import request from "supertest";

import app from "./app"; // your Express app

import { Pool } from "pg";

const db = new Pool({ connectionString: process.env.DATABASE\_URL });

beforeAll(async () => {

await db.query("DELETE FROM users");

await db.query("DELETE FROM posts");

});

describe("Auth", () => {

it("rejects weak passwords", async () => {

const res = await request(app).post("/signup").send({ email: "a@b.com", password: "123" });

expect(res.status).toBe(400); // validation should fail

});

it("creates and logs in a user", async () => {

await request(app).post("/signup").send({ email: "a@b.com", password: "StrongPass123!" });

const res = await request(app).post("/login").send({ email: "a@b.com", password: "StrongPass123!" });

expect(res.status).toBe(200);

expect(res.headers["set-cookie"]).toBeDefined();

});

});

describe("Posts", () => {

let cookie;

beforeAll(async () => {

const res = await request(app)

.post("/login")

.send({ email: "a@b.com", password: "StrongPass123!" });

cookie = res.headers["set-cookie"];

});

it("rejects creating post without auth", async () => {

const res = await request(app).post("/posts").send({ title: "No auth", content: "x" });

expect(res.status).toBe(401);

});

it("creates a post with sanitized content", async () => {

const res = await request(app)

.post("/posts")

.set("Cookie", cookie)

.send({ title: "Hello", content: "<script>alert(1)</script><p>safe</p>" });

expect(res.status).toBe(200);

const { rows } = await db.query("SELECT content FROM posts ORDER BY id DESC LIMIT 1");

expect(rows[0].content).toBe("<p>safe</p>"); // XSS stripped

});

});