1. Office Supplies (Pens & Pencils) • A stationery store sells different types of pens and pencils. Each pen has a brand, color, and ink type, while each pencil has a hardness level (HB, 2B, etc.) and an eraser (Yes/No). • What would be the class, objects, properties, and methods
   • Class: pen,pencil
   • Objects: • Pen1 • Pencil1
   • Properties: • color • inktype • hardness level
   • Method: • write() → Simulates writing with the pen.

2. Library System • A library has thousands of books. Each book has a title, author, and genre. The library allows users to borrow and return books. • Define the class, objects, properties, and methods for this scenario.
   • Class: book
   • Properties: Title Author genre
   • Method: Borrow() Return()
   • Objects: Book1 Book2

3. Car Showroom • A car showroom has different brands of cars. Each car has a model name, color, fuel type, and price. Customers can purchase a car. • What would be the class, objects, properties, and methods?
   • Class: Car
   • properties:Model name color fuel type price
   • Method: Purchase()
   • Objects: car1 car2 4. Online Food Ordering System

4. Online Food Ordering System • A restaurant has an online menu with different food items. Each food item has a name, price, and category (veg/non-veg). Users can place an order. • How would you represent this using classes and objects?

•Class:food item

•properties:name,price and categegeory

•methods:order()

•object:food item1,food item 2

5.Mobile Phone Shop • A mobile shop sells smartphones. Each phone has a brand, model, RAM, and price. Customers can buy a phone and check phone details. • Define the class, objects, and methods.

•class:phone
•object:phone1
•properties:model,ram,price

•method:buy(),check()

6. Movie Ticket Booking System • A movie theatre allows users to book movie tickets. Each movie has a name, duration, and language. Users can book, cancel, or check available seats.

•class:movie ticket

•object:movie

•properties:name,duration,language

•method:book(),cancel(),check()

7. Hospital Management System • A hospital has multiple doctors and patients. Each doctor has a name, specialization, and years of experience, while each patient has a name, age, and disease. • What are the classes, objects, and methods?

•class:hospital

•object:doctor and patient

•properties:name,specialization,year,age,diseases

•method:check(),priscribe()

8. E-Commerce Website • An e-commerce website has multiple products. Each product has a name, price, brand, and category. Users can add items to cart and purchase them. • Identify the classes, objects, properties, and methods.

•class:website

• Object:product1,product2

•properties:name,price,brand,categegory

•method:cart(),purchase()

9. Banking System • A bank manages multiple customer accounts. Each account has an account number, balance, and account type. Customers can deposit, withdraw, and check balance.

•class:banking system

•object:coustomer1,customer2

•properties:number,balance,accout type

•method:deposite(),withdraw(),check balance()

10. Student Management System • A school manages students. Each student has a name, roll number, grade, and subjects. Students can enroll in courses and check their marks.

•class:student management

•object:student1,student2

•properties:name,rollnummber,grade,subject

•method:enroll(),check()

11. Taxi Booking App • A taxi booking app allows users to book a ride. Each car has a driver name, car model, and availability status.

•class:car

•object:car1,car2

•properties: name, car model, and availability status.

•method:ride()

12. Gaming System • A video game has players. Each player has a name, level, and score. Players can earn points and level up.

•class:player

• Object: Player1, Player2

•properties: name, level, and score

•method: earn points(),level up()

13. Hotel Management System • A hotel has multiple rooms. Each room has a room number, type (Single/Deluxe), and price per night. Customers can book, check-in, and check-out.

•class:rooms

•object:room1,room2

•properties: number, type (Single/Deluxe), and price per night

•method: check-in(), and check-out().

14. Weather Forecast System • A weather app provides weather updates. Each location has a temperature, humidity, and forecast. Users can check the weather of a specific location.

•class:location

•Object:location1,location2

•properties: temperature, humidity, and forecast

•method: check()

15. Smart Home Automation • A smart home has multiple devices like lights, fans, and ACs. Each device has a status (On/Off) and power consumption.

•class:home

•object:devise

•properties: lights, fans, and Acs

•method: status (), power consumption().

16. Social Media Platform • A social media app has multiple users. Each user has a name, email, and list of friends. Users can send messages, post updates, and like posts.

•class:user

•object:user1,user2

•properites: name, email, and list of friends

•method: send messages(), post updates(), and like posts().

16. Social Media Platform • A social media app has multiple users. Each user has a name, email, and list of friends. Users can send messages, post updates, and like posts.

•class:user

•object:user1,user2

•properties: name, email, and list of friends

•method: messages(), post updates(), and like posts().

17. Flight Booking System • An airline allows customers to book flights. Each flight has a flight number, destination, and seat availability

•class:flight

•object:customer1,customer2

•properties: flight number, destination, and seat availability

•method:book()

18. Music Player App • A music player has multiple songs. Each song has a title, artist, and duration. Users can play, pause, and skip songs.

•class:app

•object:song1,song2

•properties: title, artist, and duration

•method:play(),pause(),skip()

19. Supermarket Billing System • A supermarket has multiple products. Each product has a name, price, and stock quantity. Customers can buy products and receive a bill.

•class:supermarket

•object:product

•properties: name, price, and stock quantity

•method:buy(),receive()

20. Online Learning Platform • An e-learning platform offers courses. Each course has a title, instructor, and duration. Students can enroll and complete courses.

•class:couse

•object:course1,course2

•properties: title, instructor, and duration

•method:enroll(),complete()