1. Basic Socket Request (20 pts)

Write a Python **program** that connects to a web server using the `socket` module and sends a simple HTTP GET request.

Requirements

- Connect to the public web server **`example.com`** on **port 80**.
- Send a valid HTTP `GET` request for the root path `/`.
- Receive the server's response and print the **first 1024 bytes**, decoded as **UTF-8 text**.
- Close the socket after receiving the response.

Example

```
# Expected output (truncated)
HTTP/1.1 200 OK
Content-Type: text/html
ETag: "bc2473a18e003bdb249eba5ce893033f:1760028122.592274"
...
```

Hints

- Use the following functions from the `socket` module:
  - `socket.socket()`
  - `.connect()`
  - `.sendall()`
  - `.recv()`
  - `.close()`
- The HTTP request format:

  ```
  GET / HTTP/1.1
  Host: example.com
  ```

  You can create the request by:

  ```
  request = "GET / HTTP/1.1\r\nHost: example.com\r\n\r\n"
  ```

- Use `encode` to encode the request and `decode` to decode the response.

2. Bytes and Strings Conversion (20 pts)

Write a Python **program** that allows the user to input a string, and then converts it between **bytes** and **string** formats.

Requirements

- Prompt the user to enter a string using `input`.
- Convert the input string into a **UTF-8 encoded bytes object** and print it.
- Convert the bytes object back into a **UTF-8 decoded string** and print it.
- Finally, print a short explanation of the difference between `encode()` and `decode()`.

Example

```
Enter a string: Hello, Python!
b'Hello, Python!'
Hello, Python!
```

Hints

- Use `str.encode("utf-8")` to convert a string to bytes.
- Use `bytes.decode("utf-8")` to convert bytes back to a string.

3. HTTP Request Inspection (20 pts)

Write a Python **program** that uses the `requests` module to send an HTTP GET request and explore the **methods and attributes** of the response object.

Requirements

- Use `requests.get()` to fetch data from:
  `http://universities.hipolabs.com/search?name=california`
- This request searches for universities whose name contains **“California”**.
- Print the following information:
  - The **status code**
  - The **header**
  - The **final URL** (after any redirects)

  - The **first 3 universities** returned in the JSON response, including at least their `name` , `country` , and `web_pages`
  - The **total number of universities** returned by the request

Example

```
Status Code: 200
{'Server': 'nginx/1.14.0', 'Date': 'Thu, 30 Oct 2025 04:46:53 GMT', 'Content-Type': 'application
Final URL: http://universities.hipolabs.com/search?name=california
Number of universities returned: 48
Response preview: [{'alpha_two_code': 'US', 'name': 'Carrington College California-Sacramento',
```

Hints

- Use `response.json()` to convert the response body into a Python list or dictionary.
- To get the total number of universities, you can use the `len()` function on the list returned by `response.json()` .