📘 MCQs: output\_type Behavior & Schema Strictness – Fundamentals of Agentic AI

**1. What does output\_type define in Agentic AI?**

A) The tool execution method  
B) The structure and type of the agent’s output  
C) The retry policy  
D) The agent’s system prompt

**Answer:** B  
**Explanation:** output\_type specifies the **format/schema** the model’s output must follow.

**2. Which library is commonly used to enforce output\_type schemas?**

A) TensorFlow  
B) Pydantic  
C) NumPy  
D) Scikit-learn

**Answer:** B  
**Explanation:** **Pydantic models or dataclasses** are used to validate outputs against schemas.

**3. What happens if the model’s response doesn’t match the output\_type schema?**

A) It is ignored  
B) A validation error is raised  
C) The model retries automatically  
D) The run is canceled

**Answer:** B  
**Explanation:** **Schema validation** triggers a **ValidationError** on mismatch.

**4. Which setting controls how strict the schema enforcement is?**

A) strict\_schema  
B) schema\_mode  
C) strictness  
D) validate

**Answer:** C  
**Explanation:** **Schema strictness** determines how rigidly the model must follow the schema.

**5. Which output\_type is best if you want structured JSON outputs?**

A) str  
B) dict  
C) Pydantic BaseModel  
D) int

**Answer:** C  
**Explanation:** A **Pydantic model** enforces JSON schema with strong validation.

**6. What is a common use case for setting output\_type to a dataclass?**

A) Tool retries  
B) Returning structured business objects like User(name, age)  
C) Handling exceptions  
D) Managing prompts

**Answer:** B  
**Explanation:** Dataclasses allow agents to output **structured Python objects**.

**7. If output\_type is set to str, what behavior is expected?**

A) Output must be a schema object  
B) Output is returned as freeform text without validation  
C) Output is always converted into JSON  
D) Output is retried until valid

**Answer:** B  
**Explanation:** A str output type allows **freeform natural language text**.

**8. Which is the main benefit of strict schema enforcement?**

A) Lower token usage  
B) Predictable and structured outputs  
C) More creative results  
D) Faster runs

**Answer:** B  
**Explanation:** Strictness ensures outputs are **machine-usable and predictable**.

**9. Which strictness level allows the model more freedom in formatting?**

A) Strict  
B) Relaxed  
C) Exact  
D) Blocked

**Answer:** B  
**Explanation:** A **relaxed schema mode** allows the model to deviate slightly.

**10. What happens in strict mode if the model generates extra fields not in schema?**

A) They are ignored  
B) Validation fails  
C) They are auto-corrected  
D) They are appended

**Answer:** B  
**Explanation:** **Strict validation** rejects unexpected fields.

**11. Which output\_type is best for tool interoperability?**

A) Pydantic BaseModel  
B) str  
C) list  
D) int

**Answer:** A  
**Explanation:** Tools often require **structured validated outputs**, best enforced by BaseModel.

**12. Which error is most likely if output\_type = BaseModel but model produces plain text?**

A) MaxTurnsExceeded  
B) ValidationError  
C) ToolExecutionError  
D) TimeoutError

**Answer:** B  
**Explanation:** **ValidationError** occurs when output cannot be parsed into schema.

**13. Why might a developer prefer output\_type=str in early prototyping?**

A) To avoid strict parsing failures  
B) To enforce schemas  
C) To limit randomness  
D) To control tokens

**Answer:** A  
**Explanation:** str is **lenient**, making experimentation easier.

**14. How can schema strictness affect retries?**

A) Strict schemas reduce retries  
B) Strict schemas increase retries due to validation failures  
C) Strictness has no effect  
D) Retries always happen

**Answer:** B  
**Explanation:** If strictness is high, **model retries** may be triggered until schema is satisfied.

**15. Which strictness level is best for production APIs?**

A) Relaxed  
B) Strict  
C) None  
D) Experimental

**Answer:** B  
**Explanation:** **Strict schemas** are necessary for **reliable production pipelines**.

**16. Which output\_type behavior allows list of structured objects?**

A) list[BaseModel]  
B) dict  
C) str  
D) tuple

**Answer:** A  
**Explanation:** You can set output\_type to **list of models** for structured collections.

**17. If schema strictness is relaxed, what happens to unexpected fields?**

A) They are allowed and passed through  
B) They break validation  
C) They reset output  
D) They trigger retries

**Answer:** A  
**Explanation:** Relaxed mode is **forgiving**, allowing unexpected fields.

**18. What is the trade-off of using strict schema enforcement?**

A) Higher reliability, but more retries and possible latency  
B) Lower reliability, but faster runs  
C) More creative, but less structured  
D) None

**Answer:** A  
**Explanation:** Strict schemas ensure correctness, but may **increase retries and latency**.

**19. Which is the most flexible output\_type for natural conversations?**

A) str  
B) BaseModel  
C) dataclass  
D) JSON

**Answer:** A  
**Explanation:** **Freeform text (str)** is best for conversational agents.

**20. Why is schema strictness important in Agentic AI pipelines?**

A) Ensures interoperability and downstream processing correctness  
B) Reduces GPU cost  
C) Increases temperature  
D) Controls retries

**Answer:** A  
**Explanation:** Schema strictness ensures **agents and tools integrate reliably**.