**Bai 1**

Một HTTP Request (yêu cầu HTTP) là thông điệp mà trình duyệt hoặc client gửi đến server để yêu cầu tài nguyên. HTTP Request bao gồm các thành phần chính sau:

* **Method**: Xác định hành động cần thực hiện, ví dụ: GET để lấy dữ liệu, POST để gửi dữ liệu, PUT để cập nhật dữ liệu, và DELETE để xóa dữ liệu.
* **URL** (Uniform Resource Locator): Đường dẫn đến tài nguyên cụ thể trên máy chủ, ví dụ: /api/products/1.
* **Headers**: Chứa thông tin bổ sung về yêu cầu như định dạng dữ liệu (Content-Type), thông tin xác thực (Authorization), loại trình duyệt (User-Agent),...
* **Body** (thân yêu cầu): Chỉ xuất hiện trong các method như POST hoặc PUT, chứa dữ liệu được gửi đến server, thường ở dạng JSON, XML hoặc biểu mẫu.

Tất cả các thành phần này giúp server hiểu rõ yêu cầu và phản hồi chính xác cho client.

**Bai 2**

| **Mã Trạng Thái** | **Tên Mã** | **Mô Tả Ngắn Gọn** |
| --- | --- | --- |
| 200 | OK | Yêu cầu thành công, server trả về dữ liệu như mong đợi. |
| 201 | Created | Tài nguyên đã được tạo thành công. |
| 204 | No Content | Yêu cầu thành công nhưng không có nội dung trả về. |
| 400 | Bad Request | Yêu cầu không hợp lệ từ phía client. |
| 401 | Unauthorized | Chưa xác thực hoặc thông tin xác thực không hợp lệ. |
| 403 | Forbidden | Bị từ chối truy cập, không đủ quyền. |
| 404 | Not Found | Không tìm thấy tài nguyên yêu cầu. |
| 405 | Method Not Allowed | Phương thức không được hỗ trợ cho URL này. |
| 500 | Internal Server Error | Lỗi nội bộ từ phía server. |
| 503 | Service Unavailable | Dịch vụ tạm thời không khả dụng, thường do quá tải. |

**Bai 5**

**1. Thêm sách mới – POST /books**

* **Request Body**:

{

"title": "Clean Code",

"author": "Robert C. Martin",

"price": 450.0

}

* **Kết quả mong đợi**: Trả về mã trạng thái 200 OK hoặc 201 Created, body chứa thông tin sách vừa tạo, bao gồm id.
* **Kết quả thực tế**:

{

"id": 1,

"title": "Clean Code",

"author": "Robert C. Martin",

"price": 450.0

}

**Kết luận**: Thêm sách thành công.

**2. Lấy danh sách sách – GET /books**

* **Request**: Gửi yêu cầu GET đến /books.
* **Kết quả mong đợi**: Trả về mã 200 OK và danh sách tất cả các sách đã lưu.
* **Kết quả thực tế**:

[

{

"id": 1,

"title": "Clean Code",

"author": "Robert C. Martin",

"price": 450.0

}

]

**Kết luận**: Truy xuất danh sách sách thành công.

**3. Cập nhật sách – PUT /books/1**

* **Request Body**:

{

"title": "Clean Code (2nd Edition)",

"author": "Robert C. Martin",

"price": 500.0

}

* **Kết quả mong đợi**: Trả về 200 OK, body chứa thông tin sách đã cập nhật.
* **Kết quả thực tế**:

{

"id": 1,

"title": "Clean Code (2nd Edition)",

"author": "Robert C. Martin",

"price": 500.0

}

**Kết luận**: Cập nhật sách thành công.

**4. Xóa sách – DELETE /books/1**

* **Request**: Gửi DELETE đến /books/1.
* **Kết quả mong đợi**: Trả về 200 OK hoặc thông báo "Deleted successfully".
* **Kết quả thực tế**:

"Deleted successfully"

**Kết luận**: Xóa sách thành công.

**Tổng kết kiểm thử**

| **Endpoint** | **Phương thức** | **Kết quả** | **Ghi chú** |
| --- | --- | --- | --- |
| /books | POST | ✅ | Tạo mới sách thành công |
| /books | GET | ✅ | Lấy danh sách sách |
| /books/{id} | PUT | ✅ | Cập nhật sách thành công |
| /books/{id} | DELETE | ✅ | Xóa sách thành công |

**Bai 6**

**Thêm nhân viên – POST /employees**

json

{

"name": "Nguyễn Văn A",

"email": "a@example.com",

"position": "Developer",

"salary": 1500.0

}

**Kết quả**: Nhân viên được thêm thành công với id.

**Lấy danh sách – GET /employees**

**Kết quả**: Trả về mảng JSON chứa các nhân viên đã thêm.

**Cập nhật nhân viên – PUT /employees/1**

{

"name": "Nguyễn Văn B",

"email": "b@example.com",

"position": "Senior Dev",

"salary": 2000.0

}

**Kết quả**: Nhân viên id=1 được cập nhật thành công.

**Xóa nhân viên – DELETE /employees/1**

**Kết quả**: "Deleted successfully" nếu nhân viên tồn tại.

Bai 7

| **STT** | **Endpoint** | **Method** | **Chức năng** | **Request Body (nếu có)** | **Response** |
| --- | --- | --- | --- | --- | --- |
| 1 | /users | GET | Lấy danh sách người dùng | — | Danh sách người dùng |
| 2 | /users | POST | Thêm người dùng mới | { "username": "...", "email": "...", "password": "..." } | Người dùng mới được tạo |
| 3 | /users/{id} | PUT | Cập nhật người dùng theo ID | JSON giống như POST | Người dùng đã cập nhật |
| 4 | /users/{id} | DELETE | Xóa người dùng theo ID | — | "User deleted successfully" |