How to approach design problems

1) Types of LLD Interviews
2) How to approach LLD interviews
3) Design a Pen.

Class starts at 9:10 PM

Types of LLD Interviews

Theoretical

1) Old tech companies
- Oracle, Banks

2) Test your Knowledge

eg: explain what is singleton pattern

30-45 mins, 3-4 Ques

Design

1) Problem solving round

2) Often there is a case study / design BMS

3) Generally asked in Big MNCs
→ Microsoft
→ Amazon
→ Walmart

Machine Coding

1) Asked in most of the startups
↳ Flipkart, PhonePe, Myntraa

2) They want does/s

→ no code is required, at max a very high level implementation

→ Paypal

(3) They want to see, whethere you are able to apply your knowledge

~ 45 mins | 1 ques

(4) Write code for few imp. classes at max, - depends on interviewer

(3) They expect people to come up with design + code a working solution

1 ques + follow-ups

Design	Machine Coding
(1) Typically given a single line Ques eg: Design BMS	(2) Entire doc with all the reqs is given, there could be some mandatory reqs and there could also be some optional

② The interviewee is expected to lead the conversation

③ ⓐ gather all the reqs.
ⓑ Class Diagrams
ⓒ Schema Diagrams
ⓓ Write code for some classes

ones

② ⓐ Clarify the doubts/requirements

ⓑ Class Diagrams
ⓒ Schema Diagrams
ⓓ Write end-to-end code
ⓔ Follow up ques. by interviewer

Machine coding

25-30 mins ⟹ Reqs / Class Diagram / Schema Diagrams

generally { 90 mins ⟹ code the solution
interviewer goes on mute / leaves

25-30 mins ⟹ Run the code / follow up ques

At Scaler ⟹ 1 line PS

Gather the reqs
Clarifying Ques
Class Diagram
Schema Diagram
E2E working solution

Deep Dive into each Step

Design a Pen

Before we start with step 1, gather reqs., there is a step 0
⟹ get an overview

⟹ to align your mind with what the interviewer has in mind.

Overview

- You already Know about the system
- You dont Know about

① align yourself with the expectation of the interviewer

② Some questions you should ask what exactly should you be building

- ① Entity ⇒ Bird, Pen
- ② Real world S/W design ⇒ BMS

⇒ ② ⓐ ⇒ whether I should persist the data in a database or whether I should store in memory.

ⓑ whether an input needs to be taken from the user

- → Cmd line
- → Rest API
- → hardcoding

The system

① Perfectly OK to ask clarifying questions

STEP 1 → Gather & Clarify the reqs.

Instead of asking the interviewer directly I can start suggesting the features

⇒ Suggest 5-8 features and check with the interviewer if that is fine.

⇒ for coming up with the features, try to visualize the system

⇒ Jot down all the features

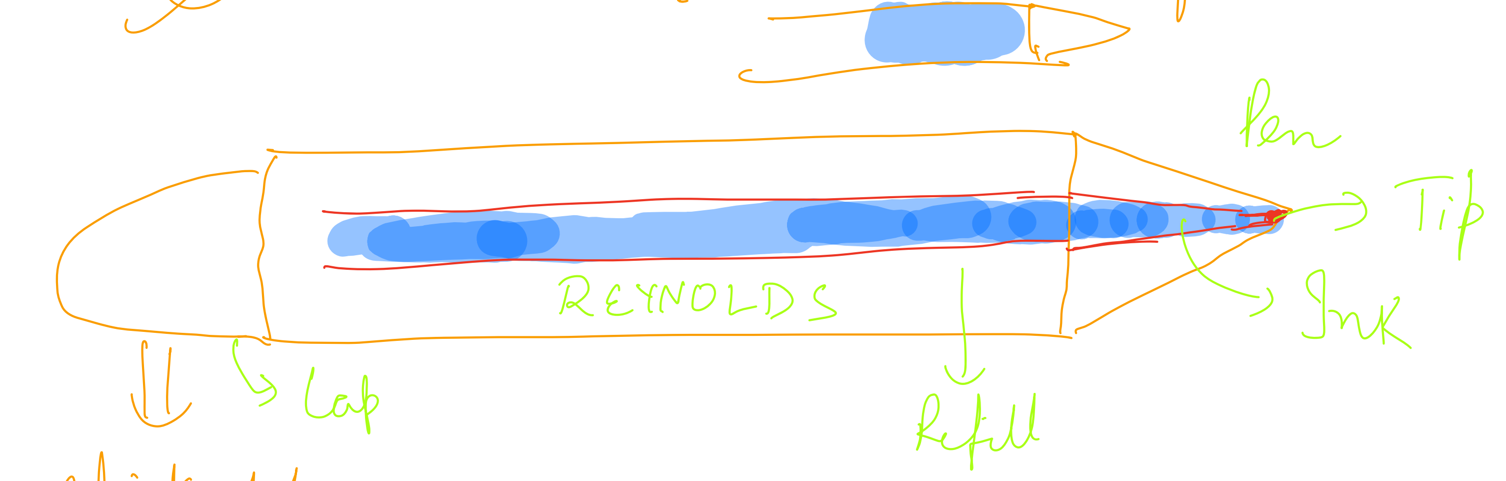
⇒ Think about the edge cases that can be there or might come in future.

⇒ Filling Ink ⇒ Refill the pen

① Gel Pen, Ball Pen, Fountain Pen

② Refillable, or not-refillable

3



clickable
Cap

3. Diff ink have diff. color
4. For every pen, we need to store brand, price, name, weight
5. Pen can be opened via a cap or its a clickable pen,
6. Refill can also have diff. radius for nibs
7. Fountain pen doesnt have any refill

Behaviour

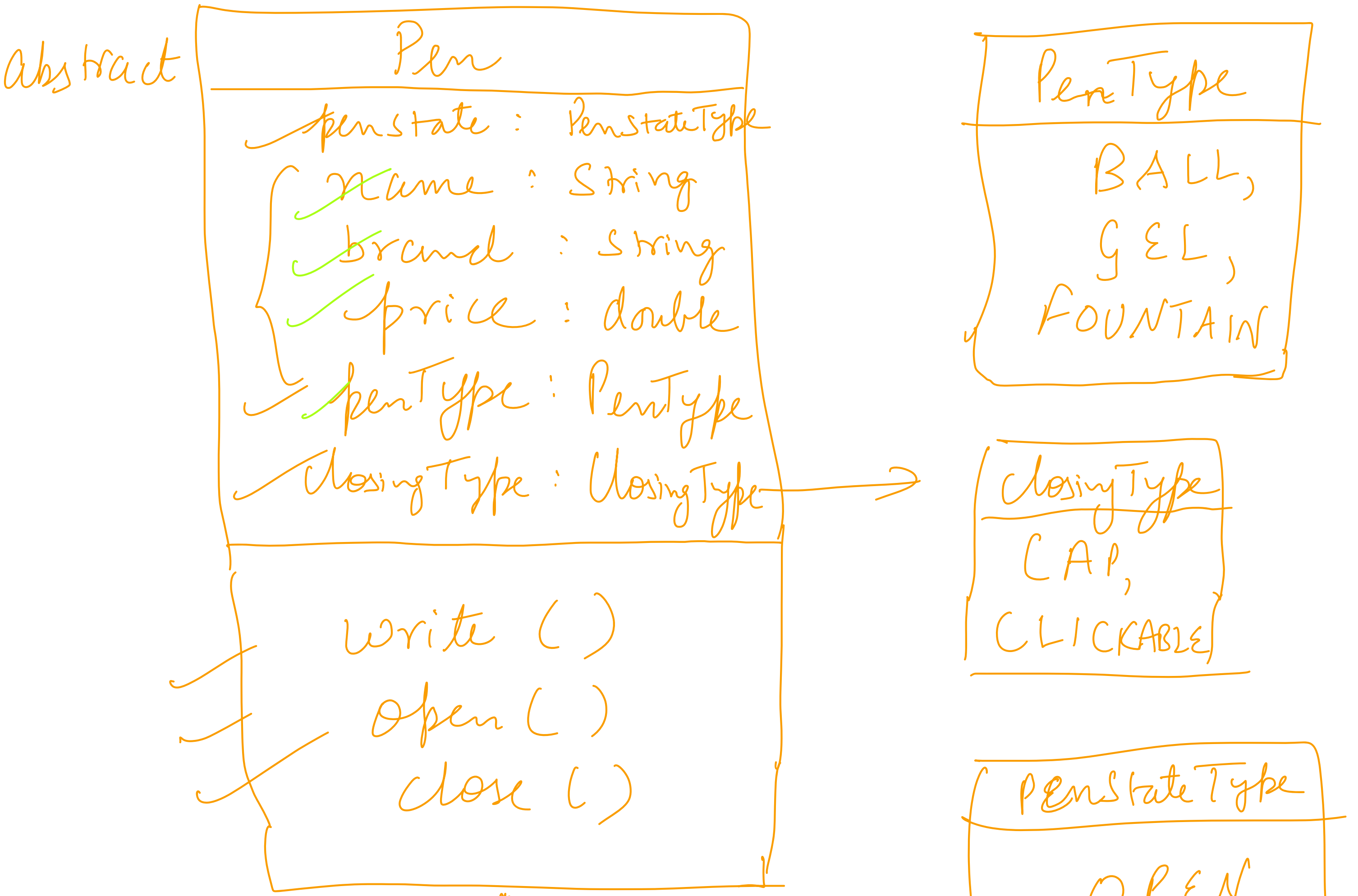
1. Pen should be able to write
2. We should be able to open/close a pen
3. We should be able to refill the pen.

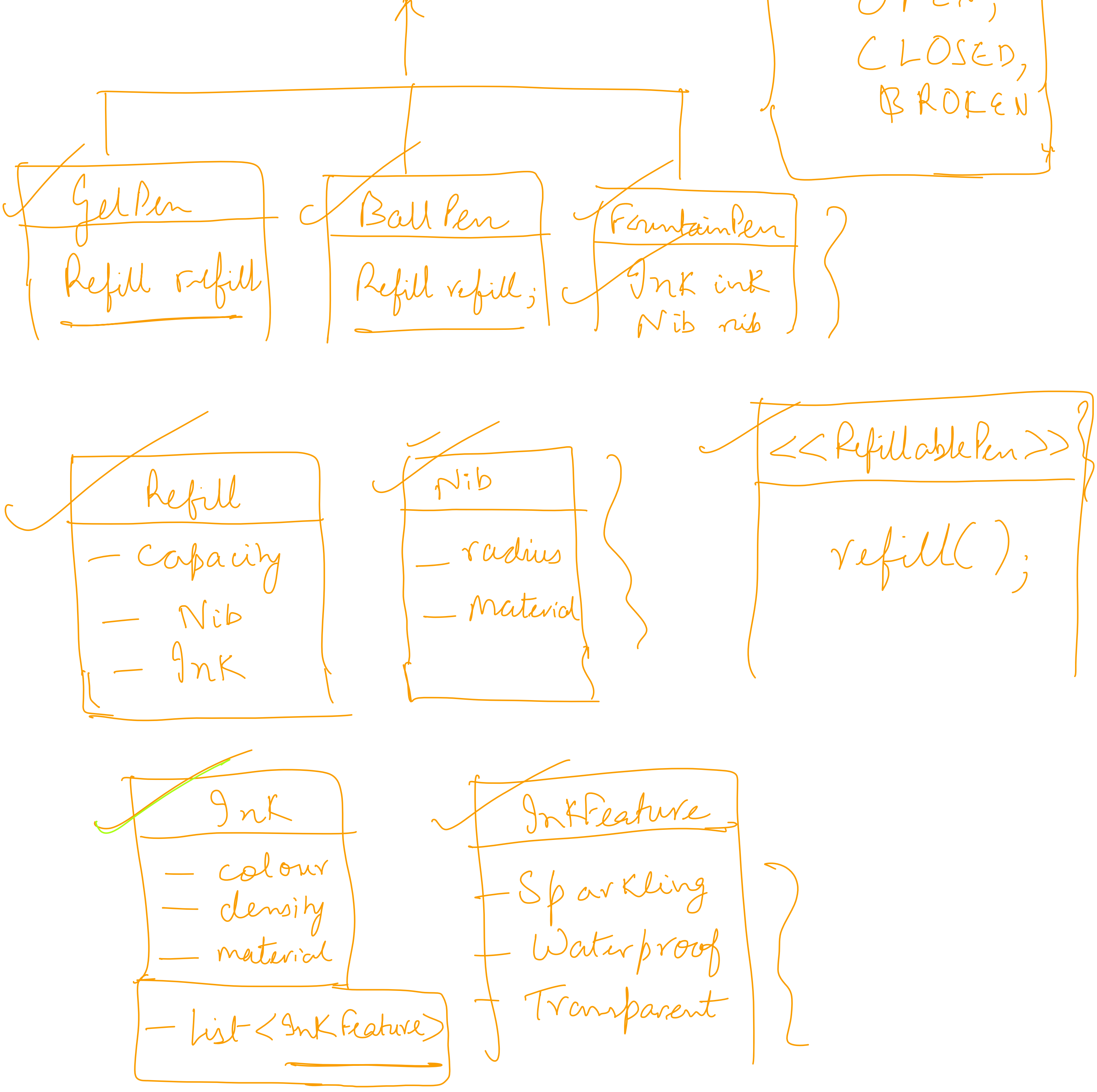
STEP-2 ⟹ Class Diagram

⟹ Try to visualize the system

⟹ Or try to think about all the nouns/entities.

⟹ You are not expected to come up with all the classes (edge classes) but imp classes, interfaces, enums, maybe even design patterns.




OPEN,
CLOSED,
BROKEN
GelPen
Refill refill
BallPen
Refill refill;
FountainPen
Ink ink
Nib nib
Refill
— capacity
— Nib
— Ink
Nib
— radius
— material
<<RefillablePen>>
refill( );
Ink
— colour
— density
— material
— List<InkFeature>
InkFeature
Sparkling
Waterproof
Transparent