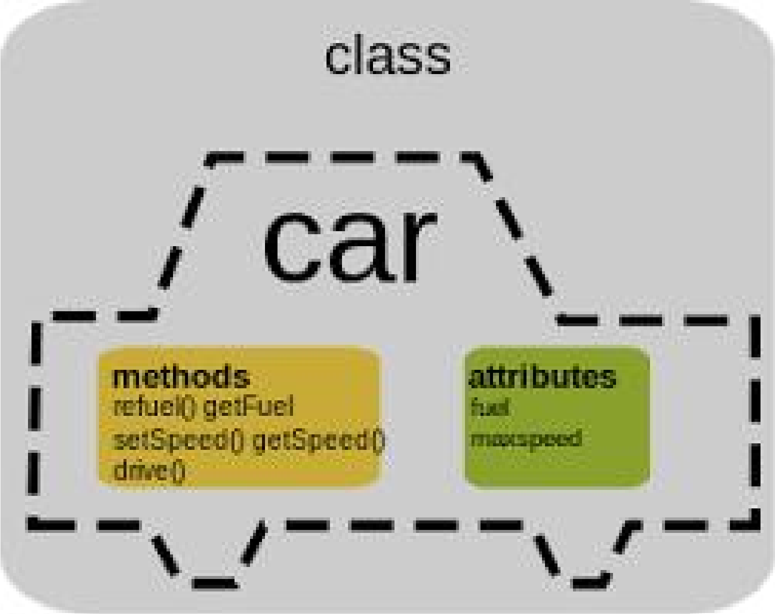
Object

- In general object is a real world entity which has properties and behaviours.
- Object is a run time entity of an object oriented system.
- Object is a variable of class type.
- Object is partitioned area of memory containing data and function together.
- Object is an Instance of a Class.

Class

- Class is a collection of similar types of objects.
- Class is an user-defined data type, which holds its own data members and member functions.
- Class is like a blueprint for an object.
- Common properties and behaviours of similar types of objects are define inside a Class.

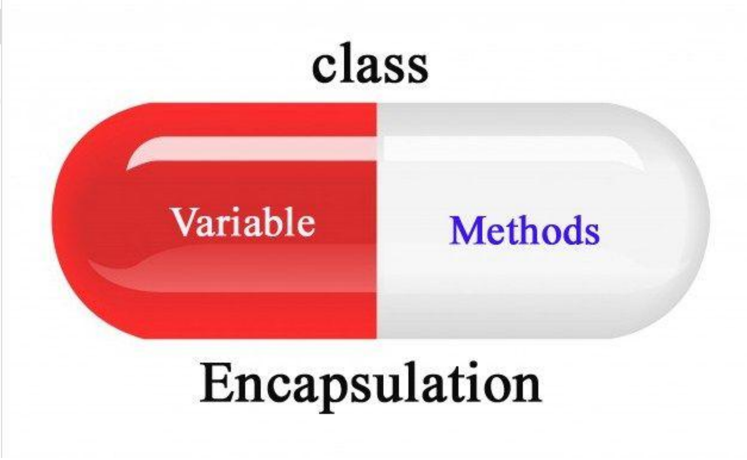
Class & Object



Objects

Encapsulation

- Wrapping up of data and function together.
- Mechanism of combining data and function into a single unit so that data can be accessed by the function inside the unit.
- Encapsulation also lead to data hiding.



Data Hiding

- Due to encapsulation data can only be accessed by the function inside the class. This insulation of data from being accessed by outside function is known as Data Hiding.
- Anything declared as private can be accessed through the member functions.

Data Abstraction

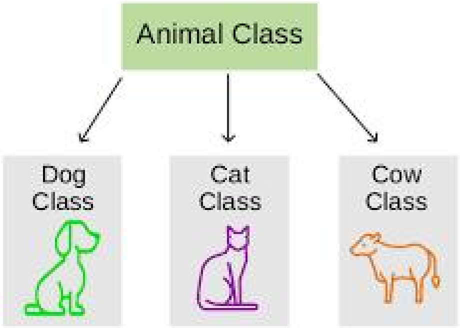
- Abstraction refers to implementing something without including the details about it.
- Class use this concept by abstracting some or all properties and behaviours of an Object so the Class is also known as Abstract Data Type(ADT).
- It hides unnecessary properties

  and behaviours of Objects.

I don't know how my phone works internally but I know **how to make calls**
STM

Inheritance

- It is the ability to derive new classes from existing classes
- New class is known as child/derived class and old class is known as parent/base class.
- Derived class can inherit properties and behaviours of base class.
- It is used to represent IS_A or KIND_OF relationship between classes.
- Main benefit of Inheritance is reusability.



Polymorphism

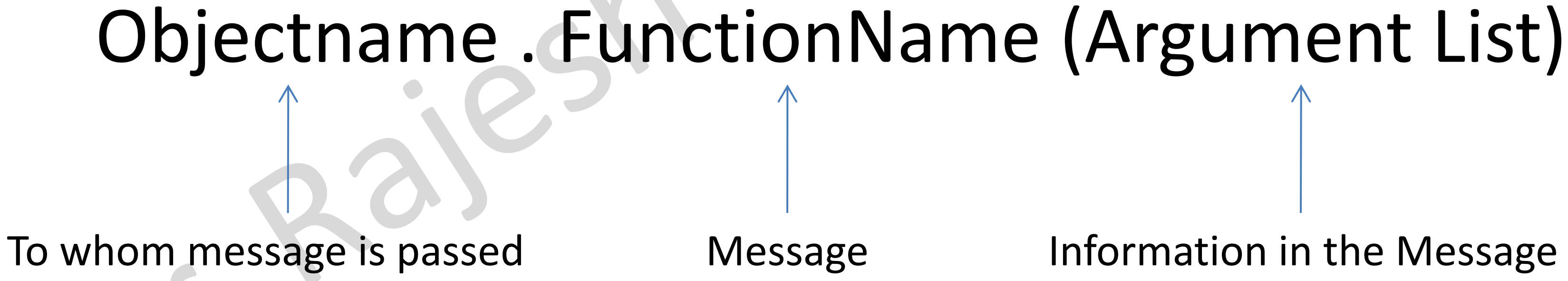
- Polymorphism is the ability to take more than one form.
- We can define more than one function with same name called function overloading. One operator can be redefined to exhibit different task on different data types called operator overloading. These are different forms of Polymorphism.
- Real life example of polymorphism, a person at the same time can have different characteristic. Like a man at the same time is a father, a husband, an employee.

Dynamic Binding

- Binding refers to linking of function call with function definition.
- Linking of function call with function definition at compile time is known as Static Binding or early binding.
- Linking of function call with function definition at runtime is known as Dynamic Binding or late binding.
- Dynamic binding uses Objects to resolve binding at Runtime.

Message Passing

- Two Objects can communicate with each other through functions that is called message passing.



Objectname . FunctionName (Argument List)

To whom message is passed

Message

Information in the Message

Genericity

- Genericity is the generic programming approach in which we can define classes and methods those can take data types as arguments and can work for variety of data types.
- It supports reusability of code.
- Example: Templates in C++ , Generic Classes and Methods in JAVA

Delegation/Containership

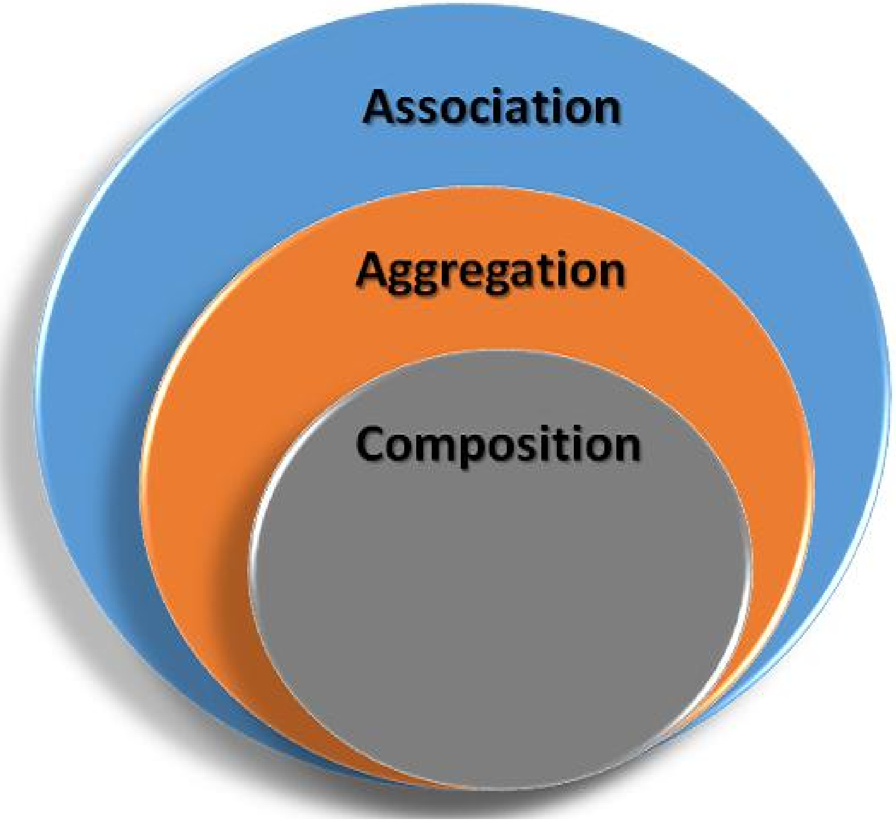
- One class can have objects of other classes as its member.
- This is another way of accessing properties and behaviour of other objects.
- It is used to establish HAS_A relationship between classes.
- Delegation is the passing of some tasks to other objects.
- Example: Car has Wheels, Engine.

Persistence

- Persistence is the ability to store the current state of an object in secondary memory or files.
- We can store the objects permanently in files or database.

Association

Association is a relationship between two separate classes and the relationship can be one to one, One to many, many to one and many to many. It joins two entirely separate entities.



ASSOCIATION

- Association is a relationship between two objects.
- Objects might not be completely dependent on each other.
- One-to-many, many-to-one, many-to-many all these words define an association between objects
- *Example:* A Student and a Faculty are having an association.

Aggregation

- Aggregation is a special form of association which is a unidirectional one way relationship between classes (or entities).

  Example: Wallet and Money classes.

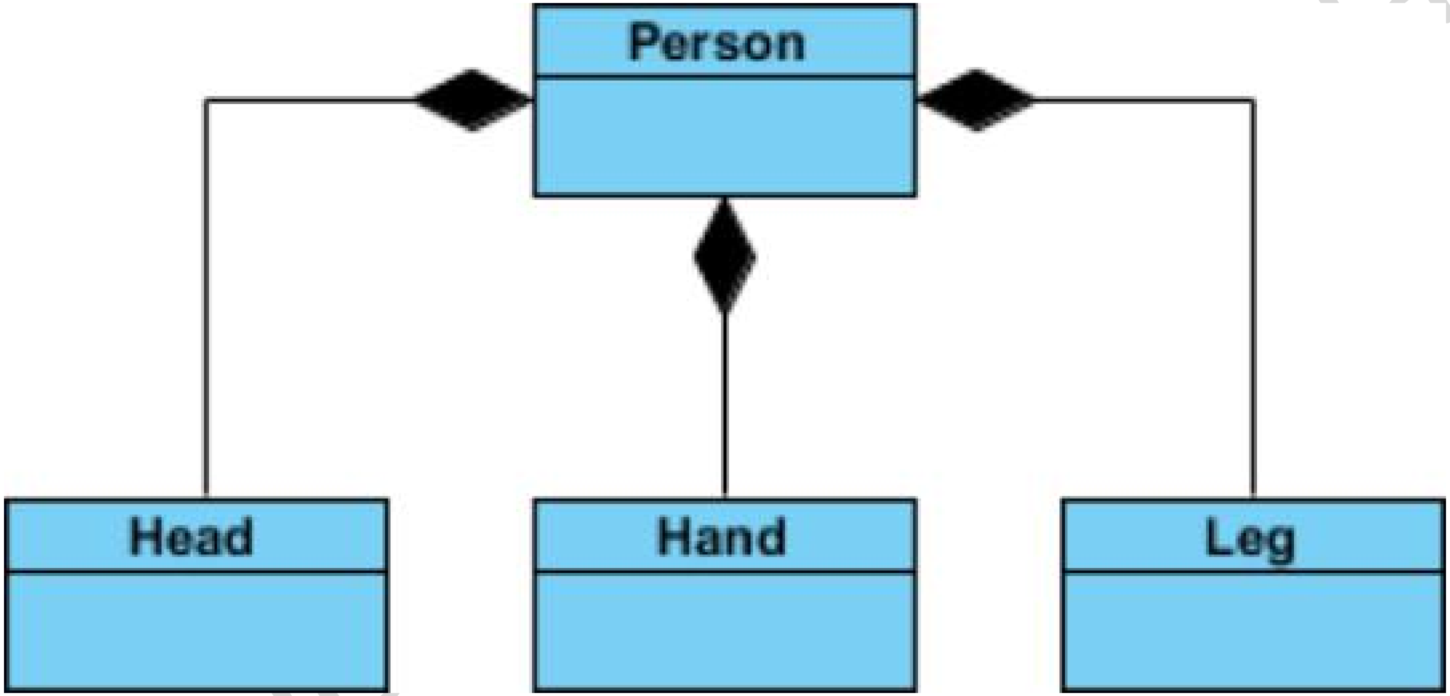
  Wallet has Money but money doesn’t need to have Wallet necessarily so its a one directional relationship. In this relationship both the entries can survive if other one ends. In our example if Wallet class is not present, it does not mean that the Money class cannot exist.

Composition

- **Composition** is a restricted form of Aggregation in which two entities (or you can say classes) are highly dependent on each other.

  Example: Human and Heart.

  A human needs heart to live and a heart needs a Human body to survive. In other words when the classes (entities) are dependent on each other and their life span are same (if one dies then another one too) then its a composition. Heart class has no sense if Human class is not present.

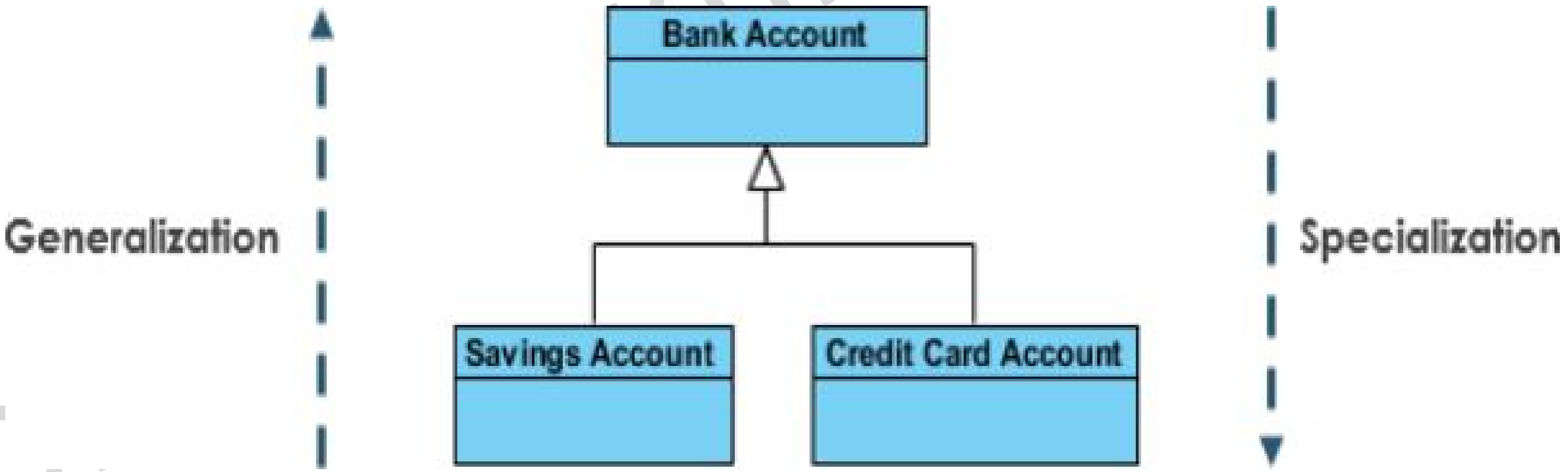

Person
Head
Hand
Leg

Generalization vs Specialization

- **Generalization** is a mechanism for combining similar classes of objects into a single, more general class. Generalization identifies commonalities among a set of entities. The commonality may be of attributes, behavior, or both. In other words, a superclass has the most general attributes, operations, and relationships that may be shared with subclasses. A subclass may have more specialized attributes and operations.
- **Specialization** is the reverse process of Generalization means creating new sub-classes from an existing class.

Example

- a Bank Account is of two types - Savings Account and Credit Card Account. Savings Account and Credit Card Account inherit the common/ generalized properties like Account Number, Account Balance, etc. from a Bank Account and also have their specialized properties like unsettled payment etc.



Sample Interview Questions

C Programming

1. What is the difference between Procedure Oriented & Object Oriented programming
2. What type of language C is?
3. Why the name given C?
4. What are the Application of C Language?
5. What is a keyword ?
6. How many Keywords are there?
7. What is an identifier?

8. What is a variable?

9. What is data type?

10 What is storage class?

11. How many storage class are there?

12. Use of each Storage class

13. What is scope of a variable?

14. What is the max & min value of int?

15. What is void?

16. What is size of ‘x’?

17. Memory representation of integer?
18. What is Lvalue and Rvalue?
19. What are qualifiers in C?
20. What is the length of variable name?
21. What is the syntax of declaring variable?
22. What is the difference between declaration and definition?
23. What is the difference between char variable and a char Constant?
24. What is size of 2.5?

25. What is coercion?

26. What is the size of a character literal in C?

27. What is the use %g?

28. What is ellipsis?

29. How many arguments main() function can take?

30. main() function is predefined or user defined?

31. What header files contain?

32. What is a preprocessor

33. What are the basic task of a preprocessor?

34. When to give “ ” and when to give < > in file inclusion?

35. Where the library functions are defined?

36. What is a macro?

37. What are the advantages and disadvantages of Macro?

38. Use of #pragma

39. Preprocessor operators

40. The keyword which is an operator

41. How many operators are there?

42. Which operator has the highest and which has lowest precedence.

43. Different types of operator

44. How many types of assignment are there?

45. What is %n refers to.

46. What is the prototype of printf.

47. What is the prototype of scanf.

48. What is an expression.

49. What are the difference between an expression and a statement?

50. What is a Dummy operator?

51. How to exchange value between two variables without using third variable without using arithmetic operator?

52. What is a hanging statement?

53. What is loop inversion?

54. What is side effect of a variable?

55. Which char has the ASCII value 0?

56. What is a generic pointer?

57. What is a wild pointer?

58. What is dangling pointer?

59. What is a NULL pointer?

60. How many types of pointer are there?

61. What is the size of a pointer?

62. What is the pointer representation of an element in one dimensional array.

63. What is the pointer representation of an element in two dimensional array.

65. What is the different between far pointer and huge pointer?

66. What is the difference between exit() and _exit() functions?

67. What is the difference between strcpy() and memcpy()?

68. What are the jumping statements in C?

69. What is the max value stored in long int var?

70. Why the array index starts from 0?

71. How many max chars allowed in a char constant.

72. Can array have negative size?

73. Can array index be negative?

74. What is Size of Union Variable?

75. What is a self referential structure?

76. Can a union be self-referenced?

77. What is a dummy function?

78. What is recursion?

79. Which DS is used for recursion?

80. is Array a pointer?

81. What type of pointer Array is?

82. What are the difference between array and structure?

83. What are the difference between structure and union?

84. What is enumeration?

85. Different types of stream in c.

86. what is the difference between getch(), getche() and getchar()?

87. Difference between text file and binary file .

88.Can we have an empty structure in C.

89.______ is used to rename a structure.

90.What is the default storage class for global data.

91.What is the difference between %f and %e?

92. what is the difference between %d and %i?

93. what is the difference between i++; and ++i;?

94. what is the difference between i++ and i=i+1?

95. what is the value of -1>>5?

96. What are the default files a program has access when it begins execution?

97. What are the basic file operations?

98. What printf() returns?

99. What scanf() returns?

100. Is there any bound checking in Array?

101. What is an array?

102. What is call by value and call by address?

103. What do you mean by signature of function?

104.What is actual parameter and formal parameter?

105. What is dynamic memory allocation?

106. What are the difference between malloc() and calloc().

107. Difference between array and pointer.

Sample Interview Questions

Python

- What is __init__?
- What is the difference between Python Arrays and lists?
- What is slicing in Python?
- What is docstring in Python?
- What is break, continue and pass in Python?
- What is the use of self in Python?
- What are lists and tuples? What is the key difference between the two?
- What is Scope in Python?

- What is a dynamically typed language?
- What is lambda in Python? Why is it used?
- What are the membership operators in python?
- What is pickling and unpickling?
- How is memory managed in Python?
- What are Python namespaces? Why are they used?

Sample Interview Questions

JAVA

- What are the three main component of JVM?
- What is a Class Loader?
- What are the different types of Class Loader?
- How many types of memory areas are allocated by JVM?
- What is JNI?
- Why Java is a robust language?
- Why Java is a high performance language?
- What is Reflection in JAVA?

- What are the different types of Inner Classes in JAVA
- Which method invokes garbage collector?
- What is exception propagation?
- Which is the super class of all Exception in JAVA?
- What is String Pool?
- What is serialization in Java?
- What is Deserialization?
- What is the difference between final, finally and finalize?

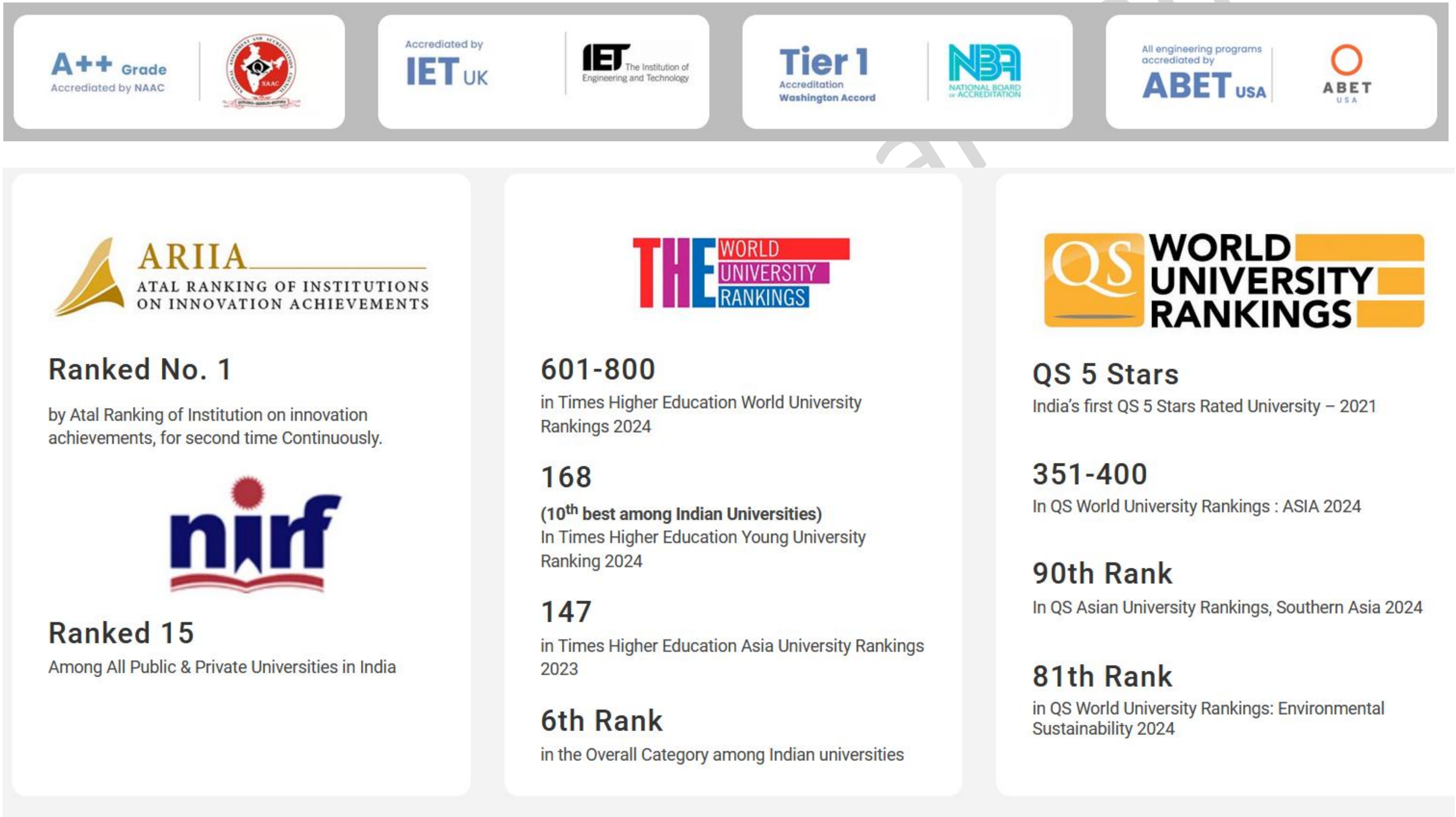
- What is object cloning?
- What is covariant return type?
- What is the final blank variable?
- What is a marker interface?
- What is a Concrete Method in Java?
- What is a default method in java?
- How will you invoke any external process in Java?
- What is the difference between yield() and sleep() method?

- What is the transient keyword?
- What are autoboxing and unboxing?
- What is the purpose of the strictfp keyword?
- What is a singleton class?
- What about the daemon threads?

Thank You

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

Deemed to be University U/S 3 of UGC Act, 1956



A++ Grade
Accredited by NAAC

Accredited by
IET UK

Tier 1
Accreditation
Washington Accord

All engineering programs accrediated by
ABET USA

ARIIA
ATAL RANKING OF INSTITUTIONS ON INNOVATION ACHIEVEMENTS

Ranked No. 1

by Atal Ranking of Institution on innovation achievements, for second time Continuously.

Ranked 15

Among All Public & Private Universities in India

601-800

in Times Higher Education World University Rankings 2024

168

(10th best among Indian Universities)
In Times Higher Education Young University Ranking 2024

147

in Times Higher Education Asia University Rankings 2023

6th Rank

in the Overall Category among Indian universities

QS 5 Stars

India's first QS 5 Stars Rated University – 2021

351-400

In QS World University Rankings : ASIA 2024

90th Rank

In QS Asian University Rankings, Southern Asia 2024

81th Rank

in QS World University Rankings: Environmental Sustainability 2024

IET The Institution of Engineering and Technology
NBA NATIONAL BOARD of ACCREDITATION
ABET USA
THE WORLD UNIVERSITY RANKINGS
QS WORLD UNIVERSITY RANKINGS
nirf