1-

Dans la class Passenger:  
**○ La propriété PassportNumber doit être:  
■ la clé primaire  
■ un champ de 7 caractères**

[Key]

[StringLength(7)]

public string PassportNumber { get; set; }

2-

La propriété FirstName doit être :  
■ **de longueur minimale 3 caractères  
■ de longueur maximale 25 caractères  
■ un message d'erreur doit être affiché si les règles ne sont pas respectées**

[StringLength(25, MinimumLength = 3, ErrorMessage = "Le prénom doit comporter entre 3 et 25 caractères.")]

public string FirstName { get; set; }

**3-La propriété BirthDate doit être:  
■ affichée “Date of Birth”  
■ une date valide**

[DataType(DataType.Date)]

[Display(Name ="Date Of Birth")]

**La propriété EmailAddress doit être:  
■ une adresse mail valide**

[EmailAddress(ErrorMessage = "Veuillez saisir une adresse e-mail valide.")] public string EmailAddress { get; set; } // Validation de l'adresse e-mail

**○ La propriété TelNumber doit être :  
■ un numéro de téléphone contenant 8 chiffres**

[RegularExpression(@"^\d{8}$", ErrorMessage = "Le numéro de téléphone doit contenir exactement 8 chiffres.")]

public string TelNumber { get; set; } // Validation du numéro de téléphone

**La propriété Capacity doit être: ■ un entier positif**

[Range(1, int.MaxValue, ErrorMessage = "La capacité doit être un entier positif.")]

public int Capacity { get; set; } // Validation pour les entiers positifs

**Dans la class Flight:  
○ La propriété PlaneId doit être  
■ La clé étrangère correspondante à l’entité Plane**

[ForeignKey("Plane")]

public int PlaneFk { get; set; }

**d. Dans la class Staff :  
○ La propriété Salary doit être:  
■ une valeur monétaire**

[DataType(DataType.Currency, ErrorMessage = "Le format de la valeur salariale n'est pas valide.")] [Range(0, double.MaxValue, ErrorMessage = "Le salaire doit être une valeur monétaire positive.")]