

Timestamp Addition

How I Solved It:

1. Get the Current Date and Time:

- I used JavaScript's Date object to get the current date and time. The Date object provides methods to retrieve the year, month, day, hours, minutes, and seconds.

2. Format the Date and Time:

- I had to format the date and time in YYYY-MM-DD HH:MM:SS format.
 - **Year:** Directly retrieved using getFullYear().
 - **Month:** Retrieved using getMonth(), but since it's zero-indexed (January is 0), I added 1 and used padStart(2, '0') to ensure it's always two digits.
 - **Day:** Retrieved using getDate(), and padded to two digits.
 - **Hours, Minutes, Seconds:** Similarly, retrieved and padded to two digits to maintain consistent formatting.

3. Combine Date and Time into Timestamp:

- I created a timestamp string by concatenating the formatted date and time. This string was then added to the message object as a new field.

4. Return the Updated Message:

- I returned a new object that included all the original message fields plus the new timestamp.

Problems I Faced:

• Property Existence:

- One challenge was ensuring that the function handled cases where msg.propertyName did not exist in msg.payload. If the property is missing, I needed to return null to indicate it wasn't found. This required a check for undefined.

• Input Validation:

- Ensuring that msg.propertyName was a valid property name and msg.payload was an object were additional concerns. If msg.propertyName was not a string or msg.payload was not an object, the function might not work as expected. However, the basic implementation assumes that inputs are correctly formatted.

Problems I Faced:

• Property Missing:

- One hiccup was handling cases where the property wasn't in `msg.payload`. If you asked for a property that didn't exist, I had to make sure the function didn't crash and just returned null.
- **Input Validity:**
 - I had to assume that `msg.propertyName` was a valid string and `msg.payload` was an object. If these weren't what I expected, things could go sideways. But for this task, I kept it simple and focused on the basic functionality.