# Test Cases for Rewards Calculation

## Test Case: calculateRewardPoints

Description: This test case checks the calculation of reward points based on transaction amounts.

Test Cases:

1. Transaction amount is 120:

- Input: 120

- Expected Output: 90

2. Transaction amount is 75:

- Input: 75

- Expected Output: 25

3. Transaction amount is 50:

- Input: 50

- Expected Output: 0

## Test Case: calculateRewards

Description: This test case checks the calculation of rewards for multiple transactions.

Test Cases:

1. Transactions:

- Input: [{ customerId: 1, date: '2024-01-15', amount: 120 }, { customerId: 1, date: '2024-02-15', amount: 75 }]

- Expected Output: rewards[1].total is 115, rewards[1].monthly[1] is 90, rewards[1].monthly[2] is 25

Test Cases for RewardsComponent

This document describes the test cases for the RewardsComponent using the Jest and React Testing Library. The RewardsComponent fetches transaction data, calculates rewards, and displays the rewards for each customer. The test cases ensure the component's functionality and error handling.

# Test Case 1: Renders Rewards for Customers

\*\*Description\*\*: Verifies that the component renders rewards for each customer correctly.

\*\*Mocks Used\*\*:

1. `fetchTransactions`: Returns a mock list of transactions.

2. `calculateRewards`: Returns a mock rewards object.

\*\*Steps\*\*:

1. Render the RewardsComponent.

2. Wait for the `fetchTransactions` function to be called.

3. Check if the customer IDs and total points are rendered.

4. Check if the monthly points are rendered.

\*\*Expected Results\*\*:

1. The component should render customer IDs and their total points.

2. The component should render the monthly points for each customer.

# Test Case 2: Displays Error Message on API Failure

\*\*Description\*\*: Verifies that the component displays an error message when the API call fails.

\*\*Mocks Used\*\*:

1. `fetchTransactions`: Rejects with an error to simulate a failed API call.

\*\*Steps\*\*:

1. Mock `fetchTransactions` to reject with an error.

2. Render the RewardsComponent.

3. Wait for the `fetchTransactions` function to be called.

4. Check if the error message is displayed.

\*\*Expected Results\*\*:

1. The component should display an error message: "Failed to load transactions."