

Top CGI Python Interview Questions and Answers

1. What is CGI in Python?

Answer: CGI stands for Common Gateway Interface, a standard protocol that allows web servers to execute external programs, usually scripts, to generate dynamic content. In Python, CGI scripts can handle web forms, interact with databases, and dynamically generate web pages.

2. How do you write a simple CGI script in Python?

Answer: A simple CGI script in Python starts with the `#!/usr/bin/env python3` shebang and includes the `cgi` module to handle form data. Here's an example:

```
#!/usr/bin/env python3
import cgi
print("Content-Type:
text/html\n")
print("<html><body>")
print("<h1>Hello, World!</h1>")
print("</body></html>")
```

3. How do you process form data in a Python CGI script?

Answer: Form data in a Python CGI script is processed using the `FieldStorage` class from the `cgi` module. This class parses the form data sent via GET or POST methods and makes it accessible within the script. Here's an example:

```
import cgi

form = cgi.FieldStorage()

name = form.getvalue('name')

print("Content-Type: text/html\r\n\r\n")

print(f"<html><body><h1>Hello, {name}!</h1></body></html>")
```

4. How do you handle file uploads in CGI using Python?

Answer: To handle file uploads, use the `cgi.FieldStorage` object and write the uploaded file to the server. Example:

```
form = cgi.FieldStorage()

fileitem = form['file']

if fileitem.filename: with
open(f'/path/to/uploads/{fileitem.filename}', 'wb') as f:
f.write(fileitem.file.read())
```

5. What are the common headers used in CGI scripts?

Answer: The most common headers are **Content-Type**, which defines the MIME type of the response (e.g., `text/html`), and **Status**, which specifies the HTTP status code (e.g., `200 OK`).

6. What are the risks and how can you prevent them?

Answer: CGI scripts can be risky if not handled properly. Here are some of the risks:

- **Injection Attacks:** When user input is not sanitized, it can lead to command injection or SQL injection attacks.
- **Information Disclosure:** Poor error handling can expose your server's sensitive info to users.
- **File Permission Issues:** Incorrect file permissions can allow access to scripts or data without permission.

To prevent these risks, always validate and sanitize user input, don't expose sensitive info in errors and make sure your CGI scripts and directories have the right permissions.

7. What's the difference between GET and POST in CGI?

Answer: GET appends form data to the URL, visible in the browser's address bar. It's used for retrieving data and should not be used for sending sensitive info. POST sends form data in the HTTP request body, not visible in the URL, good for sensitive or large data.

8. How do you do file uploads in CGI using Python?

Answers: File uploads in a CGI script are managed using the **FieldStorage** class. Here's an example of handling file uploads in a Python CGI script:

```
import cgi, os
form = cgi.FieldStorage()
fileitem = form['file']
if fileitem.filename:
```

```
filepath = os.path.join('/path/to/uploads',
os.path.basename(fileitem.filename)) with open(filepath, 'wb') as f:
f.write(fileitem.file.read())
message = 'File uploaded successfully' else: message = 'No file was
uploaded' print("Content-Type: text/html\r\n\r\n")
print(f"<html><body><h1>{message}</h1></body></html>")
```

9. What is the role of the **Content-Type** header in CGI?

Answer: The Content-Type header is important in CGI scripts because it tells the browser what to expect. For example Content-Type: text/html means the server is sending back an HTML document. This header must be included in the CGI script's output before any other content is sent to the browser.

10. Can you explain the difference between CGI and WSGI?

Answer: CGI and WSGI (Web Server Gateway Interface) are both protocols for talking to web servers and web applications. The main difference is CGI creates a new process for each HTTP request which can be resource intensive. WSGI is a more modern and efficient standard for Python web applications, the web server can talk directly to the Python application without the overhead of creating a new process for each request.

11. How do you debug a Python CGI script?

Answer: Debugging a Python CGI script is tough because there is no direct output during execution. Here are the steps to debug:

- Check Server Logs: Look at the server's error logs for any error messages.
- Print Debugging: Use print statements to output variables and execution flow to the HTML output.
- Error Handling: Implement try-except blocks to catch and handle errors in the script.
- File Permissions: Make sure your CGI script and its directory has the correct permissions to be executed by the server.

[Get hands-on with our python course – sign up for a free demo!](#)

12. How do you handle errors in CGI scripts?

Answer: Use try-except blocks to catch exceptions, log errors and give user friendly error messages without showing server internal details.

13. What is the Content-Type header in CGI?

Answer: The Content-Type header tells the browser what to expect, text/html for HTML documents or application/json for JSON data.

14. How do you pass parameters to a CGI script?

Answer: Parameters can be passed via the query string in the URL (for GET requests) or in the body (for POST requests). They can be accessed using the `cgi.FieldStorage` object.

15. What is the shebang line in CGI scripts?

Answer: The shebang line (`#!/usr/bin/env python3`) at the top of the script tells the server which interpreter to use to run the script.

16. How do you handle sessions in CGI?

Answer: Sessions can be managed by storing session data in cookies or server-side files/databases and associating it with a unique session ID sent to the client.

17. What is `cgi.escape()` and why is it important?

Answer: The `cgi.escape()` function (deprecated in Python 3) was used to escape special characters in strings to prevent HTML injection attacks. It's important for sanitizing user inputs that will be displayed as HTML.

18. What is the typical directory structure for CGI scripts?

Answer: CGI scripts are usually stored in a specific directory on the web server, often named `cgi-bin`. This directory needs to have the correct permissions to run scripts.

19. How do you debug CGI scripts?

Answer: Debugging can be done by checking server error logs, printing debug information to the HTML output and using Python's `pdb` debugger for step by step execution.

20. How do you configure a web server to run CGI scripts?

Answer: Configuration depends on the server. For Apache, CGI scripts are enabled by setting the `ScriptAlias` directive and making sure `ExecCGI` is allowed for the directory.

[Get hands-on with our python course – sign up for a free demo!](#)