**HttpContext không phải là một middleware**, mà là một **đối tượng trung tâm** đại diện cho toàn bộ **ngữ cảnh của một HTTP request**.

Tuy nhiên:

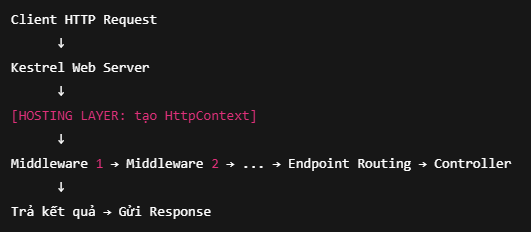
HttpContext được **tạo ra và gắn vào pipeline từ rất sớm**, trước bất kỳ middleware nào — nhờ **Kestrel server** và **ASP.NET Core hosting layer**.

**HttpContext là gì?**

HttpContext là một object chứa thông tin về:

| **Thành phần** | **Mô tả** |
| --- | --- |
| Request | URL, Headers, Query, Body... |
| Response | Headers, Body, Status code |
| User | Claims, Identity của user |
| Session, Items | Dùng để chia sẻ dữ liệu giữa middleware |
| Connection, Trace | Kết nối mạng, mã trace |

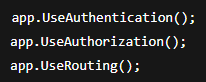
Vòng đời của HttpContext trong Pipeline:



HttpContext được truyền dọc theo request pipeline dưới dạng tham chiếu.

**Vậy Middleware là gì?**

Middleware là các **thành phần trung gian** mà bạn **cấu hình thủ công** trong Program.cs, như:

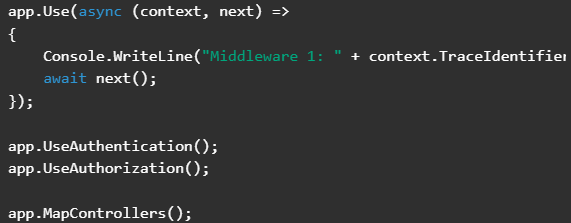


Mỗi middleware **nhận** HttpContext, có thể **đọc / chỉnh sửa / thêm thông tin** vào đó (ví dụ: HttpContext.User, HttpContext.Items["some-key"], v.v).

**Mô phỏng chi tiết**

1. **Client** gửi request HTTP tới server ASP.NET Core (qua Kestrel).
2. **Kestrel** nhận gói tin, phân tích và tạo request.
3. **Hosting Layer (ASP.NET Core host)** tạo một đối tượng HttpContext đại diện request đầu tiên.
4. HttpContext được truyền vào chuỗi middleware.
5. **Middleware** (được đăng ký trong Program.cs) nhận HttpContext, có thể thay đổi, lưu, log, xác thực,...
6. Khi qua routing đến endpoint (controller/action), controller truy cập HttpContext đã được thiết lập.
7. Response được tạo từ controller đi ngược lại qua pipeline.
8. Cuối cùng, Kestrel gửi response về cho client.

**Minh hoạ**

****

* Trong mỗi middleware, context là HttpContext vẫn chuyển.
* Trong controller: HttpContext.User, HttpContext.Request, HttpContext.Items...
* HttpContext được tạo sớm nhất ngay khi request vào pipeline.
* Mỗi request có 1 HttpContext duy nhất.
* Được truy cập thông qua DI (IHttpContextAccessor) hoặc từ ControllerBase.