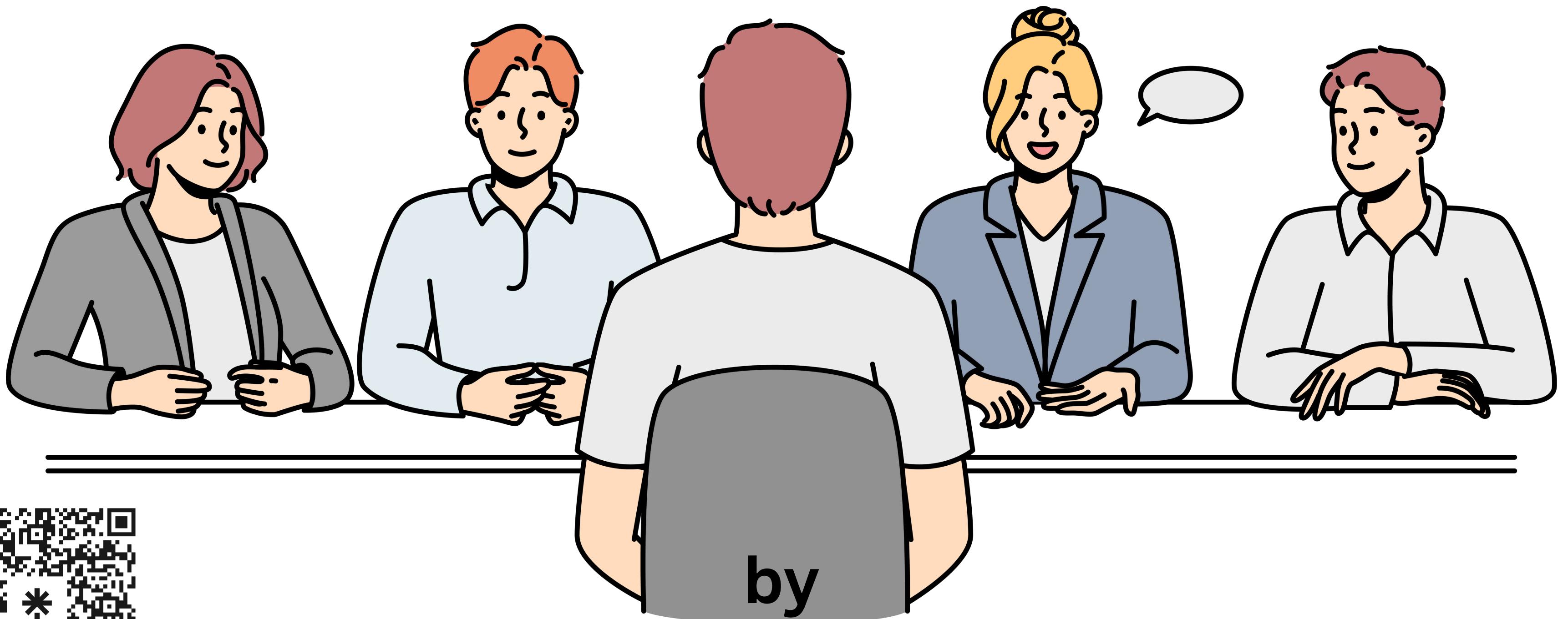


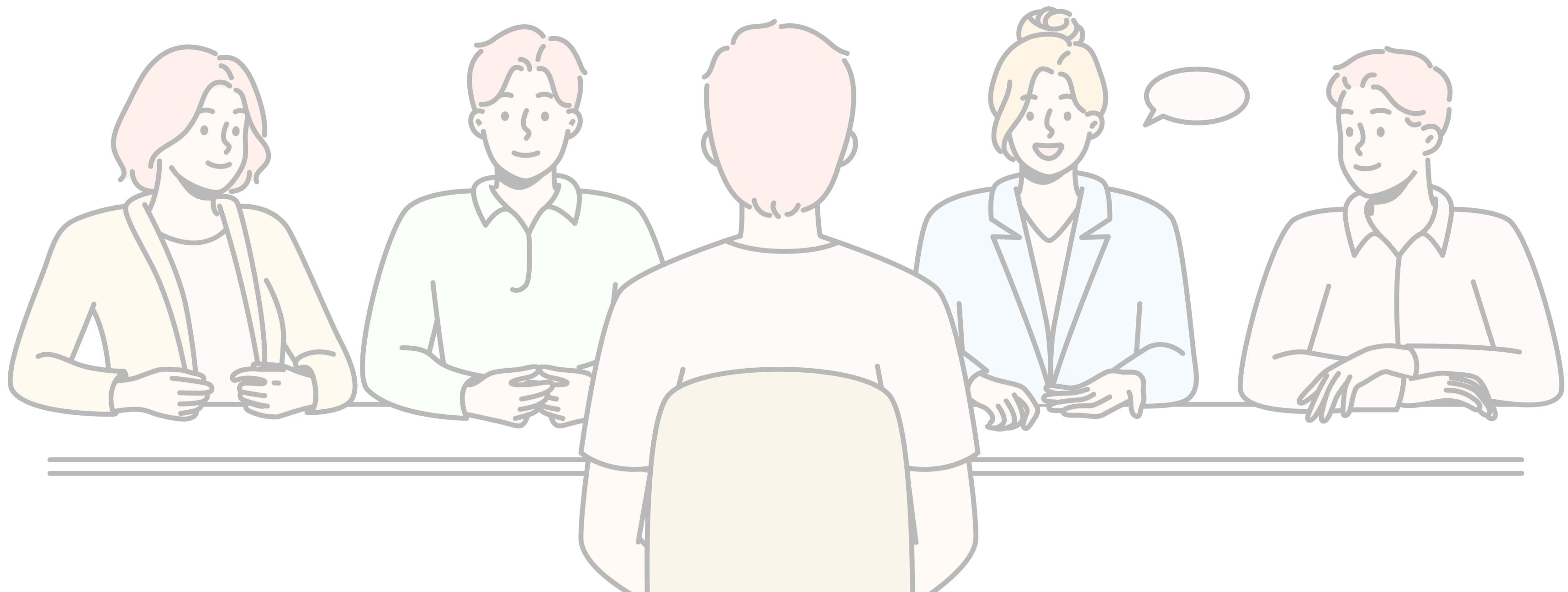
Cracking The Interview Code



by

Dr. Amar Panchal

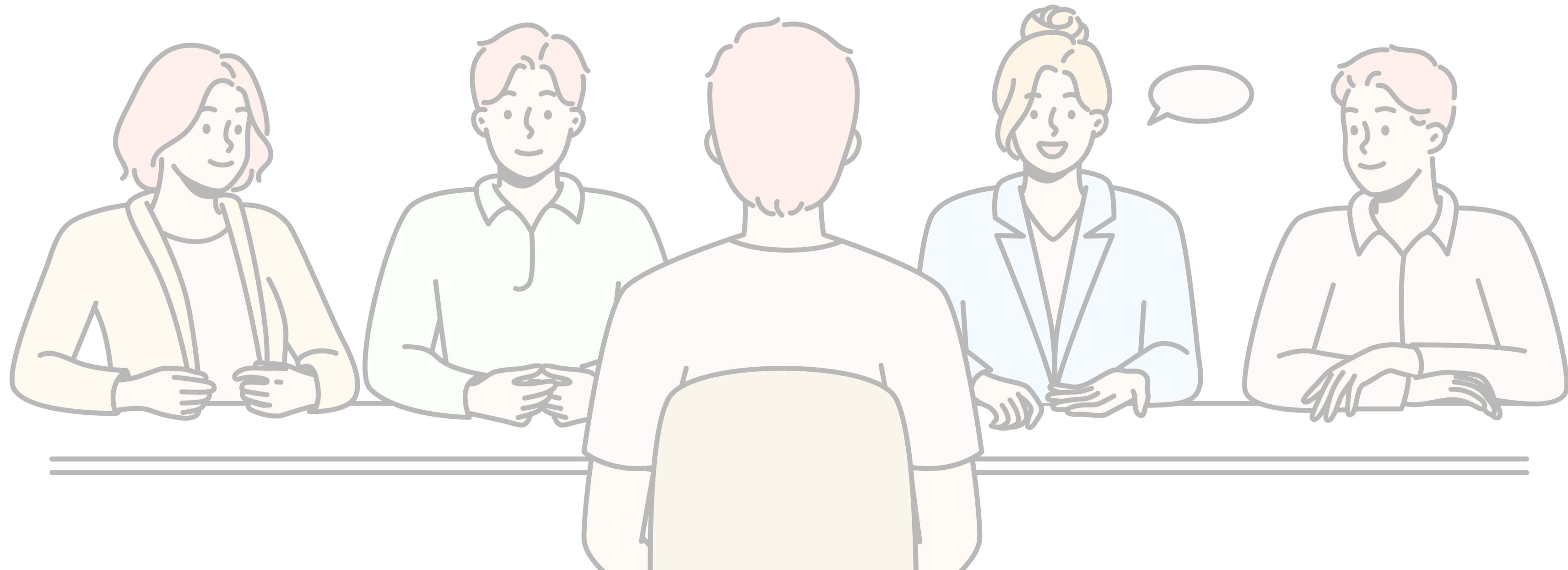
What is Interview ?



Know You Self ?

what are your strength ?

what are your weaknesses ?



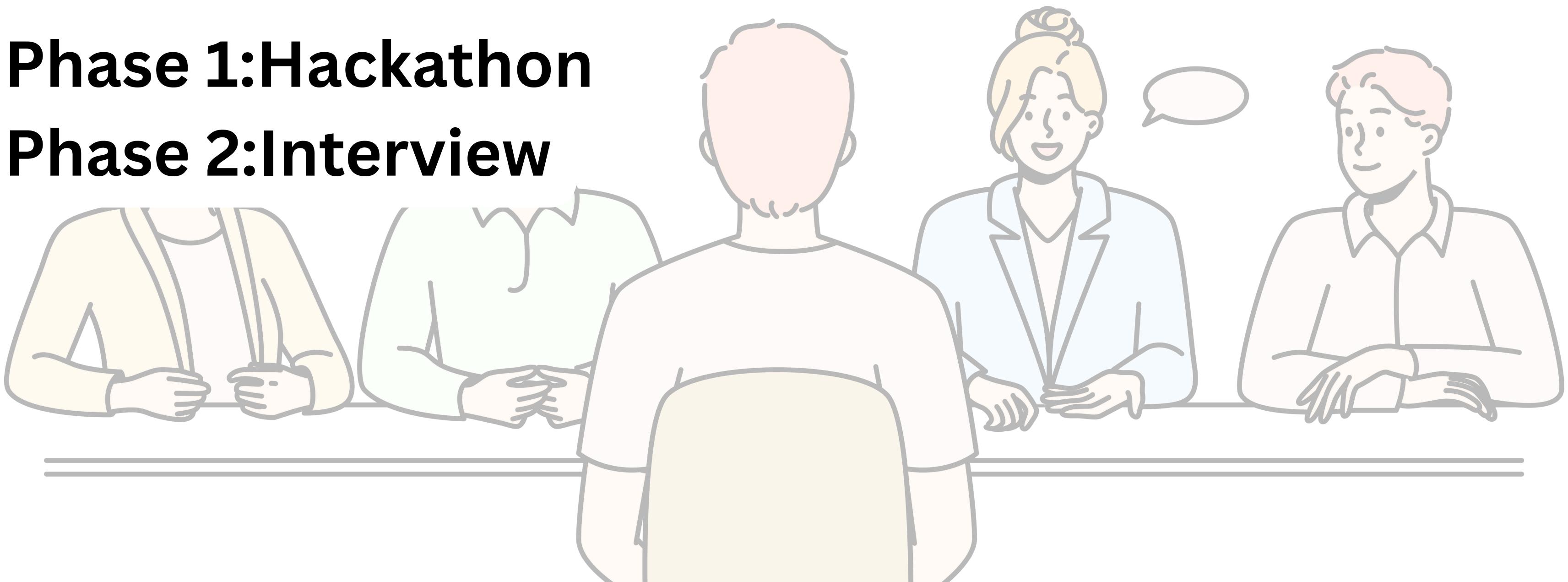
Most Common Rounds

Phase 1:Aptil+Technical+Coding

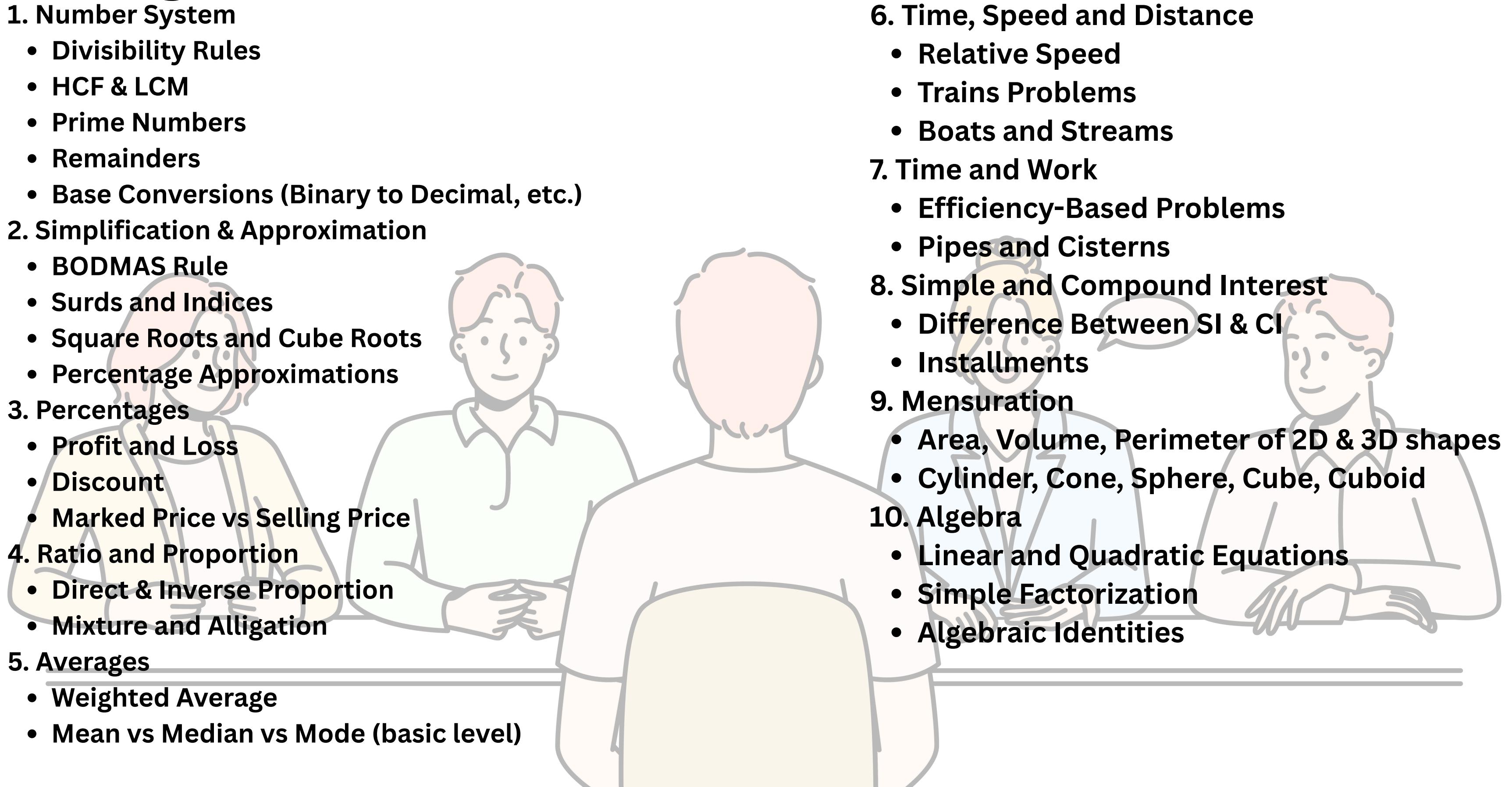
Phase 2:Interview

Phase 1:Hackathon

Phase 2:Interview



Major Quantitative Topics

- 
1. Number System
 - Divisibility Rules
 - HCF & LCM
 - Prime Numbers
 - Remainders
 - Base Conversions (Binary to Decimal, etc.)
 2. Simplification & Approximation
 - BODMAS Rule
 - Surds and Indices
 - Square Roots and Cube Roots
 - Percentage Approximations
 3. Percentages
 - Profit and Loss
 - Discount
 - Marked Price vs Selling Price
 4. Ratio and Proportion
 - Direct & Inverse Proportion
 - Mixture and Alligation
 5. Averages
 - Weighted Average
 - Mean vs Median vs Mode (basic level)
 6. Time, Speed and Distance
 - Relative Speed
 - Trains Problems
 - Boats and Streams
 7. Time and Work
 - Efficiency-Based Problems
 - Pipes and Cisterns
 8. Simple and Compound Interest
 - Difference Between SI & CI
 - Installments
 9. Mensuration
 - Area, Volume, Perimeter of 2D & 3D shapes
 - Cylinder, Cone, Sphere, Cube, Cuboid
 10. Algebra
 - Linear and Quadratic Equations
 - Simple Factorization
 - Algebraic Identities

Major Logical Reasoning & Aptitude Topics

1. Blood Relations

- Puzzle-based
- Coded Relations

2. Seating Arrangement

- Circular / Linear Arrangements
- With conditions

3. Direction Sense

- Distance calculation
- Turns (Left, Right, etc.)

4. Syllogism

- Venn Diagram based questions

5. Coding-Decoding

- Letter and Number based patterns

6. Series Completion

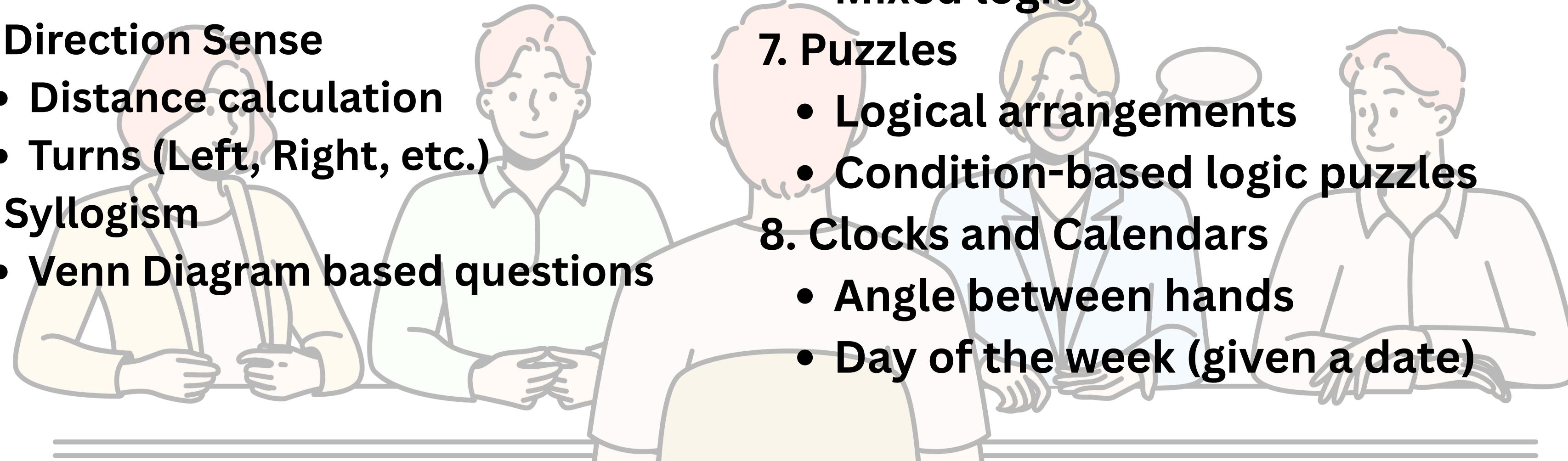
- Number series
- Alphabet series
- Mixed logic

7. Puzzles

- Logical arrangements
- Condition-based logic puzzles

8. Clocks and Calendars

- Angle between hands
- Day of the week (given a date)



Core Technical Topics

- 1. Programming Languages (C / C++ / Java / Python)**
- 2. Data Structures and Algorithms**
- 3. Database Management (SQL + DBMS)**
- 4. Operating System Concepts**
- 5. Computer Networks**
- 6. Object-Oriented Programming (Java / C++)**
- 7. Web Technologies (mostly for full stack or DBDA students)**



Facts

10%-20% only clears phase-1 of interview

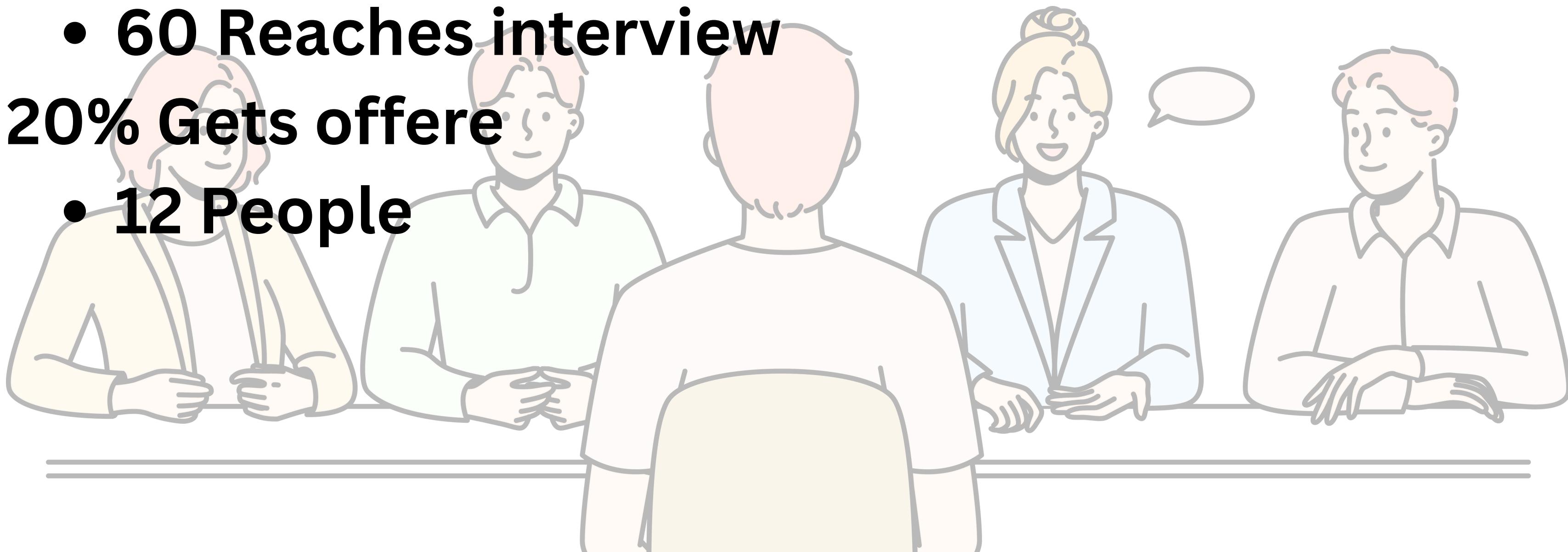
- 1000--->200 will go to next

30% of that gets the interview round

- 60 Reaches interview

20% Gets offer

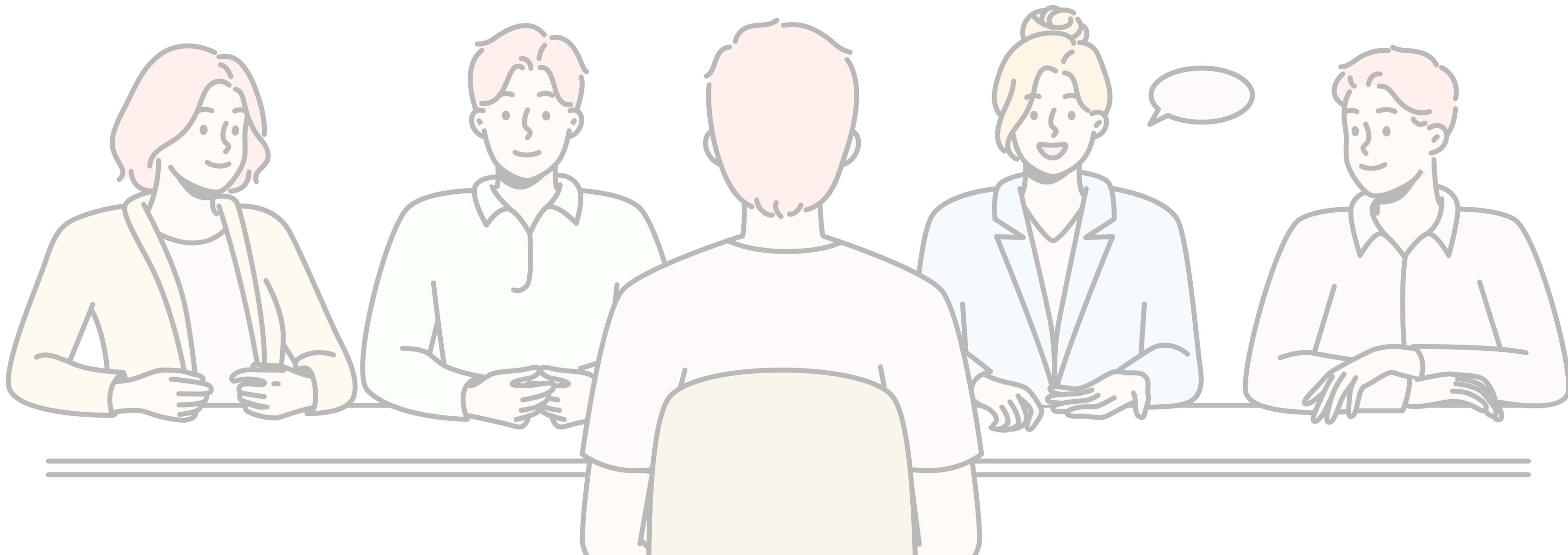
- 12 People



How to prepare ?

First learn what you bad at ?

Practice what you good at ?

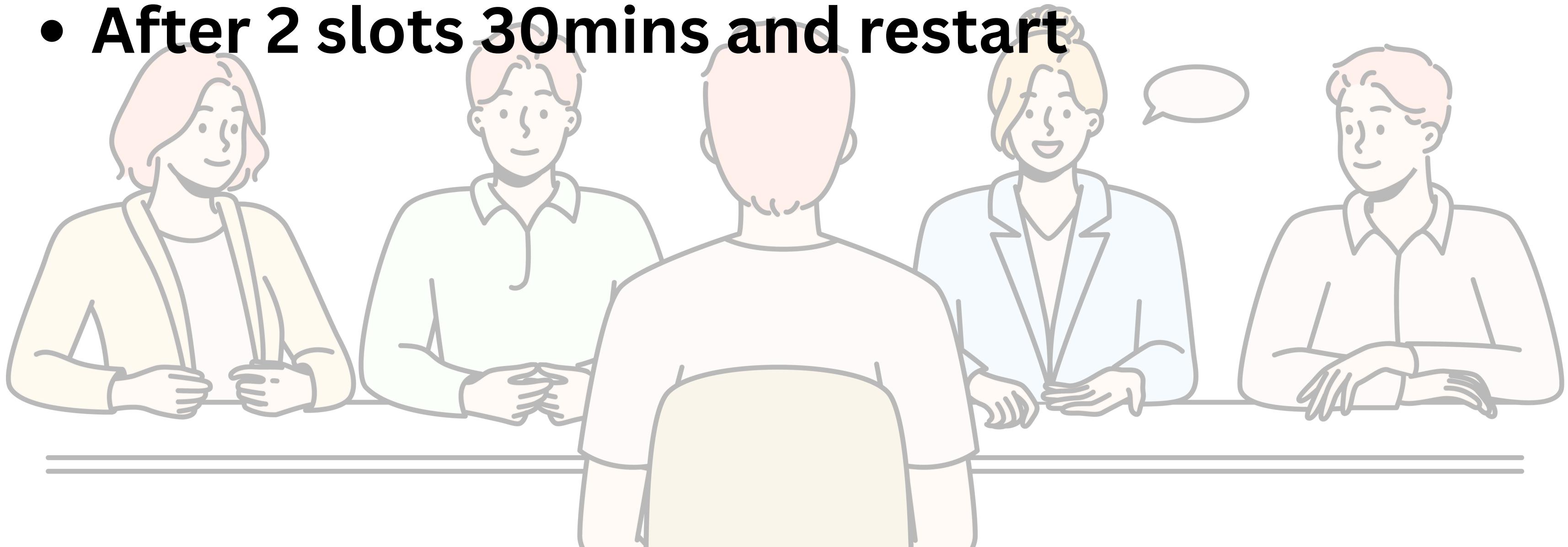


Time Management ?

One Thing at a time(Uniprocessing)

Time Based Preparation(Time Sloting)

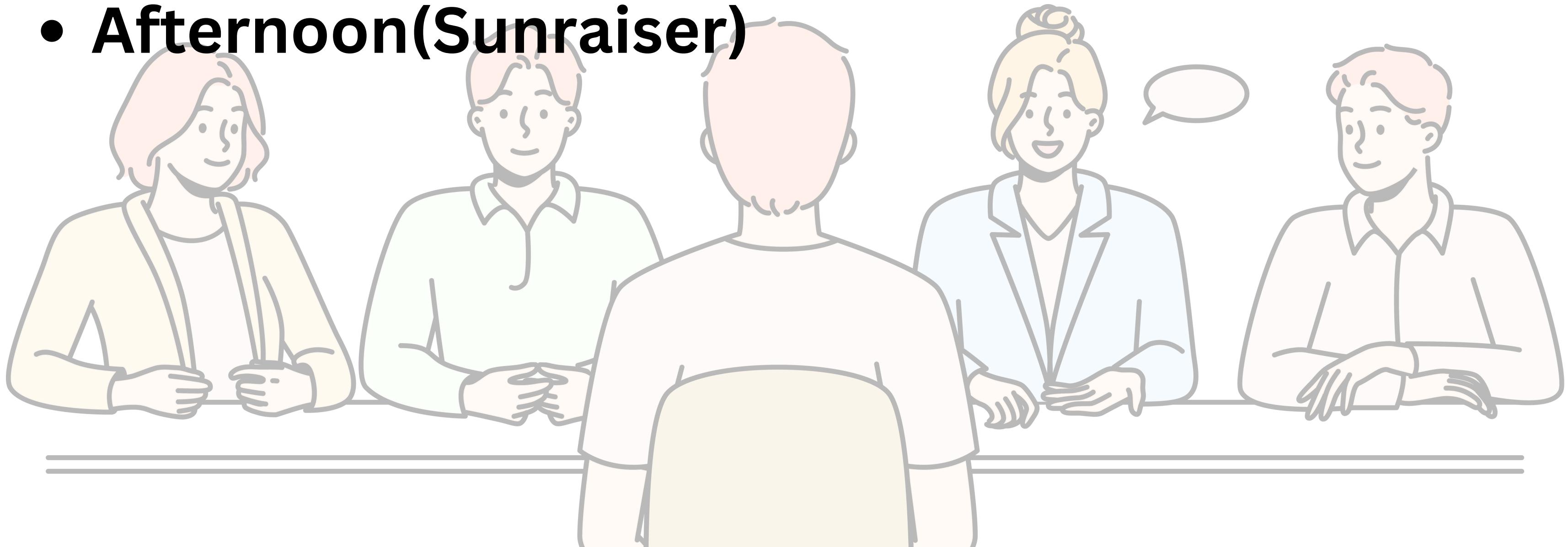
- Takes Break(every 45mins-5mins)
- After 2 slots 30mins and restart



Time Management ?

When

- Morning(Early-Birds)
- Night(Night-Owl)
- Afternoon(Sunraiser)

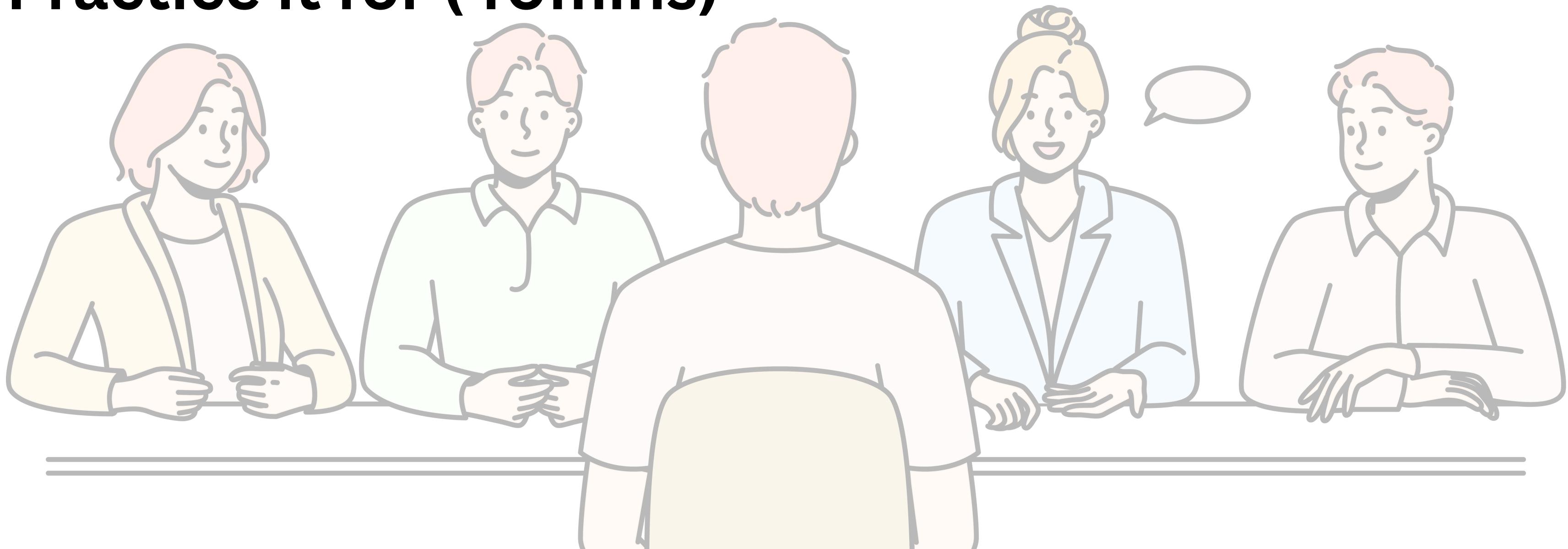


How To Remember ?

1+4=5

Read/Listen/Watch topic and make notes(10mins)

Practice it for (40mins)



Use AI: To prepare not to cheat

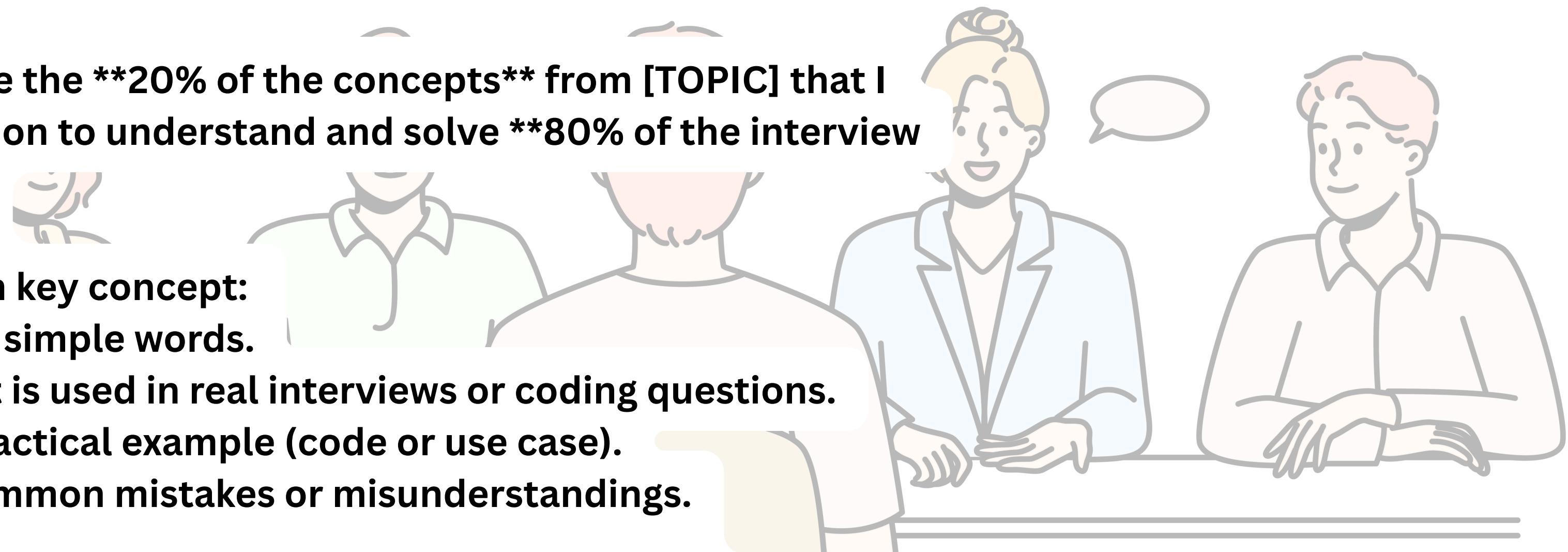
use:

I am preparing for interviews and I want to learn the most important parts of [TOPIC].

Please tell me the ****20% of the concepts**** from [TOPIC] that I should focus on to understand and solve ****80% of the interview questions****.

Also, for each key concept:

- Explain it in simple words.
- Show how it is used in real interviews or coding questions.
- Give one practical example (code or use case).
- Mention common mistakes or misunderstandings.



Finally, give me a checklist to revise this topic before my interview.

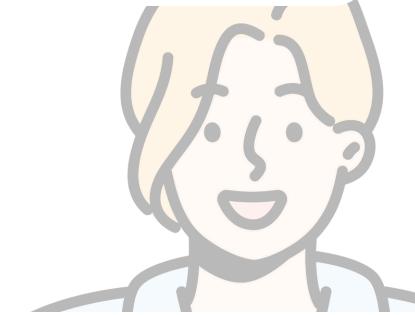
Use AI: To Check Knowledge

use:

Act as a technical interviewer.

I am preparing for interviews and I want you to ask me the most commonly asked and important questions from the topic [TOPIC].

Please:



1. Ask me 8–10 high-impact interview questions.
2. For each question, give me a brief idea of what a good answer should include.
3. Highlight the most important points I should definitely mention.
4. Ask 2 bonus advanced or tricky questions from [TOPIC].
5. At the end, show me a checklist to mark how prepared I am for [TOPIC].

Use simple English and assume I am preparing for an entry-level or fresher position.

Interview Process

1. Tell Me about yourself
2. Basic+Concept
3. Technology+use
4. Project+what you did ?
5. What Company wants



Myths

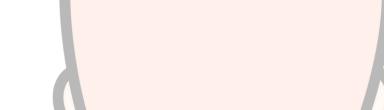
Myth 1:

You must know everything to crack the interview.

Myth 2:

Interviewers always expect perfect solutions.

Myth 3: 

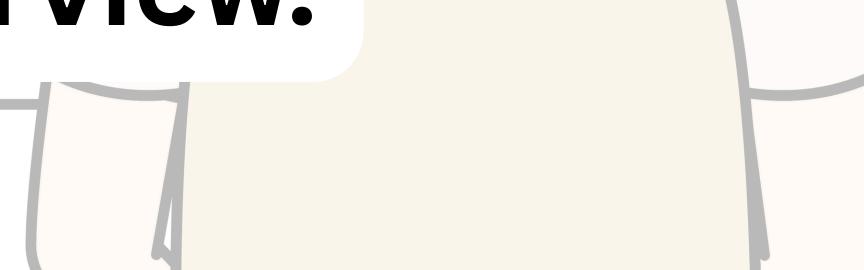


Only people from top colleges get selected.

Myth 4: 



You need to solve complex problems under pressure in every interview.



Myths

Myth 5:

**Interviews are just about technical knowledge;
soft skills don't matter.**

Myth 6:

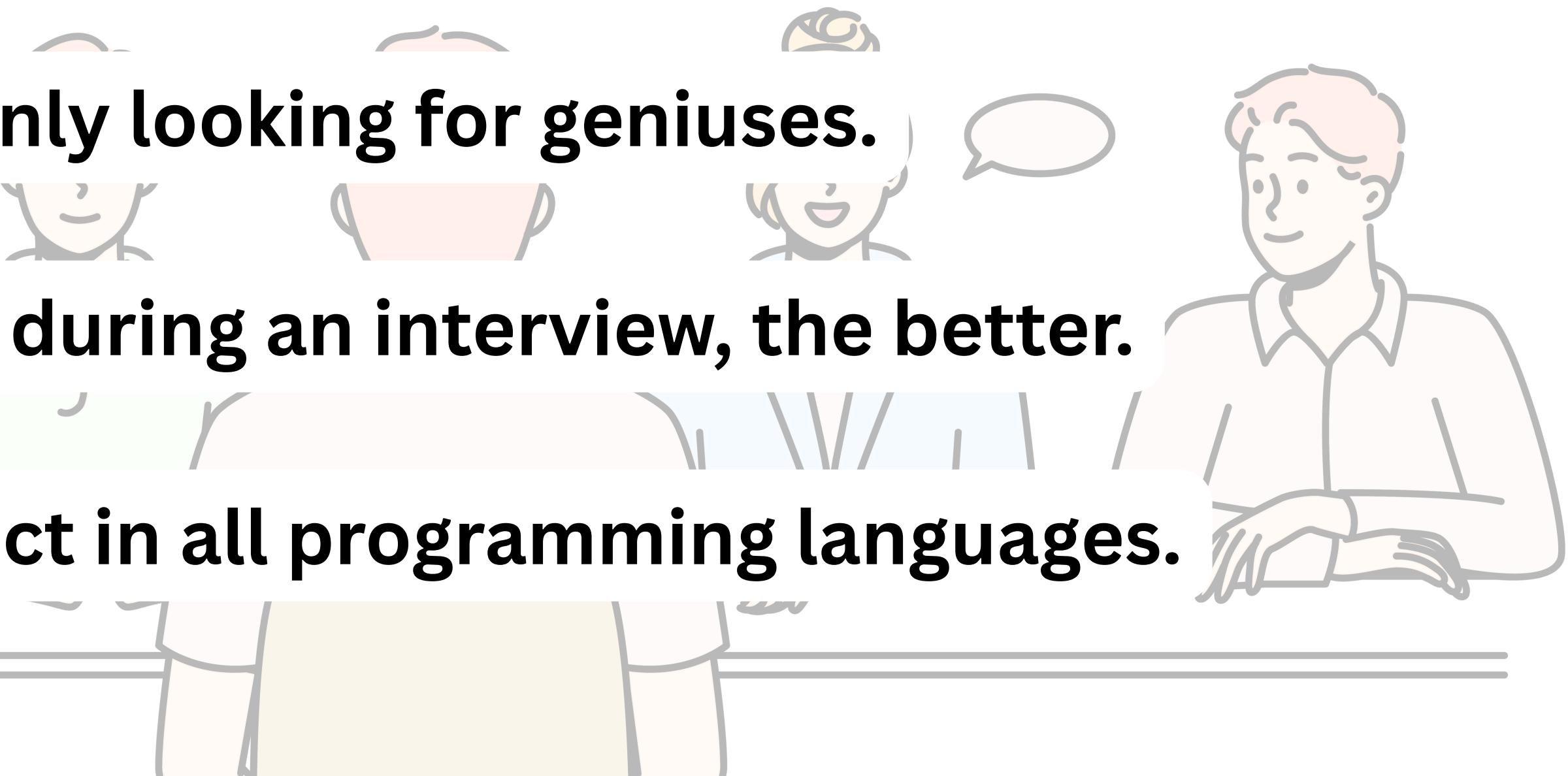
Interviewers are only looking for geniuses.

Myth 7:

The more you talk during an interview, the better.

Myth 8:

You must be perfect in all programming languages.



Myths

Myth 9:

Technical interviews are only about solving problems in a computer.

Myth 10:

Once you are rejected, you won't be considered again.

Myth 11:

Interview preparation can be done in a week or two.



All
the
Best

