

The course teaches what each of these things *does*. The exam asks what it is **called**. That is a short, learnable list, and it is the whole of this sheet. Learn the rule in the orange box under each block — that is the bit that decides the answer.

1 · Sending things to a model (Responses API)

<code>input_text</code>	A text part.
<code>input_image</code>	An image part — an https URL or base64 data. Both are allowed.
<code>input_audio</code>	An audio part. The WAV must be base64-encoded — not a filename, not a path.
<code>input_file</code>	A file part (documents).
<code>modalities</code>	What you want <i>back</i> . Want audio returned? Ask for text and audio .

Name the part by what you are **SENDING**. Ask for modalities by what you want **RETURNED**. A recorded WAV in, text and speech out = `input_audio` (base64) plus both modalities.

2 · Wiring up the client

<code>AIProjectClient</code>	Created with the project endpoint and a credential.
<code>.get_openai_client()</code>	Call this next. It hands back an authenticated OpenAI client.
<code>.responses.create()</code>	The unified call. Not <code>chat.completions</code> , not <code>deployments.get()</code> , not <code>models.list()</code> .
<code>model=</code>	The deployment name — never the base model name. To reach an existing agent, put the agent name here.

`AIProjectClient` → `get_openai_client()` → `responses.create()`. Never "use the deployment URL as the project endpoint".

3 · Pictures: making one vs changing one

Text-to-image	A prompt and nothing else. A new image out of thin air.
Image-to-image	An existing image plus a prompt. This is editing.

If the question hands you a PNG, it is **EDITING**. If it only hands you words, it is **GENERATION**.

4 · Vision feature words

tagging	Metadata that summarizes the attributes of the image. A flat list of words.
categorizing	Places the image in a hierarchy / taxonomy.
object detection	What is there and where — bounding boxes.
OCR	Reads text that appears inside the picture.

The phrase "metadata that summarizes the attributes" is Microsoft's own wording for **TAGGING**. Tags are flat; categories are hierarchical.

5 · Content Understanding

analyzer	Defines the content type , extraction settings and output structure . All three.
enableOcr	Turns on reading text in images and frames . It does NOT produce transcripts.
transcriptPhrases	Speaker-attributed speech , with start and end times.
keyFrameTimesMs	Timestamps of representative visual frames .
cameraShotTimesMs	Where the shot changes . Not the same as representative frames.

Transcript = words. KeyFrame = pictures. OCR = text printed in the picture. Audio and video transcripts never come from enableOcr.

6 · Speech

speech_synthesis_voice_name	Sets the voice , e.g. en-US-Ava:DragonHDLatestNeural.
speech_synthesis_language	Sets the language only. It will not pick a named voice.
default speaker output	Play it out loud.
file output	Save it to disk.

Named a specific voice? It is voice_name. Asked to play on the default speaker? Speaker output, not file.

7 · Language service methods

recognize_entities()	People, places, organisations — with matching text and category.
recognize_linked_entities()	The same, but linked out to a knowledge base.
extract_key_phrases()	The main talking points.
analyze_sentiment()	Positive / negative / neutral, with confidence scores.
Language detection	The primary language and a confidence score. Nothing else.

Language detection does NOT redact PII and does NOT do sentiment. Those are separate features.