**PHÍM TẮT**

**CTRL + F5**: Chạy project

**F5:** Chạy project trong chế độ debug

**CTRL + SHIFT + A:** Chọn vào thư mục và bấm sẽ hiển thị những mục tạo mới như class, …..

**Prop TAB TAB:** Tạo một thuộc tính public mới (chứa cả phương thức get, set)

**Region + TAB TAB:** Tạo region

**///**: Comment method

**CTRL + K + C:** Comment một dòng

**CTRL + K + U:** Uncomment một dòng

**CTRL + SHIFT + /:** Comment nhiều dòng

**Ctrl + .** : Suggest using

**Propf:** rend cả field và property

**Ctor:** tạo hàm khởi tạo

**:**  Kế thừa bằng cú pháp hai chấm



**CTRL + TAB:** Chuyển Tab

**CTRL + SHIFT + B:** Build Solution

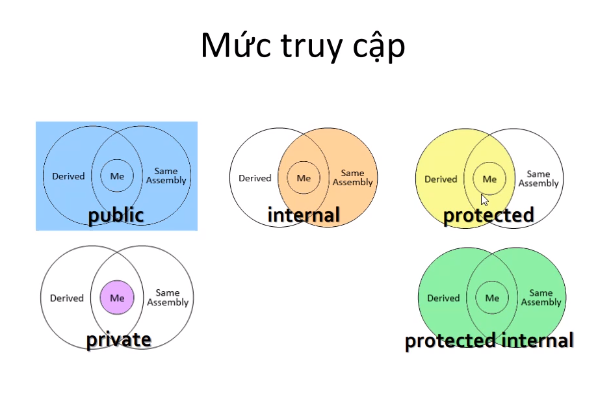
**CTRL + R + R:** đổi tên biến

**F9:** đặt breakpoint

**CW:** Console Writeline

**NOTES**

PHẠM VI TRUY CẬP



Khi không tạo phạm vi truy cập thì mặc định sẽ là **INTERNAL** (Chỉ tương tác trong cùng một **namespace**)

Mỗi class nên ở một file khác nhau

**Field** không thể can thiệp trong việc hạn chế quyền truy cập nên phải sử dụng **Property**

Kiểu dữ liệu Guid dùng cho Id => Guid? Có thể nhận null

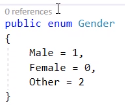
Khai báo hằng bằng từ khoá **const** (cần giá trị khởi tạo)

Với từ khoá **readonly** (không cần giá trị khởi tạo) thì chỉ set được giá trị ở hàm khởi tạo

**RESOURCES:** dùng cho việc đa ngôn ngữ, dễ dàng thay đổi được (Phải sử dụng resources cho đúng convention)

**Switch –case** (so sánh trực tiếp) **case < 20** v.v

**Enum**





Enum có thể convert về int bằng **(int)employee.Gender**

**STATIC:** Được tạo ngay khi build chương trình

Sử dụng đơn giản màk hông cần khởi tạo đối tượng chỉ cần gọi lớp và truy cập

**GENERIC:**

Viết phương thức cho những đối tượng mà không cần biết kiểu (Có thể truyền kiểu vào)



Có thể truyền qua class (các phương thức bên trong sẽ tự hiểu)

**CONVENTIONS**

**Xem ở file conventions C#**

**Field Public –** FullName

**Field Private -** fullName