# **Contraintes**

## **Description des entités**

|  |  |
| --- | --- |
| *Nom* | *Description* |
| Voiture |  |
| Notoriété |  |
| Bureau/Dépot |  |
| Client |  |
| Réservation |  |
| Forfait |  |
| Ville |  |
| Pays |  |
| Prix |  |

## **Règles de structure**

**Notoriété**

|  |  |
| --- | --- |
| int | Une notoriété a un Id unique |
|  | DB > Table Notoriete > Key > PK\_Notoriete : Clé primaire sur IDNotoriete |
| Nvarchar(50) | Une notoriété a un libelle |
|  |  |
| Decimal(4,2) | Une notoriété a un coéfficient multiplicateur afin de pouvoir calculer le prix |
|  |  |
| bit | Une notoriété a un champ ‘Inactif’ qui signifie si cette notoriété est toujours utilisée ou pas. |
|  |  |
|  | Une notoriété peut être reliée a 0 à plusieurs voitures |
|  |  |

**Voiture**

|  |  |
| --- | --- |
| int | Une voiture a un Id unique |
|  | DB > Table Voiture > Clé > PK\_Voiture > Clé primaire sur IDVoiture |
| Nvarchar(50) | ~~Une voiture doit avoir une marque~~ Une voiture a une marque |
|  |  |
|  | Une voiture doit être contenue dans un et un seul dépôt à la fois (à un moment donné) |
|  | DB > Table Voiture > Clé > FK\_Voiture\_Depot> Clé étrangère sur IDDepot, pointe sur IDDepot de la table Depot |
| Nvarchar(10) | Une voiture doit avoir une plaque d’immatriculation unique |
|  | DB > Table Voiture > Clé > UK\_Immatriculation > Clé unique sur Immatriculation |
|  | Une voiture posséde une et une seul notoriété |
|  | DB > Table Voiture > Clé > FK\_Voiture\_Notoriete > Clé étrangère sur IDNotoriete, pointe sur IDNotoriete de la table Notoriete |
| bit | Une voiture a un champs ‘Inactif’ pour savoir si elle est toujours utiliser par la société ou pas |
|  |  |
|  | Une voiture doit être assignée à de 0 à 1 réservation (sur une période donnée) |
|  |  |

**Dépôt**

|  |  |
| --- | --- |
| int | Un Dépôt à un ID Unique |
|  | DB > Table Depot > Clé > PK\_Depot > Clé primaire sur IDDepot |
|  | Un dépôt peut contenir de 0 à plusieurs voitures |
|  |  |
| int | ~~Un Dépôt à un ID ville qui spécifie la ville auquel il appartient~~ Un dépôt appartient à une ville |
|  |  |
| bit | Un Dépôt a un statut ‘Inactif’ qui signifie si il est toujours en activité ou pas |
|  |  |
|  | Un Dépôt n’est relié qu’a une et une seul ville |
|  |  |
|  |  |
|  |  |

**Client**

|  |  |
| --- | --- |
| int | Un Client a un Id unique |
|  |  |
| Nvarchar(50) | Un client a un nom |
|  |  |
| Nvarchar(50) | Un client a un prénom |
|  |  |
| Nvarchar(50) | Un client a une adresse mail |
|  |  |
|  | Un client peut avoir 0 à plusieurs réservations |
|  |  |

**Réservation**

|  |  |
| --- | --- |
|  | Une réservation ne concerne qu’un et un seul client |
|  |  |
|  |  |
|  |  |
|  | Une réservation doit avoir un et un seul dépôt de départ |
|  |  |
|  | Une réservation peut avoir de 0 à 1 dépôt d’arrivée lors de la réservation. Une réservation doit avoir un et un seul dépôt d’arrivée lors du retour du véhicule |
|  |  |
|  | Une réservation peut avoir de 0 à 1 Forfait (voir régles de validation =>…) |
|  |  |
| Datetime | Une réservation à une de date de Reservation |
|  |  |
| Datetime | Une réservation a une date de début |
|  |  |
| Datetime | Une réservation a une date de retour (Champ peut être nul tant que la réservation est en cours) |
|  |  |
|  | (Une réservation a une information si elle est déjà payée) |
|  |  |
|  | Une réservation ne concerne qu’un et un seul véhicule à la fois |
|  |  |
| int | Une réservation posssède un kilométrage du véhicule au départ, enregistré au départ de la période de location |
|  |  |
| int | Une réservation possède un kilométrage du véhicule à l’arrivée, enregistré au retrour du la location |
|  |  |

**Forfait**

|  |  |
| --- | --- |
| int | Un forfait a un ID unique |
|  |  |
| int | Un forfait à un IDDepot de départ |
|  |  |
| int | Un forfait à un IDDepot de fin |
|  |  |
| datetime | Un Forfait doit avoir une date de début. |
|  |  |
| datetime | Un forfait peut avoir une date de fin |
|  |  |
| Decimal(8,2) | Un forfait à un Prix de base |
|  |  |

**Ville**

|  |  |
| --- | --- |
|  | Une ville peut contenir de 0 à 1 Dépot |
|  |  |
| int | Une ville a un ID unique |
|  |  |
| int | Une ville à un IDPays qui référence le pays auquel elle appartient |
|  |  |
| Nvarchar(50) | Une ville a un nom |
|  |  |
|  | Une ville appartient un et un seul pays |
|  |  |

**Pays**

|  |  |
| --- | --- |
|  | Un pays peut contenir de 0 à plusieurs villes |
|  |  |
| int | Un pays a un ID unique |
|  |  |
| int | Un pays a un IDPrix qui référence le prix au km effectif dans le pays |
|  |  |
| Nvarchar(50) | Un pays a un nom |
|  |  |
|  | Un pays posséde un et un seul prix au km effectif à la fois |
|  |  |

**Prix**

|  |  |
| --- | --- |
| int | Un Prix à un Id Unique |
|  |  |
| Datetime | Un Prix doit avoir une date de début. |
|  |  |
| Datetime | Un prix peut avoir une date de fin. |
|  |  |
| DEcimal(8,2) | Un prix posséde une valeur |
|  |  |
|  | Un prix peut appartenir à 0 ou plusieurs pays |
|  |  |

## **Règles de validation**

|  |  |
| --- | --- |
|  | Le choix de formule « prix au km » est obligatoire si pas de forfait existant entre deux dépots |
|  |  |
|  | Le dépot de départ et le dépôt d’arrivée choisis doivent être différents si Forfait |
|  |  |
|  | Un forfait est obligatoirement lié à deux dépots différents |
|  |  |
|  | Une voiture ne peut pas être réservée si une réservation concerne déjà cette voiture pour la période |
|  |  |
|  | Un Client ne peut pas effectuer deux réservations pour des périodes qui se chevauchent |
|  |  |
|  |  |
|  |  |

## **Règles de calcul**

|  |  |
| --- | --- |
|  | Prix = … |
|  |  |
|  |  |
|  |  |
|  | Si Dépot de retour choisi lors de la réservation et correspond au dépôt de retour effectif => Réduction de 5 % sur le prix total |
|  |  |
|  | Si Dépot de retour choisi lors de la réservation et ne correspond pas au dépôt de retour effectif => Pénalité de 10% sur le prix total |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## **Autres règles**

|  |  |
| --- | --- |
|  | Un seul prix/km peut être actif à la fois. |
|  | Le delete d’un objet de la classe voiture rend inactif celle-ci |
|  | Le delete d’un objet de la classe Depot rend inactif celui-ci |
|  | Le delete d’un objet de la classe Prix rend inactif celui-ci |
|  |  |
|  |  |
|  |  |
|  |  |