

```

def test_sign_in_with_wrong_password_raises_runtime_error(monkeypatch, mocker):
    # Mocks for driver
    driver = mocker.Mock()
    driver.implicitly_wait.return_value = None
    driver.get.return_value = None
    driver.current_url = ""
    # Patch handle_same_page_username_password and home_page, etc. to avoid side effects
    monkeypatch.setattr('mintapi.signIn.handle_same_page_username_password', lambda *a,**kw: None)
    monkeypatch.setattr('mintapi.signIn.user_selection_page', lambda *a,**kw: None)
    monkeypatch.setattr('mintapi.signIn.handle_login_failures', lambda d: (_ for _ in []).throw(RuntimeError('Login to Mint failed: incorrect password')))
    try:
        sign_in(email='test@example.com', password='bad', driver=driver, wait_for_sync=False)
        assert False, 'RuntimeError should be raised for invalid password'
    except RuntimeError as e:
        assert 'incorrect password' in str(e)

```

---Extra Details---

Filename: signIn.py

Description: Verifies that entering an incorrect password triggers the correct RuntimeError, as expected in handle_login_failures.

Score: 97%

Alignment: 98%

Validation Notes: Tests error-handling logic for wrong password, emulating the driver and login failure.