**NODEJS 08: SESSION - MIDDLEWARE - GROUP ROUTER**

A – LÝ THUYẾT

I – SESSION

**1. Khái niệm**

*Bản chất session là lưu một cookie có chuỗi ngẫu nhiên (được mã hóa) ở client, phía server nó sẽ lưu một bản ghi riêng với key là giá trị cookie ở client. Nhờ vậy, nó định danh được từng client và biết những session nào thuộc client nào.*

**2. Cài đặt & Cấu hình**

a. Cài đặt

|  |
| --- |
| npm install express-session --save |

b. Cấu hình

|  |
| --- |
| app.set('trust proxy', 1) // trust first proxy  app.use(session({  secret: 'keyboard cat',  resave: false,  saveUninitialized: true,  cookie: { secure: false }  })); |

**3. Cú pháp sử dụng**

|  |
| --- |
| // Khởi tạo Session  req.session.session\_name = value  // Xóa bỏ Session  req.session.destroy(); |

II – MIDDLEWARE

**1. Middleware cho request bất kỳ**

*Middleware có thể hiểu nôm na giống như một cầu nối giữa Resquest & Response (các tương tác của người dùng phía Client với xử lý logic trên Server)*

*Trong nội dung bài này thì Middleware các bạn có thể hiểu là phần trung gian giữa Router và Controller, trong đó Middleware đóng vai trò kiểm tra đầu vào từ Router để cho phép chạy Controller tương ứng.*

* Cú pháp:

|  |
| --- |
| app.use(function (req, res, next) {  console.log('Middleware')  next()  }) |

* Trong đó **next()** chính là phương thức cho phép tiếp tục thực thi hành động Request

**2. Middleware cho Router cố định**

* Cú pháp

|  |
| --- |
| app.use('/test', function (req, res, next) {  console.log('Middleware')  next()  }) |

**3. Middleware cho phương thức GET**

* Cú pháp:

|  |
| --- |
| app.get('/test', function (req, res, next) {  console.log('Middleware')  next()  }) |

**4. Sử dụng nhiều Middleware cho một Router cố định**

|  |
| --- |
| app.get('/test',  function (req, res, next) {  console.log('Middleware 1');  next();  },  function (req, res, next) {  console.log('Middleware 2');  next();  }  ); |

III – GROUP ROUTER

**1. Group Router**

a. Khái niệm

*Group Router cho phép gom nhiều Router có chung các phân đoạn*

b. Cài đặt

|  |
| --- |
| npm install express-group-routes --save |

c. Cú pháp sử dụng

|  |
| --- |
| var app = require('express');  require('express-group-routes');  app.group("/test", (router) => {  router.get("/list", function(req, res){  console.log(req.originalUrl);  });  }); |

**2. Group Router với Middleware**

* Cú pháp:

|  |
| --- |
| var app = require('express');  require('express-group-routes');    app.group((router) => {  router.use(middleware);  }); |

B – TRAINING DỰ ÁN

I – CÔNG VIỆC TRÊN LỚP

**1. Hoàn thiện chức năng Login (với việc tạo ra phiên làm việc Session)**

**2. Xây dựng chức năng Logout**

**3. Kiểm tra quyền xem nội dung với Middleware**

* Có quyền (Session) thì được truy cập các Router trong Admin
* Không quyền (Session) thì chỉ truy cập Router Login

**4. Tối ưu hóa mã nguồn**

* Nhóm Router (Group Router)
* Tạo thư viện Auth riêng để kiểm tra có quyền (check) hay không có quyền (guest)

II – CÔNG VIỆC VỀ NHÀ