Python Debug Challenge - Payment Calculator

Problem Statement

You are given a function `calculate_payments(applicants)` that processes payment data for multiple applicants and calculates total payments grouped by currency. The function appears to work for basic cases but contains subtle bugs that cause incorrect results in certain scenarios.

Current Code

python

```
def calculate_payments(applicants):
    payment_by_currency = {}  # Dictionary to store total payments by currency
    for app in applicants:
        # Get currency, default to "GEL" if missing, convert to uppercase
        currency = str(app.get("currency", "GEL")).upper()
        payments = app.get("payments")  # Get list of payments for this applicant
        for pay in payments:
            # Only process active payments (default to True if not specified)
            if pay.get("active", True):
                # Set incomeshare to 1 if it's None, otherwise use the value
                incomeshare = float(pay.get("incomeshare") or 1)
                base = pay.get("base")  # Get the base value for ratio calculation
                ratio = incomeshare / base  # Calculate the ratio
                amount = float(pay.get("amount", 0))  # Get payment amount, default to 0
                # Add calculated payment to currency total
                payment_by_currency[currency] = payment_by_currency.get(currency, 0) + amount *
    return payment_by_currency
```

What the Code Should Do

The function should:

1. Process a list of applicants, each with a currency and list of payments
2. For each payment that is active, calculate: `(incomeshare / base) * amount`
3. Sum all calculated payments by currency
4. Return a dictionary with currencies as keys and total payment amounts as values

Input Format

python

```python
applicants = [
    {
        "currency": "USD",
        "payments": [
            {
                "active": True,
                "incomeshare": 0.2,
                "amount": 1000,
                "base": 0.5
            },
            {
                "active": False,
                "incomeshare": 0.3,
                "amount": 500,
                "base": 0.4
            }
        ]
    }
]
```

Your Task

1. **Understand the Code**: Explain what each part of the function does
2. **Find the Bugs**: Identify all subtle bugs in the code by creating test cases
3. **Fix the Issues**: Provide a corrected version of the function
4. **Test Thoroughly**: Create comprehensive test cases that expose the bugs

Example Test Case

Here's one test case to get you started:

python

```python
applicants = [
    {
        "currency": "USD",
        "payments": [
            {"active": True, "incomeshare": 0, "amount": 100, "base": 0.5}
        ]
    }
]

result = calculate_payments(applicants)
print("Result:", result)
# What should this output? What does it actually output?
```

Requirements

1. **Bug Identification**: List all bugs you find with explanations
2. **Fixed Function**: Provide a corrected version that handles edge cases
3. **Test Suite**: Create test cases that demonstrate both the bugs and fixes
4. **Error Handling**: Add appropriate error handling for invalid inputs