🔷 Lecture 20: Google AI Studio & Gemini APIs

---

1 What is Google AI Studio?

- A platform by **Google** offering access to its **Gemini models**
- Supports text generation, conversations, multimodal tasks
- API-based usage for developers to integrate AI into apps

---

2 Why Learn Google AI APIs?

- ☑ Different APIs = Different implementations
- ☑ Broaden understanding of cloud-based AI tools
- ☑ Gemini models offer unique capabilities
- ☑ Learn key management and multi-platform adaptability

---

3 Features of Gemini APIs

- 🧠 **Multi-turn chat**: Keeps memory of conversation
- 🎯 Tunable params:
  - `temperature` (creativity)
  - `top_p`, `top_k`
  - `max output tokens`
- 🛠 Highly **customizable and flexible**

---

4 Get Started with Google AI API (Python)

🔷 Install Required Package

```
bash
CopyEdit
pip install google-generative-ai
```

🔷 Import & Configure

```
python
CopyEdit
import google.generativeai as genai
genai.configure(api_key="YOUR_API_KEY")
```

👉 Always store your key in `.env`, not in code!

5️⃣ Listing Models (Check Setup)

```
python
CopyEdit
models = genai.list_models()
gemini_models = [model.name for model in models if 'gemini' in
model.name.lower()]
```

✅ Output example:

```
bash
CopyEdit
Available Gemini models:
- models/gemini-1.5-pro
- models/gemini-pro
```

6️⃣ Create a Prompt Function

```
python
CopyEdit
def getGeminiResponse(prompt: str) -> str:
    model = genai.GenerativeModel("gemini-pro")
    response = model.generate_content(prompt)
    return response.text
```

7️⃣ Where to Get API Key?

- Go to **Google AI Studio**
- Accept usage terms
- Create a **new API key**
- Securely store it in `.env`:

```
ini
CopyEdit
GOOGLE_API_KEY=your_key_here
```

---

8 Debugging & Environment

- Use **Python debuggers** to trace code flow
- Check variables at breakpoints
- Helps understand how AI responses are generated
- Google AI supports Java too, but focus here is Python

---

9 Learn from Documentation

- Always refer to:
  - Input/output formats
  - Model-specific params
  - Usage examples

 Official docs are your best teacher!

---

10 Best Practices ✓

Tip	Why It Matters
Use `.env` file	Keeps keys private
Use try-except	Handles errors cleanly
Read Docs	Understand all features
Practice prompts	Learn response tuning
Stay updated	APIs evolve over time

☆ Mindset Shift

- Don't wait for others — **start exploring**
- Trust your Python skills to adapt to any API
- Every new API = New skill = More powerful apps
- Use AI tools (GPT, Claude) to understand code better

✅ Summary

- **Google AI Studio** = Gateway to Gemini models
- API allows text generation & conversations
- Secure key mgmt + `.env` handling is must
- Use Python to make requests
- Practice debugging & self-learn from docs