The template below uses a **JSON Lines (.jsonl)** format. This means each line in your file is a self-contained JSON object, making it easy for fine-tuning tools to process.

**Template Structure**

Each example should be a single JSON object with three key-value pairs:

{

"instruction": "Your agent's overall mission statement.",

"input": "The specific problem or scenario.",

"output": "The desired and structured response, including reasoning and tool calls."

}

**Concrete Example for Inventory Troubleshooting**

Here is a ready-to-use template with placeholders. Copy this and fill in the details for each of your scenarios.

**File: inventory\_data.jsonl**

{

"instruction": "You are a proactive Inventory Management Agent. Your task is to audit system logs, identify critical inventory discrepancies, and provide a detailed analysis followed by the precise, structured tool calls needed for a resolution. Always prioritize issues that affect product availability and transaction integrity.",

"input": "System Report:\n- Product ID '{PRODUCT\_ID}' in '{WAREHOUSE\_NAME}' is reporting a negative quantity of {NEGATIVE\_QUANTITY}.\n- Transaction '{TRANSACTION\_ID}' has been flagged as 'stuck' for {HOURS\_STUCK} hours.\n- The last audit of product '{ANOTHER\_PRODUCT\_ID}' showed a supplier ID '{INCORRECT\_SUPPLIER\_ID}' that does not match the master record '{CORRECT\_SUPPLIER\_ID}'.",

"output": "Analysis: The most critical issue is the negative inventory for product '{PRODUCT\_ID}', as it directly impacts our ability to fulfill orders. The stuck transaction '{TRANSACTION\_ID}' is the next priority due to its time sensitivity. The supplier ID discrepancy is a data quality issue that should be addressed after the more critical issues. Tool Calls: 1. check\_product\_status(product\_id='{PRODUCT\_ID}'). 2. adjust\_quantity(product\_id='{PRODUCT\_ID}', new\_quantity={CORRECT\_QUANTITY}). 3. resolve\_transaction(transaction\_id='{TRANSACTION\_ID}'). 4. update\_product\_data(product\_id='{ANOTHER\_PRODUCT\_ID}', field='supplier\_id', new\_value='{CORRECT\_SUPPLIER\_ID}')."

}

**Best Practices for Creating Your Dataset**

* **Consistency is Key:** Keep the instruction field exactly the same for every single example. This trains the model on its core identity and mission.
* **Structured Output:** The output field is where the model learns how to generate structured responses. Make sure the format for your Analysis and Tool Calls is identical in every example.
* **Write the "Why":** The "Analysis" part of the output is a form of **Chain of Thought (CoT)** training. It teaches the model the reasoning behind its actions, making it much more robust and reliable.
* **Use Real Data (Anonymized):** Base your input and output on real-world scenarios and the actual APIs you have. Just be sure to anonymize any sensitive data.
* **Quantity over Quantity (to a point):** Aim for quality over a large quantity. A hundred well-crafted, diverse examples are more valuable than a thousand identical or poorly written ones.

By using this template, you'll be creating a powerful and effective fine-tuning dataset.