

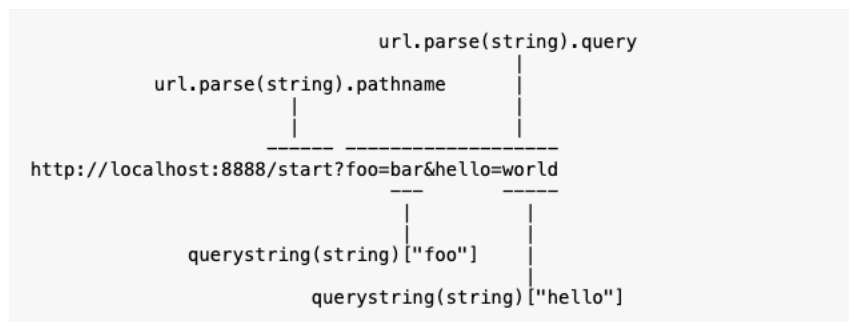
Bài thực hành – 02 – Hello ExpressJS

Mục tiêu:

- Setup môi trường lập trình ExpressJS.
- Khởi tạo một project ExpressJS đơn giản để.

Bài 1: (3 điểm)

- Tạo một project ExpressJS thực hiện công việc của một calculator đơn giản với yêu cầu sử dụng phương thức GET để thực hiện các phép tính cộng, trừ, nhân và chia 2 số nguyên hoặc số thực với các định nghĩa API sau
 - **GET** /cong?a=1&b=2 → trả về 3
 - **GET** /tru?a=4&b=2 → trả về 2
 - **GET** /nhan?a=2&b=4 → trả về 8
 - **GET** /chia?a=10&b=5 → trả về 2
 - **GET** /chia?a=10&b=0 → trả về error message “Lỗi chia 0”
 - **GET** /chiaDu?a=10&b=3 → trả về 1
 - **GET** /binhPhuong?a=2 → trả về 4
 - **GET** /giaiThua?a=5 → trả về $5! = 120$
 - **GET** /luyThua?a=2&b=3 → trả về $2^3 = 8$
 - **GET** /uocChungLonNhat?a=6&b=4 → trả về 2
 - **GET** /boiChungNhoNhat?a=6&b=4 → trả về 12
 - **GET** /laSoNguyenTo?a=6 → trả về False
 - **GET** /laSoNguyenTo?a=7 → trả về True
 - **GET** /laSoHoanHao?a=4 → trả về True
 - **GET** /laSoHoanHao?a=2 → trả về False
 - **GET** /hoanVi?a=1&b=2 → trả về đối json { a: 2, b: 1 }
- Gợi ý:
 - Các bạn có thể tái sử dụng lại bài 2 trong Lab01 để làm tiếp, tuy nhiên cần phải cấu hình lại project để sử dụng framework ExpressJS
 - Tham khảo hình sau để có thể truy xuất paramter trên queryString



Bài 2: (3 điểm)

- Tạo một project Express thực hiện quản lý thông tin của danh sách sản phẩm như sau:
 - o Toạ file data.json có cấu trúc như hình bên dưới

```
[
  {
    "id": 1,
    "name": "iPhone 12",
    "price": 1000
  },
  {
    "id": 2,
    "name": "iPad pro M1",
    "price": 1200
  },
  {
    "id": 4,
    "name": "macBook pro M1",
    "price": 2000
  },
  {
    "id": 3,
    "name": "ABC",
    "price": 1200
  },
  {
    "id": 5,
    "name": "FEG",
    "price": 2222
  }
]
```

- o Sử dụng Express để định nghĩa các cấu trúc sau:
 - **GET /products** → trả về đầy đủ danh sách các object đang có dưới dạng json
 - **GET /products?id=1** → trả về object json có id = 1
 - **GET /products/first** → trả về object json đầu tiên trong data.json
 - **GET /products/last** → trả về object json cuối cùng trong data.json
-

- **GET** */products/at?value=2* → trả về object json ở vị trí thứ 2 tổng data.json
- **GET** */products/maxPrice* → trả về object json có price lớn nhất
- **GET** */products/minPrice* → trả về object json có price nhỏ nhất
- **GET** */products* → trả về
- **DELETE** */products?id=1* → xoá object json ở có id=1
- **DELETE** */products?price=222* → xoá các object json có price = 222
- **DELETE** */products?name=ABC* → xoá các object json có name = ABC

Bài 3: (4 điểm)

- Tạo project sử dụng Framework ExpressJS kết hợp với NodeJS thực hiện quản lý File cơ bản với các yêu cầu cụ thể sau:
 - Tự tạo folder Public có cấu trúc như sau:
 - Folder **public**
 - Folder **css**
 - File *style.css*
 - Folder **js**
 - File *script.js*
 - Folder **resources**
 - Folder **data**
 - File *data01.txt*
 - File *data02.txt*
 - File *data03.txt*
 - File *README.md*
 - Tạo API thực hiện các chức năng sau:
 - **GET** */list*
 - Trả về cấu trúc toàn bộ cây thư mục trong folder public
 - **GET** */file?name=abc*
 - Trả về nội dung trong file **abc** (abc là tên file, tên file này có thể thay đổi ở tham số truyền vào API khi gọi) nếu file này tồn tại trong các folder con của folder public.
 - **GET** */folder?name=abc*
 - Trả về list các folder con của folder **abc**
 - **POST** */file?name=abc&content=Hello*
 - Tạo một file có tên là **abc** với nội dung là **Hello** (được lấy giá trị từ parameter content) và lưu file abc này trong folder **data**
 - Lưu ý
 - Parameter **name** có thể thay đổi
 - Parameter **content** có thể thay đổi được
 - **POST** */folder?name=abc*
 - Tạo folder mới có tên là abc nằm trong folder public
 - **PUT** */file?old=abc.txt&new=def.txt*
 - Đổi tên file **abc.txt** thành tên mới là **def.txt**
 - **PUT** */folder?old=abc&new=def*
 - Đổi tên folder **abc** thành tên mới **def**

- **DELETE** */file?name=abc*
 - Thực hiện xoá file có tên **abc**
- **DELETE** */folder?name=abc*
 - Thực hiện xoá folder có tên **abc**

