**Introduction to the RPA Developer Role**

|  |  |
| --- | --- |
| Ý chính | Nội dung chi tiết |
| Variables & Arguments | * Tên: tên mang tính mô tả càng tốt để các lập trình viên khác có thể đọc được * Loại dữ liệu được lưu trữ trong biến. Kiểu dữ liệu được khai báo (declared) * Giá trị mặc định: Biến sẽ thay đổi liên tục trong quá trình. Nếu nó không được đặt kiểu dữ liệu thì sẽ giữ giá trị mặc định của kiểu dữ liệu đó * Phạm vi: biến cục bộ và biến toàn cục |
| Cách tạo biến | Có 3 cách tạo biến:   * Từ variables panel * Từ the designer panel * Từ properties panel |
| Quy tắc đặt tên | Camel case |
| Một số loại dữ liệu | * Numberic: bao gồm kiểu số nguyên (int32,long(int64)) và kiểu số thực (double) * Boolean: true or false * Date and time:   + System.DateTime  + System.TimeSpan (biến đo thời gian bắt đầu và kết thúc)   * String: * Collection: Xử lý các dữ liệu phức tạp   + Array - ArrayOf<T> hoặc System.DataType[] . Kích thước cố định khi tạo ra. Chứa kiểu dữ liệu giống nhau  + List – System.Collections.Genegric.List <T>. Kích thước được co giãn. Chứa kiểu dữ liệu giống nhau  + Distionary – System.Collections.Generic.Dictionary <Tkey,Tvalue> mỗi phần tử có thể chứa 2 kiểu dữ liệu khác nhau   * GenericValue: Có thể chứa bất kì kiểu dữ liệu nào khác nhau |
| Một số tình huống nghiệp vụ xử dụng array | * Khi ta muốn lưu tên các tháng vào cùng một biến * Khi một khoản thu cố định các tài khoản ngân hàng phải được lữu trữ và sử dụng trong quá trình thanh toán * Khi các hóa đơn đã thanh toán trong tháng trước cần được xử lý * Khi tên của các nhân viên trong một đơn vị phải được xác minh trong cơ sở dữ liệu |
| Array | * Khởi tạo: stringArray= {} * Thay đổi: stringArray(index)=”” * In ra:  **'string.Join' (“” stringArray)** |
| Khi nào thì dụng biến generic | * Dữ liệu được trích xuất từ giao diện người dùng và được chuyển tiếp bộ phận khác mà không cần xử lý * Hai phiên bản của cùng một tệp Excel đang được so sánh từng cột. Các cột khác nhau về kiểu dữ liệu, điều duy nhất có liên quan là mục nào có thay đổi |
| Các loại workflow | * Sequence: Các hoạt động theo trình tự. Dễ đọc và dễ bảo trì. Khuyên dùng cho các quy trình đơn giản, tuyến tính * Flowchart: Các hoạt động lẻ khó đọc và chỉnh sửa. Nhưng dòng chảy giữa chúng rõ ràng. Phân nhánh phù hợp trong các tình huống. Giải quyết các quy trình phức tạp * State Machine * Global Exception Handler |
| Câu lệnh if | * Kiểm tra trạng thái thanh toán (ok/not ok) và thực hiện các hoạt động trong từng trường hợp * Kiểm tra số dư tài khoản để đảm bảo rằng đủ tiền thanh toán hay không |
| Vòng lặp | * Chú ý ứng dụng for each cho việc thay thế tên các file có trong một thư mục |
| Switch case | * Hóa đơn có 3 trạng thái: chưa bắt đầu, đang chờ xử lý, đã phê duyệt. mỗi trạng thái tương ứng với một hành động * Ứng dụng cho việc di chuyển các file lộn xộn đến thư mục đúng của nó |
|  |  |
|  |  |
|  |  |
|  |  |

**Tổng kết**:

……………...…………………………...…………………………...…………………………...…………………………...…………………………...…………………………...…………………………...…………………………...…………………………...…………………………...………………………..