Name: Shreya Dubey **Intern ID**: 241

Tool Name:

Homoglyph URL Shortener

Description: What is this tool about?

This tool is a Python-based web application that shortens long URLs into randomized, short URLs. It incorporates **homoglyph characters** (Unicode lookalikes) into the shortened codes, making it potentially useful for phishing simulation or research into deceptive link creation. Built using Flask and SQLite.

Characteristics / Features

- Flask-based web interface for URL shortening.
- Homoglyph character use in short codes (e.g., Cyrillic and Latin mix).
- Stores long-short URL mappings in a SQLite database.
- Redirects to original URLs using shortened code.
- Uses visual spoofing with hardcoded Unicode domains (e.g., "https://google.com").

Types / Modules Available

- generate_short_code(): Random short code with homoglyphs.
- store_url(): Saves original URL and retrieves/creates a unique short code.
- get_long_url(): Resolves a short code to its original URL.
- Flask endpoints:

    - '/': Form to input long URL.
    - '/<short_code>': Redirect to original URL.
- init_db(): Initializes the SQLite database.

How Will This Tool Help?

- Demonstrates how homoglyphs can be used in malicious or misleading URLs.
- Useful in red teaming/phishing simulations.
- Provides insight into Unicode-based deception in URL handling.
- Teaches developers and analysts about homoglyph-based attack vectors.

Example Usage

1. Start the Flask app:
2. python urlshortner.py
3. Visit: http://127.0.0.1:5000
4. Enter any long URL.
5. The app generates a short URL like:
6. https://google.com/aь1ceo

Best Case Scenarios

- Testing how web filters react to homoglyph-encoded domains.
- Simulating deceptive URLs in phishing awareness campaigns.
- Studying user perception of visually similar characters.
- Academic or cybersecurity research on Unicode abuse in URLs.

How to Use in Investigation

- Track whether a homoglyph-based short URL bypasses filters.
- Investigate if security systems resolve homoglyphs as trusted domains.
- Analyze logs to see how users interact with these misleading links.

People Who Can Use the Tool

- Cybersecurity researchers
- Red teamers / Ethical hackers
- Threat intelligence analysts
- Developers studying Unicode security issues
- Educators teaching phishing defense mechanisms

Required Skills

- Basic Python and Flask knowledge
- Understanding of Unicode homoglyphs
- Familiarity with web development and security

Flaws

- May confuse users into thinking it's safe due to visual mimicry.
- Uses fixed spoofed domain ("https://google.com/")—not flexible.
- No access control or logging.

- No custom domain integration.
- Vulnerable to abuse if deployed in public without security checks.

Suggestions to Improve

- Replace hardcoded spoof domain with configurable input.
- Add analytics or logging support for red team engagement.
- Sanitize URLs to prevent injection attacks.
- Support expiry times for short URLs.
- Highlight potentially deceptive homoglyphs to users.

Good

- Simple, functional, and fast.
- Demonstrates homoglyph use in an applied scenario.
- Great for phishing simulation, training, and research.

Summary

The urlshortner.py script is a URL shortening web tool that cleverly integrates homoglyph characters into short codes and spoofed domains.
