# Module 1

# Ứng Dụng Quản Lý Khu nghỉ dưỡng Furama

# Mục đích

Học viên sử dụng những kiến thức đã học ở module 1 để viết ứng dụng quản lý Customer, Employees và các dịch vụ của khu nghỉ dưỡng Furama.

# Công nghệ phải dùng

Sau khi kết thúc khóa bài tập, học viên sẽ nắm được các kiến thức sau:

**Javascript**

|  |
| --- |
| Biến và Kiểu dữ liệu (variables) |
| Toán tử và Biểu thức (Operator, expression) |
| Cấu trúc điều kiện (condition structure) |
| Cấu trúc lặp (Loop) |
| Mảng (Array) |
| Hàm/Thủ tục (function/ method) |
| Lập trình hướng đối tượng (OOP) |
| Lớp (class), đối tượng (object), phương thức (method), thuộc tính (attributes), hàm tạo (function) |

# Mô tả hệ thống

Xây dựng hệ thống quản lý khu nghỉ dưỡng Furama tại thành phố Đà Nẵng. Hệ thống được mô tả như sau:

1. Khu nghỉ dưỡng Furama sẽ cung cấp các dịch vụ cho thuê Villa, House, Room, tất cả các dịch vụ này sẽ bao gồm các thông tin:

* Tên dịch vụ,
* diện tích sử dụng,
* chi phí thuê,
* Số lượng người tối đa,
* Kiểu thuê (bao gồm thuê theo năm, tháng, ngày, giờ).
* Riêng Villa sẽ có thêm thông tin: Tiêu chuẩn phòng, Mô tả tiện nghi khác, Diện tích hồ bơi, Số tầng.
* Riêng House sẽ có thêm thông tin: Tiêu chuẩn phòng, Mô tả tiện nghi khác, Số tầng.
* Riêng Phòng sẽ có thêm thông tin: Dịch vụ miễn phí đi kèm.

Ngoài ra, resort còn cung cấp các dịch vụ đi kèm như massage, karaoke, thức ăn, nước uống, thuê xe di chuyển tham quan resort.

Dịch vụ đi kèm sẽ bao gồm các thông tin: Tên dịch vụ đi kèm, Đơn vị, Giá tiền

1. Hệ thống quản lý còn cho phép quản lý thông tin của Employee resort. Thông tin Employee sẽ bao gồm: Họ tên Employee , Ngày sinh, Số CMND, Số ĐT, Email, Trình độ, Vị trí, mức lương.

Trình độ sẽ lưu trữ các thông tin: Trung cấp, Cao đẳng, Đại học và sau đại học

Vị trí sẽ lưu trữ các thông tin: Lễ tân, phục vụ, chuyên viên, giám sát, quản lý, giám đốc.

Mỗi Employee sẽ thuộc một bộ phận làm việc: Hiện tại resort bao gồm các bộ phận: Sale – Marketing, Hành Chính, Phục vụ, Quản lý.

1. Hệ thống cho phép quản lý Customer đến thuê và sử dụng các dịch vụ của Furama. Thông tin của Customer bao gồm: Họ tên Customer, Ngày sinh, Giới tính, Số CMND, Số ĐT, Email, Loại khách, Địa chỉ (resort sẽ phân loại Customer để áp dụng các chính sách phụ vụ cho phù hợp).

Loại Customer bao gồm: (Diamond, Platinium, Gold, Silver, Member).

Customer có thể sử dụng các dịch vụ thuê Villa, House, Phòng và các dịch vụ đi kèm.

Đối với các dịch vụ thuê Villa và House thì Customer cần phải làm hợp đồng thuê với Furama.

Hợp đồng thuê sẽ bao gồm các thông tin: Số hợp đồng, Ngày bắt đầu, Ngày kết thúc, Số tiền cọc trước, Tổng số tiền thanh toán.

# Yêu cầu hệ thống

## Yêu cầu 1 *(#1. Biến, kiểu dữ liệu và toán tử*)

**Objectives**

* *Tạo được các biến và kiểu dữ liệu thông dụng*
* *Biết cách sử dụng alert và prompt*
* *Sử dụng toán tử +, - , \*, /.*
* *Sử dụng hàm ép kiểu.*

Tạo project có tên là Module1\_AccountName để thực hiện các yêu cầu sau:

**Task 1:**

Cho phép người dùng nhập vào tên, số CMND, ngày tháng năm sinh, email, địa chỉ, loại Customer (Diamond, Platinum, Gold, Silver, Member), giảm giá, số lượng đi kèm, số ngày thuê, loại dịch vụ (Villa, House, Room), loại phòng thuê (Vip, Business, Normal).

(***cố gắng đặt tên biến bằng tiếng anh và có ý nghĩa ví dụ: số ngày thuê = rentDays***)

Bảng giá:

* Villa 500$/ 1 ngày
* House 300$/1 ngày
* Room: 100$/1 ngày

**Task 2:**

Hiển thị toàn bộ thông tin vừa nhập kèm theo giá sau khi đã được giảm giá

( Số tiền phải trả = giá loại dịch vụ X số ngày X ( 1 – giảm giá/100 ) )

(*gán giá trị trực tiếp cho giá loại dịch vụ từ bảng giá bên trên*)

***Có thể xem video demo để hiểu cách thực hiện***

## Yêu cầu 2 *(#2. Cấu trúc điều kiện)*

**Objectives**

* *Sử dụng được câu lệnh if, if else, else if , if lồng nhau*
* *Sử dụng được swich case*
* *Biết cách xử lý ngày tháng bằng 1 số hàm trong JS*

Để thuận tiện thao tác với các chức năng của hệ thống cần xây dựng 1 trình đơn (menu) cho phép lựa chọn các chức năng.

Hãy tạo sẵn thông tin cho 1 Customer như yêu cầu 1 bằng cách thực hiện thêm dữ liệu cho Customer đó trực tiếp trên mã nguồn.

Nếu người dùng chọn sai (*các lựa chọn không nằm trong menu*) hiển thị “Failed”

*(Lưu ý: Copy bài trước ra 1 bài mới để có thể xem lại code từng ngày dễ dàng hơn)*

**Task 1:**

**Show Information Customer.**

Hiển thị toàn bộ thông tin của Customer.

**Task2:**

**Edit Information Customer.**

Cho phép chỉnh sửa thông tin của Customer. **Sau đó hiển thị lại toàn bộ thông tin của Customer sau khi đã được chỉnh sửa**

(*Sau khi chọn edit hiển thị 1 menu con show thuộc tính và giá trị của thuộc tính cho phép người dùng lựa chọn thuộc tính muốn chỉnh sửa, nếu sự lựa chọn của người dùng không nằm trong menu con hiển thị “Failed”*)

**Task 3:**

**Show the amount to be paid after discount**

*( Số tiền phải trả = giá loại dịch vụ X số ngày X ( 1 – giảm giá/100 )*

*ở bài này giá loại dịch vụ sẽ phụ thuộc vào loại dịch vụ, giả sử gán loại dịch vụ là Villa chương trình sẽ thực hiện gán giá loại dịch vụ là 500 tương tự house 300 và room 100, hãy viết câu lệnh if để thực hiện yêu cầu này )*

Hiển thị số tiền Customer phải chi trả (được tính theo công thức bên trên). Số tiền Customer cần chi trả sẽ được tính dựa trên các điều kiện sau *(sau khi thực hiện công thức bên trên hãy xét thêm điều kiện bên dưới để đưa ra số tiền phải trả của Customer)*:

Mức giá theo địa chỉ:

* Đà Nẵng giảm giá 20$
* Huế giảm giá 10$
* Quảng Nam giảm giá 5$

Mức giá theo thời gian lưu trú:

* 7 ngày trở lên giảm giá 30$
* Từ 5 – 7 ngày giảm giá 20$
* Từ 2 – 4 ngày giảm giá 10$

Mức giá theo loại Customer:

* Nếu loại Customer là Diamond giảm giá 15$
* Nếu loại Customer là Platium giảm giá 10$
* Nếu loại Customer là Gold giảm giá 5$
* Nếu loại Customer là Silver giảm giá 2$
* Nếu loại Customer là Member thì không giảm giá

Mức giá dựa vào tuổi và địa chỉ:

* Tuổi từ 30 trở lên và địa chỉ tại Đà Nẵng thì giảm giá 2$
* Tuổi từ 20 đến 30 tuổi và địa chỉ tại Đà Nẵng thì giảm giá 1$

**Hướng dẫn:**

*Để xử lý ngày tháng để tính tuổi tham khảo mã sau:*

let currentYear = new Date().getFullYear();

                let birthday = new Date(birthdayCustomer).getFullYear();

                let ageCustomer = current - birthday;

***Có thể xem video demo để hiểu cách thực hiện***

## Yêu cầu 3 *(#3. Cấu trúc lặp)*

**Objectives**

* *Sử dụng câu lệnh lặp do while, while, for*
* *Sử dụng Break, Continue*
* *Sử dụng hàm isNaN, Number.isInteger trong JS*
* *Sử dụng 1 số hàm xử lý chuỗi subString, toLowerCase, toUpperCase, trim.*

*(Lưu ý: Copy bài* ***yêu cầu 1*** *để có thể xem lại code từng ngày dễ dàng hơn)*

*(Ở bài này chúng ta sử dụng copy lại mã nguồn ở yêu cầu 1 để thực hiện task sau đây)*

**Task 1:**

Để kiểm tra tính hợp lệ dữ liệu do người dùng nhập vào thì hệ thống cần xét các trường hợp sau: Áp dụng lệnh while hoặc do-while để yêu cầu người dùng phải nhập lại khi nhập dữ liệu sai định dạng.

* (\*) Địa chỉ email phải đúng theo định dạng [abc@abc.abc](mailto:abc@abc.abc)
* Số CMND phải đúng theo định dạng XXXXXXXX với X là số nguyên dương
* (\*) Ngày sinh phải đúng theo định dạng DD/MM/YYYY

(*Sử dụng boolean, for và do while để duyệt và kiểm tra tính hợp lệ cho trường hợp này*)

* Giảm giá, số ngày thuê và người đi kèm phải là số nguyên dương

*Ví dụ*: Khi nhập giá trị cho thuộc tính có kiểu dữ liệu là số nguyên thì người dùng phải nhập dữ liệu phải là kiểu số thì hệ thống mới cho phép nhập tiếp các thuộc tính khác hoặc tiến hành các bước tiếp theo.

**Task 2:**

Áp dụng vòng lặp for để xử lý xóa bỏ tất cả khoảng trắng thừa trong tên của Customer, địa chỉ, loại khách hàng và chuẩn hóa dữ liệu.

(*Giả sử nhập tên ngUYEn thANH KIEN hãy xử lý sao cho kết quả hiển thị thông tin sau khi nhập ở mục Name Customer là Nguyen Thanh Kien*)

***Có thể xem video demo để hiểu cách thực hiện***

## Yêu cầu 4 *(#4. Mảng)*

**Objectives**

* *Sử dụng được mảng 1 chiều thành thạo*
* *Sử dụng được vòng lặp để duyệt mảng*
* *Sử dụng được các thao tác cơ bản với mảng*
* *Biết thêm một số kĩ thuật xử lý mảng*

***(Lưu ý: Ở bài này không cần copy lại mã nguồn cũ, tạo ra file YeuCau4.html và tiến hành thực hiện các task sau)***

Hệ thống cần lưu được thông tin của nhiều Customer. Áp dụng mảng để thực hiện lưu thông tin nhiều Customer.

**Task 1:**

1. Tạo menu gồm có tính năng Add New Customer và Delete Customer

(*sau khi người dùng chọn Add New Customer hãy hiển thị ô nhập số lượng customer muốn thêm vào sau đó tiến hành nhập tên customer và hiển thị ra toàn bộ tên của customer có trong mảng arrNameCustomers*)

1. Xuất ra mảng arrNameCustomers được sắp xếp theo thứ tự alphabel

(*Sau khi hiển thị ra toàn bộ tên của Customer có trong mảng ở bên trên tiếp ngay sau đó hãy hiển thị ra yêu cầu này*)

**Task 2:**

Hãy thêm trực tiếp vào mã nguồn để có 1 mảng gồm 10 phần tử là tên của Customer. Sau đó từ Menu hãy chọn Delete Customer và hiển thị ra menu con chứa danh sách tên của các Customer, chọn 1 Customer muốn xóa và in ra tên của toàn bộ các Customer sau khi xóa.

***Có thể xem video demo để hiểu cách thực hiện***

## Yêu cầu 5 *(#5. Hàm)*

**Objectives**

* *Khai báo và sử dụng cơ bản về hàm*
* *Khai báo và sử dụng được hàm có tham số*
* *Hiểu biết và sử dụng 1 số kĩ thuật về hàm*
* *Xây dựng tư duy lập trình và tư duy logic về luồng dữ liệu*
* *(\*) Hiểu được tầm quan trọng về hiệu ứng domino giữa các tính năng (Important)*

*(Lưu ý: Ở bài này chúng ta không copy lại mã nguồn cũ mà sẽ tổng hợp kiến thức và mã nguồn của các yêu cầu trước để hoàn thành các task bên dưới)*

**Task 1:** Thực hiện các yêu cầu sau:

1. Xây dựng function tên displayMainMenu() hiển thị menu để người dùng lựa chọn các tính năng

* Add New Customer.
* Display Information Customer.
* Edit Information Customer.
* Delete Customer.
* Exit.

1. Xây dựng hàm addNewCustomer() để thực hiện thêm thông tin của 1 khách hàng với các thuộc tính như yêu cầu 1 vào một mảng có tên là arrListCustomers. (*Sau khi thực hiện xong trả về Main Menu*)
2. Xây dựng hàm displayCustomers() (*Sau khi chọn vào đây hãy hiển thị 1 menu con chứa danh sách toàn bộ tên và số CMND của các Customer* ***và 1 lựa chọn back.*** *Nếu người dùng chọn Customer nào thì hiển thị toàn bộ thông tin của Customer đó thông qua hàm được gợi ý bên dưới, nếu chọn back thì quay về menu trước đó*). Sau khi thực hiện xong trả về Main Menu.

**Note:** Ở bài này hãy xây dựng 1 hàm displayInfomationCustomer() để hiển thị toàn bộ thông tin của 1 khách hàng bất kì sau đó sử dụng hàm này vào bên trong của hàm displayCustomers().

**Task 2:**

1. Định nghĩa function có tên là editCustomer để chỉnh sửa thông tin cho Customer, sau khi chỉnh sửa hiển thị lại thông tin của Customer đó sau đó trả về **main menu**

(*sau khi chọn edit hiển thị 1 menu con show toàn bộ tên Customer và IdCard cho phép người dùng chọn 1 Customer tiếp theo hiển thị 1 menu show thuộc tính và giá trị của thuộc tính cho phép người dùng lựa chọn thuộc tính muốn chỉnh sửa*)

**Note:** Trong các menu con luôn có 1 lựa chọn back để quay về menu gần nhất trước đó. Tận dụng lại hàm displayCustomers() ở bên trên.

1. Tương tự cho phần Delete Customer (*làm như phần note của editCustomer*)

**Note:** Sau khi chọn Customer muốn xóa hãy hiển thị thông báo yêu cầu xác nhận 1. Yes 2. No nếu chọn 1 sẽ tiến hàng xóa và trả về Main Menu, nếu chọn 2 sẽ không tiến hành xóa và trả về Main Menu

***Có thể xem video demo để hiểu cách thực hiện***

## Yêu cầu 6 *(#6.Hướng đối tượng)*

**Objectives**

* *Khai báo và sử dụng được lớp*
* *Khai báo và sử dụng được đối tượng*
* *Xây dựng và sử dụng được phương thức và thuộc tính của đối tượng*
* *Biết cách sử dụng Regular Expression*

*(Lưu ý: Tạo file YeuCau6.html. Copy những menu của bài trước và sửa lại để hoàn thành các task sau)*

**Task 1:**

Thực hiện các yêu cầu sau:

1. Tạo đối tượng có tên là Customer bao gồm các thuộc tính ở phần mô tả như yêu cầu 1
2. Tiếp tục khởi tạo các phương thức getter/setter cho các thuộc tính của Customer
3. Định nghĩa thêm phương thức để tính tiền phòng cho các Customer theo công

thức ở Yêu Cầu 1.

1. Định nghĩa lại hàm addNewCustomer để thêm mới Customer vào mảng có tên là listCustomers.

Chức năng thêm mới Customer sẽ cho phép người sử dụng tạo mới thông tin cho 1 Customer từ trình đơn (menu) đã định nghĩa từ Yêu cầu 1.

Sử dụng regular expression để bắt buộc người dùng phải nhập đúng Email, Birthday.

1. Chạy chương trình với menu tương tự như yêu cầu 5 và các chức năng hoạt động giống yêu cầu 5 bao gồm :

* Add new customer
* Display information customer
* **Display total pay of customer** *(hiển thị danh sách khách hàng cho phép người dùng chọn vào khách hàng nào muốn hiển thị tổng tiền cần thanh toán)*
* Edit information customer
* Delete customer
* Exit

**Task 2:**

Thêm vào menu chính ở Task 1 bên trên để thực hiện các yêu cầu sau:

1. Tạo đối tượng có tên là Employee bao gồm các thuộc tính ở phần mô tả như yêu cầu 1
2. Tiếp tục khởi tạo các phương thức getter/setter cho các thuộc tính của Employee
3. Định nghĩa thêm phương thức tính tiền phòng cho các Employee
4. Định nghĩa lại hàm addNewEmployee để thêm mới Employee vào mảng có tên là arr Employees.

Chức năng thêm mới Employee sẽ cho phép người sử dụng tạo mới thông tin cho 1 Employee từ trình đơn (menu) đã định nghĩa từ Yêu cầu 1.

Xây dựng phương thức tính lương cho từng Employee theo điều kiện sau đây

* Trợ cấp 500$ cho Employee nào làm ở vị trí Manager
* Trợ cấp 300$ cho Employee nào làm ở vị trí Sale
* Trợ cấp 200$ cho Employee làm việc ở vị trí Marketing

**Task 3:**

Định nghĩa lại function addNewCustomer() có kiểm trả tính hợp lệ của dữ liệu trước khi thực hiện thêm mới theo những yêu cầu sau đây:

* Email phải có định dạng [abc@abc.com](mailto:abc@abc.com)
* Ngày tháng năm sinh phải đúng chuẩn dd/MM/yyyy
* Những thuộc tính là số thì không cho phép nhập vào kí tự
* Số chứng minh nhân dân phải có đúng số và không được nhập kí tự

***Có thể xem video demo để hiểu cách thực hiện***