HTTP 1xx Status Codes (Informational Responses)

The **1xx (Informational)** class of status codes indicates that the request has been received and the process is continuing. These responses are not final and are used to inform the client that the initial part of the request was received and has not yet been rejected.

🔹 100 Continue

☸️Meaning:

The server has received the request headers, and the client should proceed to send the request body.

✡️Use Case:

Used when the client wants to send a large payload. It helps avoid sending the body if the server intends to reject the request based on headers.

🔁Flow:

1. Client sends headers with `Expect: 100-continue`.
2. Server replies with `100 Continue` if everything looks good.
3. Client sends the request body.
4. Server sends the final status (e.g., 200 OK).

✅Summary:

- Saves bandwidth
- Reduces unnecessary server load
- Still one HTTP request, split into two phases (headers → body)

🔹 101 Switching Protocols

☸️Meaning:

The server agrees to switch the protocol as requested by the client.

✡️Use Case:

Used mainly during WebSocket upgrades or HTTP to another protocol switch.

🔁Flow:

1. Client sends request with headers:

2. `Upgrade: websocket`
3. `Connection: Upgrade`
4. Server responds with `101 Switching Protocols`.
5. The connection is now upgraded to the new protocol (e.g., WebSocket).

✅Summary:

- Initiated by client
- Server must explicitly agree
- Common in real-time applications (chat, games)

🔹 102 Processing (WebDAV)

☸️Meaning:

The server has received and is processing the request, but no response is available yet.

✡️Use Case:

Used in WebDAV operations for long-running tasks like copying/moving many resources.

✅Summary:

- Prevents client timeouts
- Not common in regular REST APIs
- Used mostly in file-sharing or collaborative editing systems

---

🔹 103 Early Hints

☸️Meaning:

The server is sending some headers (like preload links) before the final response is ready.

✡️Use Case:

To improve perceived performance by letting the browser preload resources early.

🔁Flow:

1. Server sends `103 Early Hints` with headers like:
2. `Link: </style.css>; rel=preload`
3. Browser starts loading resources immediately.
4. Server later sends final response (e.g., `200 OK`).

✅Safe Even If Final Response Fails:

- 103 is informational
- Final response (even if 500/404) does not invalidate early hints

---

♋Summary Table

Code	Name	Description
100	Continue	Client may send request body now
101	Switching Protocols	Server switching to requested protocol
102	Processing	Server is working on the request (WebDAV)
103	Early Hints	Sending headers early for resource preloading