Task 1: AI-Powered Code Completion

1. Which Version is More Efficient?

   The more efficient version is Code 2 (The AI-generated code – task_1.2.py).

2. Here is Why.

   Code 1 arranges a predefined list of dictionaries based on the 'name' key, which is simple but limited in versatility. If a different sorting criterion is required, the code must be updated, which incurs additional effort and increases the possibility of errors. It also does not encapsulate the sorting algorithm; therefore, it is less reusable.

   Code 2, on the other hand, offers a method called `sort_dicts_by_key` that allows you to sort any list of dictionaries by any key you specify. This modular approach promotes code reuse and maintainability. The function accepts parameters for the list, the key to sort by, and an optional reverse flag, allowing it to adapt to different sorting requirements without changing the fundamental logic.

   Additionally, Code 2's usage of the `get()` method protects against missing keys by returning None rather than issuing an error, which is very handy in bigger datasets. Overall, Code 2's design encourages good coding techniques, making it the preferred option for sorting dictionaries in Python.

NOTE: Code 1 is the manual code (task_1.1.py) while Code 2 is the AI-generated code (task_1.2.py).