1. **What does socket(AF\_INET, SOCK\_STREAM) do in the code?**
   * *Answer*: It creates a new socket using the Internet address family **(AF\_INET)** for IPv4 addresses and the **SOCK\_STREAM** type for TCP connections.
2. **Why is bind(('0.0.0.0', serverPort)) used, and what does it signify?**
   * *Answer*: This command binds the server to all available interfaces on the port defined by **serverPort**. The IP address **'0.0.0.0'** allows the server to listen for incoming connections on all network interfaces.
3. **What is the purpose of listen(1) in the server code?**
   * *Answer*: It makes the server listen for incoming TCP connections. The argument **1** specifies the maximum number of queued connections (the backlog).
4. **Explain the role of the accept() method in the server loop.**
   * *Answer*: The **accept()** method waits for an incoming connection. When a client connects, it returns a new socket object representing the connection and a tuple with the client's address.
5. **What happens if the requested file does not exist on the server?**
   * *Answer*: If the file does not exist, the server responds with a **404 Not Found** error, indicating to the client that the requested resource is unavailable.
6. **How does the server respond to a valid HTTP request for a file?**
   * *Answer*: For a valid request, the server sends a **200 OK** status header followed by the content of the requested file, indicating successful retrieval.
7. **Why is it important to close the connection socket with connectionSocket.close() after serving a request?**
   * *Answer*: Closing the connection socket properly ends the connection with the client, freeing up resources and ensuring that the server can handle new connections efficiently.
8. **In the context of this script, what could cause an IOError to be raised?**

* *Answer:* An **IOError** in this script is most likely raised if the requested file does not exist or the server lacks the permissions to access it.

1. **Where in the code does the server identify the requested file and how does it read its content before sending it back to the client?**

**if len(requestParts) > 1:**

**filename = requestParts[1]**

**with open(filename[1:]) as f:**

**outputdata = f.read()**

This code segment performs two main tasks:

1. It identifies the requested file by checking if the client's HTTP request has more than one part (meaning it includes a file name). If so, it extracts the file name from the request.
2. It then reads the content of the requested file by opening it (removing the leading "/" from the filename for proper file path formatting) and reading its contents into outputdata, which will be sent back to the client.