**Test Cases Documentation**

**Status Code Check Purpose**:

Ensure that the API response returns a successful status code.

**Functionality**: Verifies that the HTTP status code is 200 (OK).

pm.test("Status code is 200", *function* () {

    pm.response.to.have.status(200);

});

**Response Time Check Purpose:**

Assess the timeliness of the API response.

**Functionality**: Checks that the response time is less than 500 milliseconds.

pm.test("Response time is less than ms", *function* () {

    pm.expect(pm.response.responseTime).to.be.below(500);

});

**Response Body Validation Purpose:**

Validate the structure and content of the API response body.

**Functionality**: Ensures the presence of essential properties (info, total, pages, results). Validates the results array is not empty. Checks each result for answer\_to\_questions, sell\_price, and bid\_price properties.

pm.test("Response body has valid data, including answer\_to\_questions, sell price, and bid price", *function* () {

*const* responseData = pm.response.json().responseData;

    pm.expect(responseData).to.be.an('object').that.has.property('info');

    pm.expect(responseData.info).to.be.an('object').that.has.property('total');

    pm.expect(responseData.info).to.be.an('object').that.has.property('pages');

    pm.expect(responseData).to.be.an('object').that.has.property('results');

    pm.expect(responseData.results).to.be.an('array').that.is.not.empty;

    responseData.results.forEach(*function*(*result*) {

        pm.expect(result).to.be.an('object').that.has.property('answer\_to\_questions').and.to.be.a('string').and.to.have.lengthOf.at.least(1);

        pm.expect(result).to.be.an('object').that.has.property('sell\_price').and.to.be.a('number').and.to.be.at.least(0);

        pm.expect(result).to.be.an('object').that.has.property('bid\_price').and.to.be.a('number').and.to.be.at.least(0);

    });

});

**Product Images and Defected Images Check Purpose:**

Confirm the presence and validity of product and defected images.

**Functionality**: Verifies that each product has product\_images and defected\_images properties. Ensures both properties are arrays.

pm.test("Each product has valid product\_images and defected\_images", *function* () {

*const* responseData = pm.response.json().responseData;

    responseData.results.forEach(*function*(*result*) {

        pm.expect(result).to.be.an('object').that.has.property('product\_images').and.to.be.an('array');

        pm.expect(result).to.be.an('object').that.has.property('defected\_images').and.to.be.an('array');

    });

});

**Trade-In Status Check Purpose:**

Validate the trade\_in status for each product.

**Functionality**: Verifies that each product has a trade\_in property. Ensures the trade\_in property is a boolean.

pm.test("Each product has valid trade\_in status", *function* () {

*const* responseData = pm.response.json().responseData;

    responseData.results.forEach(*function*(*result*) {

        pm.expect(result).to.be.an('object').that.has.property('trade\_in').and.to.be.a('boolean');

    });

});

**Date Validation Purpose:**

Confirm the validity of the createdDate for each product.

**Functionality:** Checks that each product has a valid createdDate in string format. Validates the createdDate as a legitimate date.

pm.test("Each product has a valid Date", *function* () {

*const* responseData = pm.response.json().responseData;

    responseData.results.forEach(*function*(*result*) {

*const* createdDate = result.createdDate;

        pm.expect(createdDate).to.be.a('string');

        // Validate if the createdDate is a valid date

*const* isValidDate = !isNaN(*Date*.parse(createdDate));

        pm.expect(isValidDate).to.be.true;

    });

});

**Test Report Summary:** Total Test Cases: 6 Passed: Passed

**Status Code Check**

**Result**: Passed

**Comments**: The API response returned a status code of 200.

**Response Time Check**

**Result:** Passed

**Comments:** The API response time was below the expected threshold of 500 milliseconds.

**Response Body Validation**

**Result:** Passed

**Comments:** The response body structure and content were as expected. All essential properties were present, and individual result properties were validated.

P**roduct Images and Defected Images Check**

**Result:** Passed

**Comments**: Each product had valid product\_images and defected\_images properties.

**Trade-In Status Check**

Result: Passed

**Comments:** The trade\_in status for each product was a boolean value.

**Date Validation**

**Result:** Passed

**Comments:** Products had an valid createdDate format.

