**Design databases**

1. **Identifying Entities**

* Dans le développement d’un projet, la première chose à faire est de savoir les acteurs qui vont participer dans ce projet, et plus précisément, les entités qui vont être actif.

**Exemple :**

On va définir les acteurs pour un projet de site web d’un magasin ; on doit recueillir les informations avec laquelle on va travailler.

* Dans un magasin, on vend des produits aux clients
* Le magasin est une localisation
* Vendre est un événement
* Produit sont des objets
* Les clients sont des humains

Toutes ces entités formeront le besoin qui va être inclus dans la base de données.

Mais quels sont les autres informations qui se produisent lors de la vente d’un produit ? Tout ceci doit être englobé afin de compléter notre conception.



1. **Identifying Relationships**

* La prochaine étape maintenant est de déterminer la relation entre les différentes entités acteur de ce système. la relation est qu’est ce qu’une entité fait avec une autre.

🡺 Donc faut écrire les scénarios qui se passent entre les différentes entités entre eux.  
🡺 Les cardinalités sont combien dans un coté de la relation apporte a l’autre coté de la relation.

**Exemple :**

Combien de clients dépendent d’une seule vente ? Combien de ventes dépendent d’un seul client ? Combien de ventes dépendent d’un seul magasin ?

On aura comme ceci :

* **Customers --> Sales;** 1 customer can buy something several times
* **Sales --> Customers**; 1 sale is always made by 1 customer at the time
* **Customers --> Products;** 1 customer can buy multiple products
* **Products --> Customers;** 1 product can be purchased by multiple customers
* **Customers --> Shops**; 1 customer can purchase in multiple shops
* **Shops --> Customers**, 1 shop can receive multiple customers
* **Shops --> Products**; in 1 shop there are multiple products
* **Products --> Shops;** 1 product (type) can be sold in multiple shops
* **Shops --> Sales**; in 1 shop multiple sales can me made
* **Sales --> Shops;** 1 sale can only be made in 1 shop at the time
* **Products --> Sales**; 1 product (type) can be purchased in multiple sales
* **Sales --> Products;** 1 sale can exist out of multiple products

**Est ce qu’on a mentionnée toutes les relations ?** Au faite Non ! car une relation réfère à une règle de gestion, donc tout dépend de ce que l’on souhaite réaliser, on pourra l’insérer dans notre schéma.

**Comment faciliter la schématisation ?** On va mettre les cardinalités par chaque relation  
🡺**Customers --> Sales;** 1 customer can buy something several times  
🡺**Sales --> Customers;** 1 sale is always made by 1 customer at the time

So we’ll get:

* **Customers --> Sales**; --> 1:N
* **Customers --> Products**; --> M:N
* **Customers --> Shops**; --> M:N
* **Sales --> Products**; --> M:N
* **Shops --> Sales**; --> 1:N
* **Shops --> Products**; --> M:N

