1.Which of the following statements is false for dynamic programming?

a. Given problem is broken up into smaller sub-problems.
b. The optimal solutions to the subproblems contribute to the problem's optimal solution.
c. Can be implemented using a recursive algorithm.
d. Does not guarantee to find optimal solutions for problems

2.Which following statement is false about Relational Database Management System?

a. Data is an unprocessed fact.
b. Information is processed data.
c. A database is a collection of non-related data.
d. DBMS must include concurrency control.

3. What will be the output of the following code?

```c
#include <stdio.h>
int main() {
        int sum = 7 + 6 / 3 + 14 * 2;
        printf("%d", sum);
        return 0;
}
```

a. 37
b. 16
c. 17
d. 33.5

4. If the area of a rectangular region is equal to the area of a square, then the perimeter of the rectangular must be –

a. Half the perimeter of the square
b. Equal to the perimeter of the square
c. Equal to twice the perimeter of the square
d. None of the above

5. In the following code snippet, what does the keyword 'this' refer to

```
class Employee {
    private String name;
    public Employee(String name) {
        this.name = name;
}
}
```

a. The class variable.

b. The constructor variable.

c. The current object of the class.

d. The class itself.

6. In the context of the Software Development Life Cycle (SDLC), which model emphasizes the continuous iteration of the development and testing phases throughout the project, accommodating changes in requirements even late in the development process?

a. Waterfall Model
b. V-Model
c. Spiral Model
d. Agile Model

7. Mr. Joy is planning to build a web browser. Now he is analyzing requirements for the navigation system of his web browser, which will preserve the browsing history. What is the appropriate data structure to use for the navigation system?

a. Array
b. Stack
c. Queue
d. Linked list

1. **Object-Oriented Programming (OOP)**
2. **Database**
3. **Data Structure & Algorithm**
4. **Problem-Solving**
5. **Output Tracing**
6. **Analytical Ability**
7. **Software Development Life Cycle (SDLC)**

Technical (Programming) Multiple Choice Questions (MCQs)

MCQs Online Test – T4Tutorials.com

https://www.glassdoor.com/Interview/dhaka-software-tester-interview-questions-SRCH_IL.0,5_IM1237_KO6,21.htm

- **Waterfall**: Best for clear, stable projects with minimal changes.
- **V-Model**: Good for projects with clear requirements and a strong focus on testing.
- **Agile/Scrum**: Ideal for projects with changing requirements and frequent client interaction.
- **Spiral**: Suitable for high-risk projects with evolving requirements.
- **RAD**: Useful for projects needing rapid development.
- **DevOps**: Best for continuous integration and ongoing support

What do you understand by pure object-oriented language? Why is Java not a pure object-oriented programming language?

The programming language is called a pure object-oriented language that treats everything inside the program as an object. The primitive types are not supported by the pure OOPs language. There are some other features that must satisfy by a pure object-oriented language:

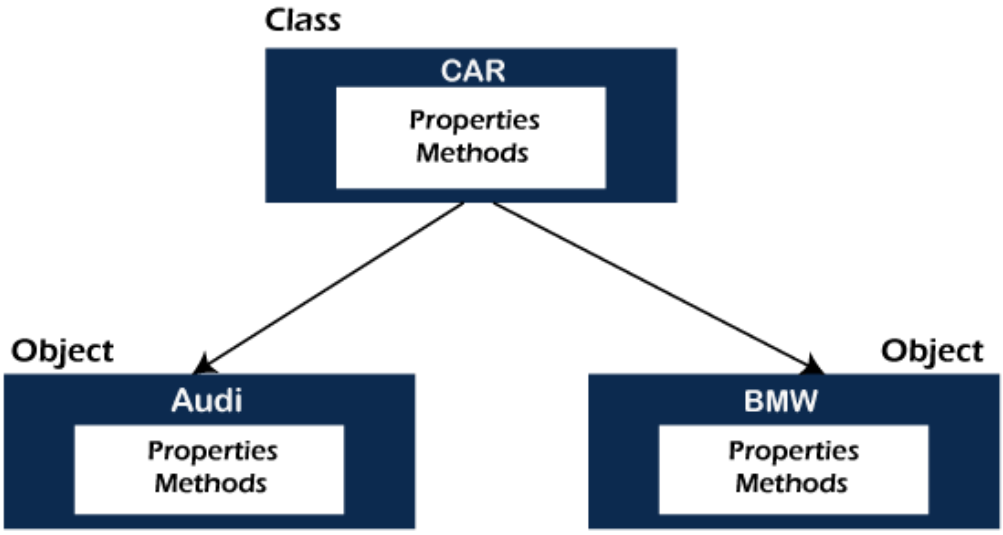
- Encapsulation
- Inheritance
- Polymorphism
- Abstraction
- All predefined types are objects
- All user-defined types are objects
- All operations performed on objects must be only through methods exposed to the objects.

Java is not a pure object-oriented programming language because pre-defined data types in Java are not treated as objects. Hence, it is not an object-oriented language.

What do you understand by class and object? Also, give an example.

Class: A class is a blueprint or template of an object. It is a user-defined data type. Inside a class, we define variables, constants, member functions, and other functionality. It does not consume memory at run time. Note that classes are not considered as a data structure. It is a logical entity. It is the best example of data binding.

Object: An object is a real-world entity that has attributes, behavior, and properties. It is referred to as an instance of the class. It contains member functions, variables that we have defined in the class. It occupies space in the memory. Different objects have different states or attributes, and behaviors.
The following figure best illustrates the class and object.



https://chatgpt.com/share/67862727-9124-800c-ae2f-3d12d56dff9d