✸ PART A — Research Questions...

1. What new improvements were introduced in Gemini 3.0?
Answer:

Dimaagi Taqat (Reasoning): Pehle se behtar sochta hai aur PhD level ki reasoning kar sakta hai.

Multimodal Samajh: Text, video, audio, aur tasweer sab ko ek saath samajh kar kaam karta hai.

Lambi Yaad-daasht (Context): Bohat zyada data (1 million tokens tak) yaad rakh kar lambe kaam karta hai, isliye baat nahi bhoolta.

Khud Kaam Karne Wala (Agentic): Sirf jawab dene ki bajaye, aapke complex, multi-step kaam plan karke khud poore karta hai.

2. How does Gemini 3.0 improve coding & automation workflows?
Answer:

Agentic Coding: Yeh model poore project ka plan banata hai aur multi-step tasks ko khud execute karta hai (ek junior developer ki tarah).

Poori Codebase Ki Samajh: Apni Lambi Yaad-daasht se, yeh poori code files ko samajh kar debugging aur refactoring karta hai.

Tez Prototyping: Developers ke aasan ideas se interactive web interfaces turant bana deta hai.

3. How does Gemini 3.0 improve multimodal understanding?
Answer:

Data Synthesis: Yeh text, images, audio, aur video ko alag nahi, balki ek saath milakar unke aapas ke complex rishton ko samajhta hai.

Deep Analysis: Yeh lambe video lectures ko analyze kar sakta hai aur kharab quality ki photos se bhi structured data nikal sakta hai.

Real-World Data: Medical scans ya factory images jaisey mushkil data ko gehrai aur safai se samjhne mein madad karta hai.

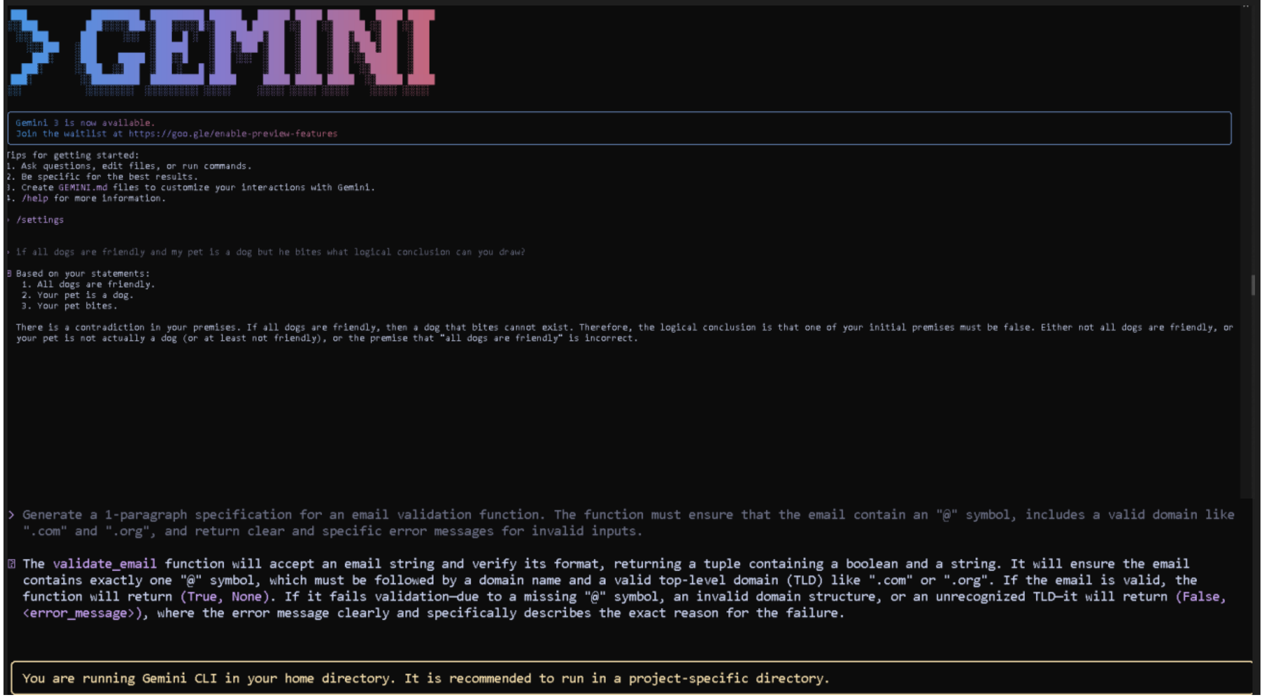
4. Name any two developer tools introduced with Gemini 3.0? (Teen Tools shamil kiye gaye hain)
Answer: Gemini 3.0 ke saath aaye teen (3) aham tools yeh hain:

Google Antigravity: Yeh naya agentic development platform hai jahan AI agents poora coding workflow khud karte hain.

Generative UI (User Interface): Yeh technology aasan prompt se interactive apps aur visual interfaces turant generate kar deti hai.

Gemini CLI (Command Line Interface): Developers ko terminal ke zariye seedha model se code generate karne aur debugging ki sahoolat milti hai.

✸ PART B – Practical Task (screenshot)



```
> GEMINI

Gemini 3 is now available.
Join the waitlist at https://goo.gle/enable-preview-features

Tips for getting started:
1. Ask questions, edit files, or run commands.
2. Be specific for the best results.
3. Create GEMINI.md files to customize your interactions with Gemini.
4. /help for more information.

> /settings


> if all dogs are friendly and my pet is a dog but he bites what logical conclusion can you draw?

▯ Based on your statements:
   1. All dogs are friendly.
   2. Your pet is a dog.
   3. Your pet bites.

  There is a contradiction in your premises. If all dogs are friendly, then a dog that bites cannot exist. Therefore, the logical conclusion is that one of your initial premises must be false. Either not all dogs are friendly, or
  your pet is not actually a dog (or at least not friendly), or the premise that "all dogs are friendly" is incorrect.

> Generate a 1-paragraph specification for an email validation function. The function must ensure that the email contain an "@" symbol, includes a valid domain like
  ".com" and ".org", and return clear and specific error messages for invalid inputs.

▯ The validate_email function will accept an email string and verify its format, returning a tuple containing a boolean and a string. It will ensure the email
  contains exactly one "@" symbol, which must be followed by a domain name and a valid top-level domain (TLD) like ".com" or ".org". If the email is valid, the
  function will return (True, None). If it fails validation—due to a missing "@" symbol, an invalid domain structure, or an unrecognized TLD—it will return (False,
  <error_message>), where the error message clearly and specifically describes the exact reason for the failure.

 You are running Gemini CLI in your home directory. It is recommended to run in a project-specific directory.
```