

DayTime/runserver.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # runserver.py
5: # Author: Bob Dondero
6: #-----
7:
8: import sys
9: import os
10:
11: def main():
12:
13:     if len(sys.argv) != 2:
14:         print('Usage: ' + sys.argv[0] + ' port', file=sys.stderr)
15:         sys.exit(1)
16:
17:     try:
18:         port = int(sys.argv[1])
19:     except Exception:
20:         print('Port must be an integer.', file=sys.stderr)
21:         exit(1)
22:
23:     os.system('python simplehttpserver.py --cgi ' + str(port))
24:
25: if __name__ == '__main__':
26:     main()

```

DayTime/index.html (Page 1 of 1)

```

1: <!DOCTYPE html>
2:
3: <!-- ===== -->
4: <!-- index.html -->
5: <!-- Author: Bob Dondero -->
6: <!-- ===== -->
7:
8: <html>
9:
10:     <body>
11:
12:         <a href="/cgi-bin/daytime.py">
13:             Click here to display the day and time</a>
14:         <br>
15:         <br>
16:         <form action="/cgi-bin/daytime.py" method="get">
17:             Click on the button to display the day and time<br>
18:             <input type="submit">
19:         </form>
20:
21:     </body>
22:
23: </html>

```

DayTime/cgi-bin/daytime.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # daytime.py
5: # Author: Bob Dondero
6: #-----
7:
8: import time
9:
10: #-----
11:
12: def main():
13:
14:     # Write the response headers to stdout.
15:     print('Content-type: text/html; charset=utf-8')
16:     print()
17:
18:     # Write the beginning of the response body to stdout.
19:     print('<!DOCTYPE html>')
20:     print('<html>')
21:     print('<body>')
22:
23:     # Compute the day and time.
24:     daytime = time.asctime(time.localtime())
25:
26:     # Write the day and time to stdout.
27:     print(daytime)
28:
29:     # Write the end of the response body to stdout.
30:     print('</body>')
31:     print('</html>')
32:
33: if __name__ == '__main__':
34:     main()

```

urlencode.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # urlencode.py
5: # Author: Bob Dondero
6: #-----
7:
8: import sys
9: import urllib.parse
10:
11: #-----
12:
13: def main():
14:
15:     if len(sys.argv) != 2:
16:         print('Usage: python %s unencodedstr' % sys.argv[0])
17:         sys.exit(1)
18:
19:     original_str = sys.argv[1]
20:
21:     encoded_str = urllib.parse.quote_plus(original_str)
22:     decoded_str = urllib.parse.unquote_plus(encoded_str)
23:
24:     print('Original string: ' + original_str)
25:     print('Encoded string: ' + encoded_str)
26:     print('Decoded string: ' + decoded_str)
27:
28: #-----
29:
30: if __name__ == '__main__':
31:     main()

```

EchoGet/index.html (Page 1 of 1)

```
1: <!DOCTYPE html>
2:
3: <!-- ===== -->
4: <!-- index.html -->
5: <!-- Author: Bob Dondero -->
6: <!-- ===== -->
7:
8: <html>
9:
10:   <body>
11:
12:     <form action="/cgi-bin/echo.py" method="get">
13:       Enter a line to be echoed:<br>
14:       <input type="text" name="line" value="Enter the line here"><br>
15:       <input type="submit">
16:     </form>
17:
18:   </body>
19:
20: </html>
```

blank (Page 1 of 1)

```
1: This page is intentionally blank.
```

EchoGet/cgi-bin/parseargs.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # parseargs.py
5: # Author: Bob Dondero
6: #-----
7:
8: import urllib.parse
9:
10: #-----
11:
12: def _parse_help(arg_str):
13:     if '=' not in arg_str:
14:         name = urllib.parse.unquote_plus(arg_str)
15:         return (name, None)
16:     name, value = arg_str.split('=')
17:     name = urllib.parse.unquote_plus(name)
18:     value = urllib.parse.unquote_plus(value)
19:     return (name, value)
20:
21: #-----
22:
23: def parse(args_str):
24:     args = {}
25:     arg_strs = args_str.split('&')
26:     for arg_str in arg_strs:
27:         name, value = _parse_help(arg_str)
28:         args[name] = value
29:     return args
30:
31: #-----
32:
33: # For testing:
34:
35: def _test():
36:     args_str = 'name1=value1&name2=value2&name3=value+3&name4=&name5'
37:     args = parse(args_str)
38:     for name, value in args.items():
39:         print(name, ': ', value)
40:
41: if __name__ == '__main__':
42:     _test()

```

EchoGet/cgi-bin/echo.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # echo.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import parseargs
10:
11: #-----
12:
13: def main():
14:
15:     # Write the response headers to stdout.
16:     print('Content-type: text/html; charset=utf-8')
17:     print()
18:
19:     # Write the beginning of the response body to stdout.
20:     print('<!DOCTYPE html>')
21:     print('<html>')
22:     print('<body>')
23:
24:     # Fetch the argument string from the QUERY_STRING environment
25:     # variable.
26:     args_str = os.environ.get('QUERY_STRING', '')
27:
28:     # Fetch the line to be echoed from the argument string.
29:     args = parseargs.parse(args_str)
30:     line_to_be_echoed = args.get('line', None)
31:
32:     # Write the line to be echoed to stdout.
33:     if line_to_be_echoed is not None:
34:         print('<p>This is the line that you entered:</p>')
35:         print('<p>' + line_to_be_echoed + '</p>')
36:     else:
37:         print('<p>You did not enter a line.</p>')
38:
39:     # Write the end of the response body to stdout.
40:     print('</body>')
41:     print('</html>')
42:
43: if __name__ == '__main__':
44:     main()

```

EchoPost/index.html (Page 1 of 1)

```

1: <!DOCTYPE html>
2:
3: <!-- ===== -->
4: <!-- index.html -->
5: <!-- Author: Bob Dondero -->
6: <!-- ===== -->
7:
8: <html>
9:
10:   <body>
11:
12:     <form action="/cgi-bin/echo.py" method="post">
13:       Enter a line to be echoed:<br>
14:       <input type="text" name="line" value="Enter the line here"><br>
15:       <input type="submit">
16:     </form>
17:
18:   </body>
19:
20: </html>

```

EchoPost/cgi-bin/echo.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # echo.py
5: # Author: Bob Dondero
6: #-----
7:
8: import sys
9: import parseargs
10:
11: #-----
12:
13: def main():
14:
15:     # Write the response headers to stdout.
16:     print('Content-type: text/html; charset=utf-8')
17:     print()
18:
19:     # Write the beginning of the response body to stdout.
20:     print('<!DOCTYPE html>')
21:     print('<html>')
22:     print('<body>')
23:
24:     # Read the argument string from stdin.
25:     args_str = sys.stdin.readline()
26:
27:     # Fetch the line to be echoed from the argument string.
28:     args = parseargs.parse(args_str)
29:     line_to_be_echoed = args.get('line', None)
30:
31:     # Write the line to be echoed to stdout.
32:     if line_to_be_echoed is not None:
33:         print('<p>This is the line that you entered:</p>')
34:         print('<p>' + line_to_be_echoed + '</p>')
35:     else:
36:         print('<p>You did not enter a line.</p>')
37:
38:     # Write the end of the response body to stdout.
39:     print('</body>')
40:     print('</html>')
41:
42: if __name__ == '__main__':
43:     main()

```