

### Explore Integration Test:

This integration test focuses on the Explore component and its interaction with the FlightCard component. It verifies the following:

1. Correct rendering of flight information fetched from a mocked API call.
2. Proper display of FlightCard components with correct props.
3. Functionality of various filters:
  - a. Discount filter
  - b. Date filter
  - c. Price range filter
4. Clearing of filters and re-display of all flights.

The test mocks the axios library for API calls and the airport image catalogue to provide consistent test data.

### Test Command:

```
npm test -- ExploreIntegration.test.js
```

### Add Flights Integration Test:

This integration test focuses on the AddFlights component, which is responsible for adding new flights to the system. The test verifies the following:

1. Proper rendering of the AddFlights form.
2. Successful fetching of plane IDs and populating the dropdown.
3. Correct handling of user input for all flight details.
4. Successful submission of the form with the entered flight data.
5. Proper API call to add a new flight.
6. Correct display of success message after adding a flight.

The test uses mock APIs (using jest-fetch-mock and axios-mock-adapter) to simulate backend interactions without actually making network requests.

### Test Command:

```
npm test -- AddFlightsIntegration.test.jsx
```

### Sign Up Integration Test:

This integration test focuses on the SignUp component, which handles user registration. The test verifies the following:

1. Proper rendering of the SignUp form.
2. Correct handling of user input for all registration fields.
3. Enabling the Register button when terms are accepted.
4. Successful submission of the registration form with entered user data.
5. Proper API call to register a new user.
6. Correct display of success message after registration.
7. Navigation to the login page after successful registration.

The test uses fetchMock to simulate backend interactions without making actual network requests, and mocks the react-router-dom's useNavigate hook to verify navigation.

### Test Command:

```
npm test -- SignUp.test.jsx
```