

# SI Cours 5 - Modélisation Entité Association

## 1- Modélisation Entité Association (Entity Relationship)

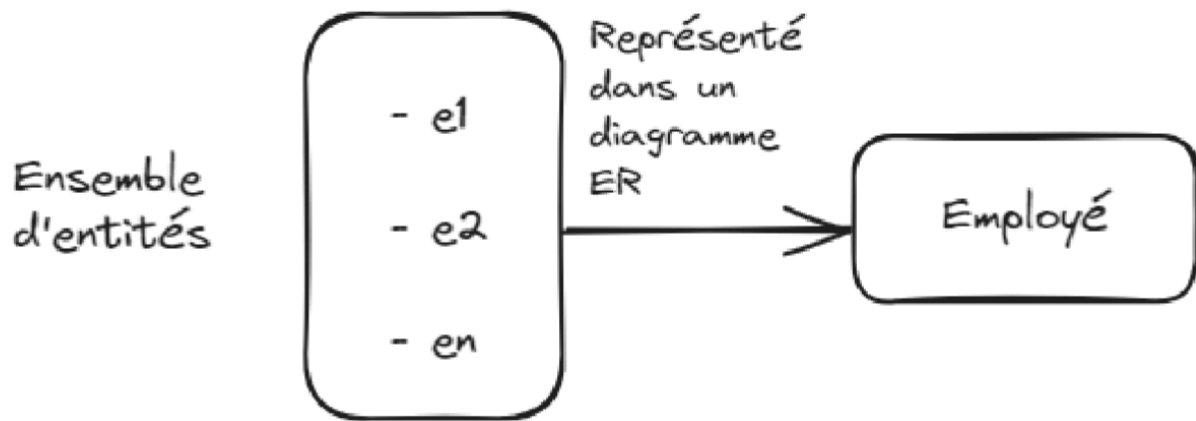
### 1-1 Modèle Entité Association

- Les modèles ER, également appelés schéma ER, sont représentés par des diagrammes ER.
- La modélisation ER est basée sur deux concepts :
  - Entités, définies comme des tables qui contiennent des informations spécifiques (données)
  - associations, définies comme les associations ou interactions entre les entités

### 1-2 Entités:

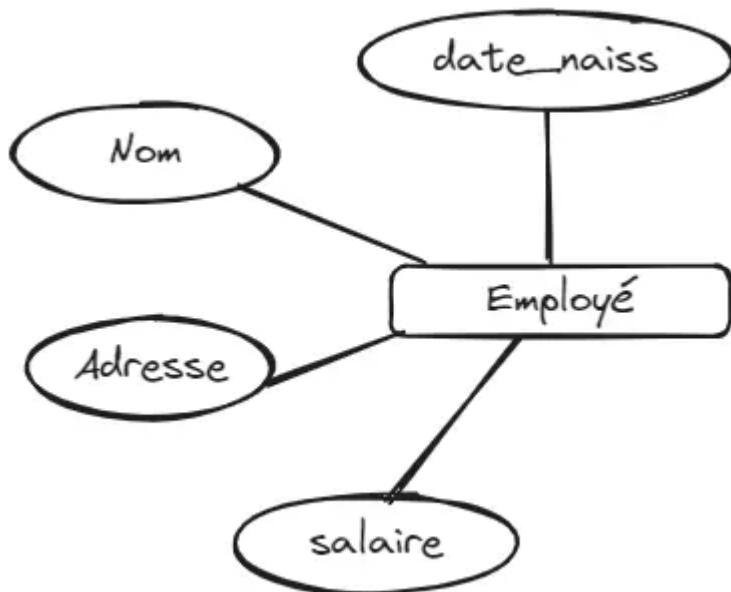
Nous utiliserons la base de données COMPAGNIE tout au long de ce chapitre pour illustrer les concepts du modèle Entité-Relation (ER). Cette base de données suit les employés, les services et les projets.

- Points clés
  - Départements : Chaque département possède un identifiant unique, un nom, une adresse et un responsable désigné.
  - Projets : Chaque service gère plusieurs projets, chacun doté d'un nom, d'un numéro et d'un budget unique.
  - Employés : Chaque employé possède un nom, un numéro d'identification, une adresse, un salaire et une date de naissance. Ils appartiennent à un département, ont un responsable direct et peuvent travailler sur plusieurs projets (la date de début de chaque projet étant enregistrée).
  - Dépendants : Nous suivons les personnes à charge de chaque employé, notamment leur nom, leur date de naissance et leur lien de parenté avec l'employé.
- Une entité est un objet du monde réel ayant une existence indépendante qui peut être différencié des autres objets.
- Elle peut représenter:
  - un objet avec une existence physique réelle (Etudiant)
  - un objet avec une existence conceptuelle (Cours)
- Un type d'entité définit une collection d'entités similaires.
- Un ensemble d'entités est une collection d'entités d'un type d'entité à un moment donné.
- Dans un diagramme entité-association (ERD), un type d'entité est représenté par un nom dans une boîte.



## 1-3 Attributs

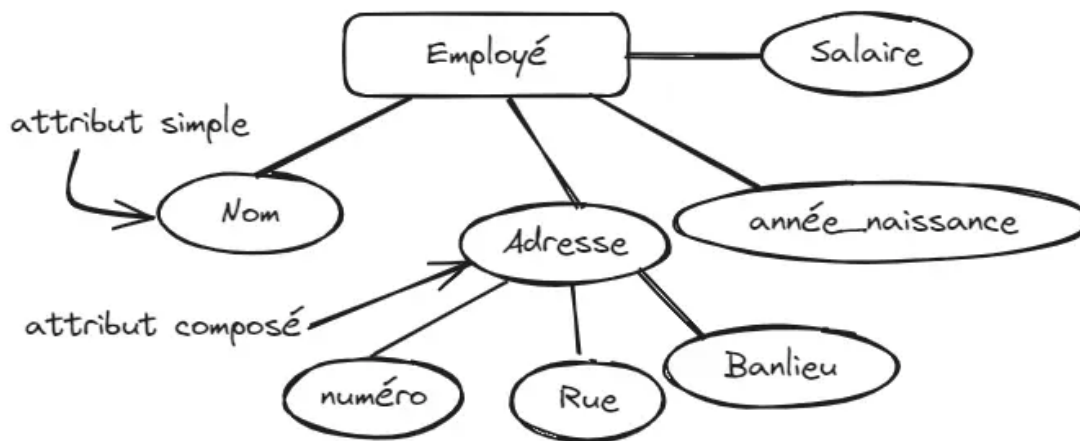
- Chaque entité est décrite par un ensemble d'attributs (par exemple, Employé = (Nom, Adresse, Date de naissance (Âge), Salaire)).
- Chaque attribut porte un nom et est associé à une entité et à un domaine de valeurs autorisées. Cependant, les informations sur le domaine attributaire ne sont pas présentées sur l'ERD.



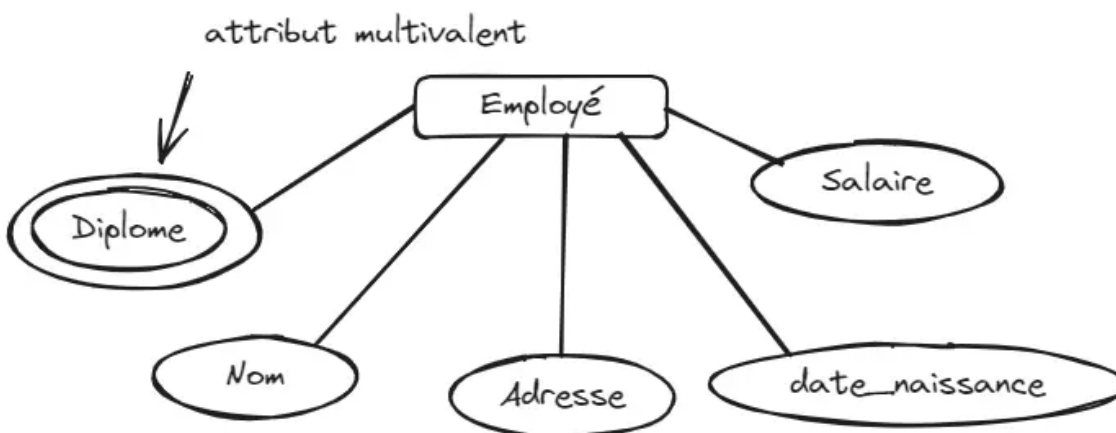
### 1-3-1 Types d'attributs

Il existe quelques types d'attributs que vous devez connaître:

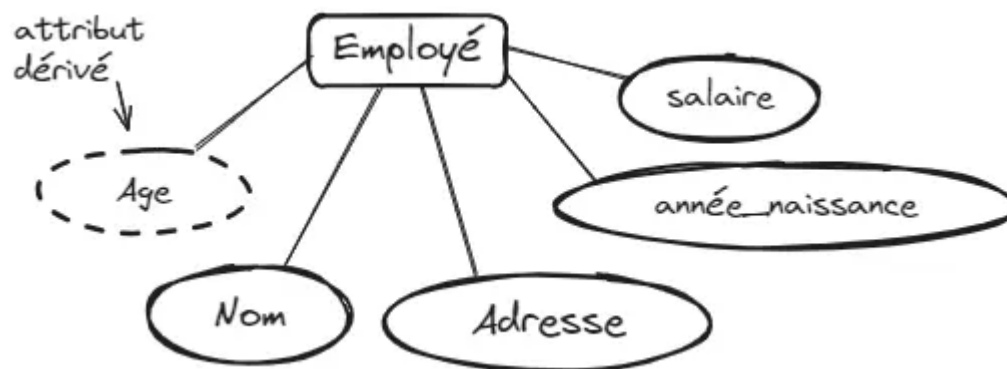
- **Attributs simples:** Ce sont des attributs tirés des domaines de valeurs atomiques; ils sont également appelés attributs à valeur unique. Dans la base de données de l'entreprise, un exemple de ceci serait: Nom={John}; Age={23}.
- **Attributs composites:** Ce sont des attributs qui se composent d'une hiérarchie d'attributs.
- Dans de la figure ci-dessous, l'adresse peut se composer du numéro, de la rue et de la banlieue. Cela s'écrirait donc comme ceci → Adresse={59 + 'Meek Street' + 'Kingsford'}.



- **Attributs multivalents:** Les attributs multivalents sont des attributs qui ont un ensemble de valeurs définies pour chaque entité. Un exemple d'attribut multivalent dans la base de données de la figure suivante, sont les diplômes d'un employé: BSc, MIT, PhD.



- **Attributs dérivés:** Les attributs dérivés sont des attributs qui contiennent des valeurs calculées à partir d'autres attributs. Un exemple de ceci peut être vu à la figure suivante. L'âge peut être dérivé de l'attribut Date de naissance. Dans ce cas, la date de naissance est appelée attribut stocké, qui est physiquement enregistré dans la base de données.



## 1-4 Clés

Une contrainte importante sur une entité est la clé. La clé est un attribut ou un groupe d'attributs dont les valeurs peuvent être utilisées pour identifier de manière unique une entité individuelle dans un ensemble d'entités.

### 1-4-1 Types de clés

Il existe plusieurs types de clés. Ils sont décrits ci-dessous:

- **Clé candidate:** Une clé candidate est une clé simple ou composite qui est unique et minimale. Elle est unique car aucune ligne dans une table ne peut avoir la même valeur à aucun moment. Elle est minimale car chaque colonne est nécessaire pour atteindre l'unicité.  
- exemple: de la BD COMPAGNIE, if the entity is Employee(EID, First Name, Last Name, SIN, Address, Phone, BirthDate, Salary, DepartmentID), possible candidate keys are:  
•EID, SIN
- Nom et Prénom – assumant qu'il n'y a pas dans la compagnie d'autres employés avec le même nom et prénom  
•Nom et DepartmentID – (pas de 2 différents employés ayant ces même valeurs)
- **Clé composée:** Une clé composée est composée de deux attributs ou plus, mais elle doit être minimale.
- **Clé primaire:** La clé primaire est une clé candidate sélectionnée par le concepteur de la base de données pour être utilisée comme mécanisme d'identification pour l'ensemble d'entités entier. Elle doit identifier de manière unique les tuples dans une table et ne pas être nulle. La clé primaire est indiquée dans le modèle ER en soulignant l'attribut.
- Exemple:
  - Employee(EID, Prénom, Nom, SSN, Adresse, Téléphone, année\_naissance, Salaire, DepartmentID)
- **Clé secondaire:** Une clé secondaire est un attribut utilisé strictement à des fins de récupération (peut être composite), par exemple: téléphone et nom de famille.
- **Clé alternative:** Les clés alternatives sont toutes les clés candidates non choisies comme clé primaire.
- **Clé étrangère:** Une clé étrangère (FK) est un attribut dans une table qui fait référence à la clé primaire dans une autre table OU elle peut être nulle. Les clés étrangères et primaires doivent être du même type de données.
- **Null**
- Un null est un symbole spécial, indépendant du type de données, qui signifie soit inconnu soit inapplicable. Cela ne signifie pas zéro ou vide. Les caractéristiques de null incluent:
  - Pas d'entrée de données
  - Non autorisé dans la clé primaire
  - Devrait être évité dans d'autres attributs
  - Peut représenter
    - Une valeur d'attribut inconnue
    - Une valeur d'attribut connue, mais manquante
    - Une condition « non applicable »
  - Peut créer des problèmes lorsque des fonctions telles que COUNT, AVERAGE et SUM sont utilisées
  - Peut créer des problèmes logiques lorsque des tables relationnelles sont liées

## 1-5 Associations

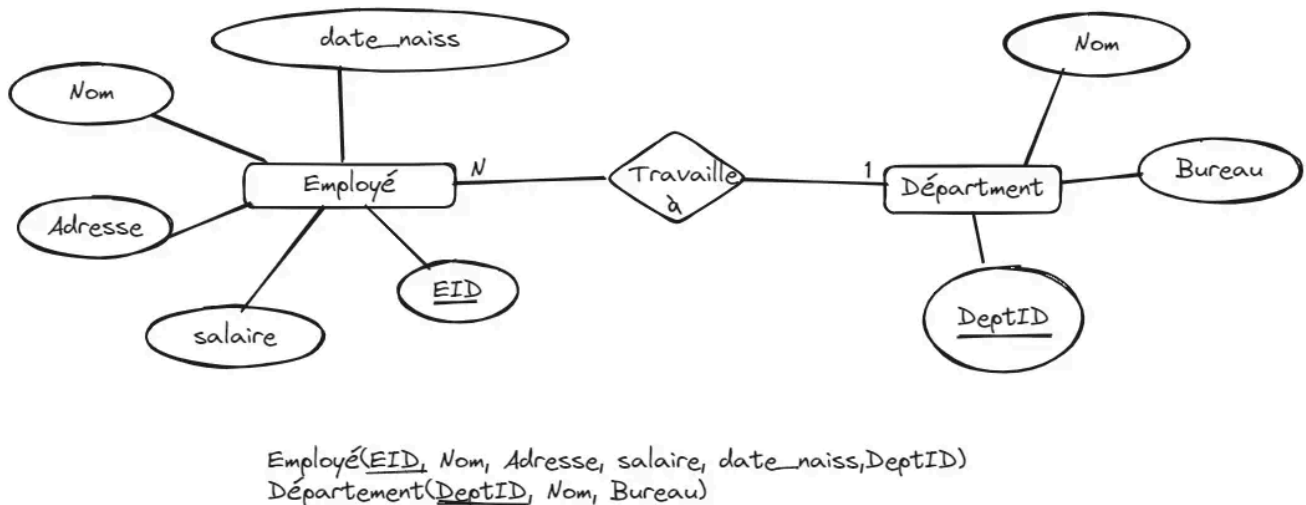
- Les associations (relations) sont le ciment qui lie les tables. Elles servent à relier des informations liées entre des tables.

## 1-5-1 Force des associations

- La force d'une association est basée sur la façon dont la clé primaire d'une entité liée est définie.
- Une association faible, ou non identifiante, existe si la clé primaire de l'entité liée ne contient pas de composant de la clé primaire de l'entité parente.
  - Customer(CustID, CustName)
  - Order(OrderID, CustID, Date)
- Une association forte ou d'identification existe lorsque la clé primaire de l'entité associée contient le composant de clé primaire de l'entité mère. Les exemples comprennent:
  - Course(CrsCode, DeptCode, Description)
  - Class(CrsCode, Section, ClassTime...)

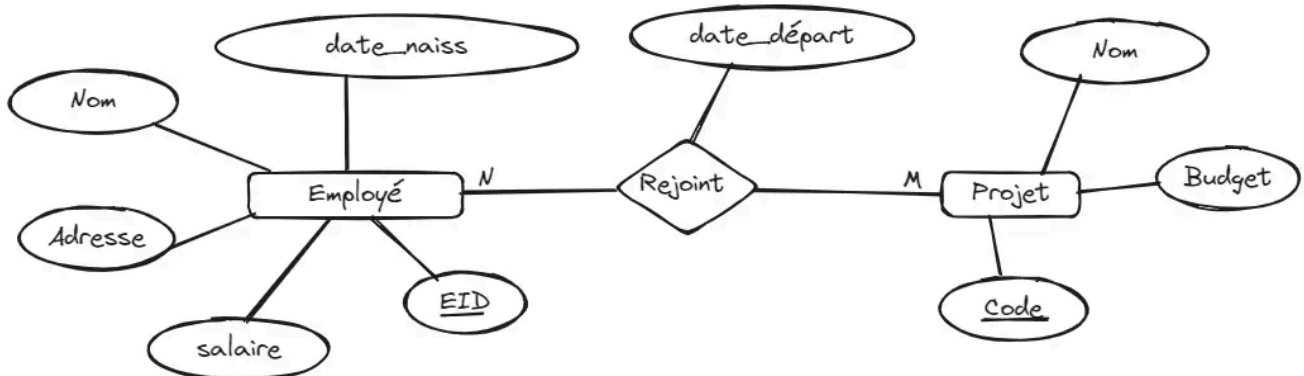
## 1-5-2 Types de associations

- association un à plusieurs (1:M)



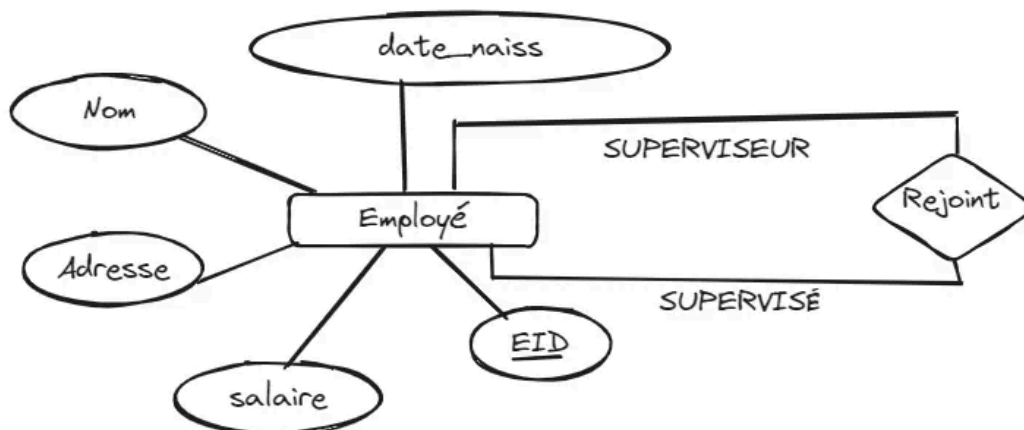
- association un à un (1:1)
- Exemple:
  - dans la base de données COMPAGNIE, il y a la table Employés, dans laquelle un employé est associé avec une épouse. Et une épouse est associée avec un employé.
- association plusieurs à plusieurs (M:N)
- Ne peut pas être implémenté comme tel dans le modèle relationnel.
- Peut être changé en deux associations 1:M.
- Peut être implémenté en décomposant pour produire un ensemble de associations 1:M.
- Implique la mise en œuvre d'une entité composite.
- Crée deux ou plusieurs associations 1:M.
- La table d'entité composite doit contenir au moins les clés primaires des tables d'origine.

- La table de liaison contient plusieurs occurrences des valeurs de clé étrangère.
- Des attributs supplémentaires peuvent être attribués selon les besoins.
- Il peut éviter les problèmes inhérents à une association M:N en créant une entité composite ou une entité pont. Par exemple, un employé peut travailler sur plusieurs projets OU un projet peut avoir plusieurs employés qui y travaillent, selon les règles métier. Ou, un étudiant peut avoir plusieurs classes et une classe peut contenir plusieurs étudiants.



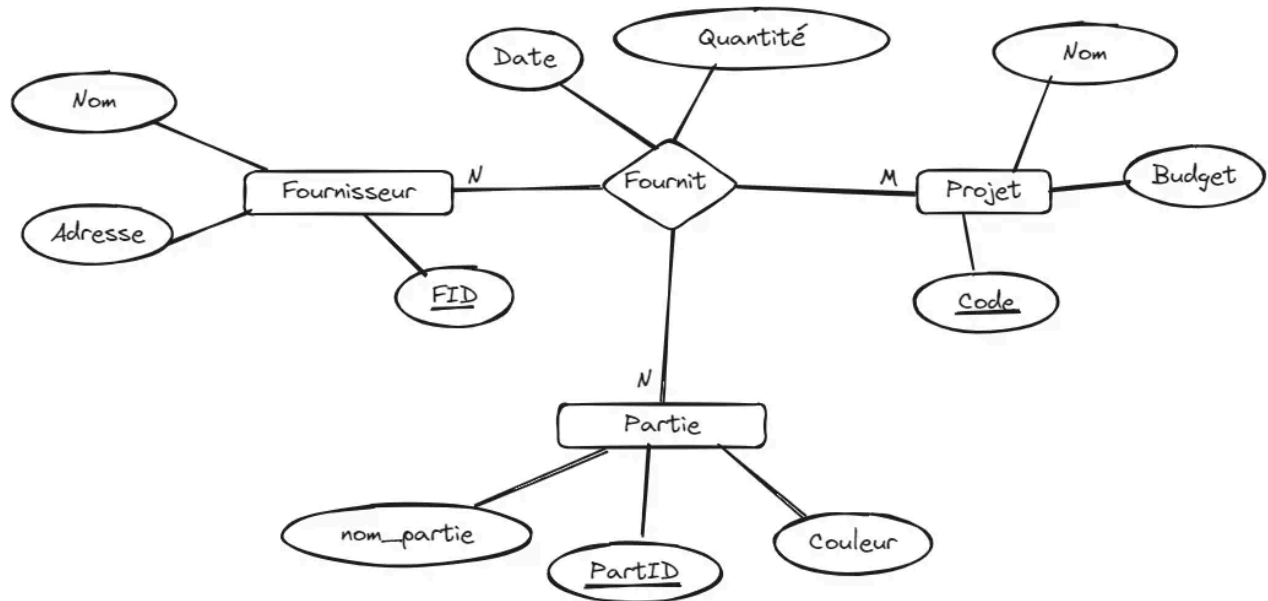
Employé(EID, Nom, Adresse, salaire, date\_naiss)  
 Projet(Code, nom, Budget)  
 Joint(EID, Code, date\_départ)

- \*Exemple de mappage d'un type de association binaire M:N
  - Pour chaque association binaire M:N, identifiez deux associations.
  - A et B représentent deux types d'entités participant à R.
  - Créez une nouvelle association S pour représenter R.
  - S doit contenir les PK de A et B. Ceux-ci ensemble peuvent être les PK dans la table S OU ceux-ci, avec un autre attribut simple dans la nouvelle table R, peuvent être les PK.
  - La combinaison des clés primaires (A et B) fera la clé primaire de S.
- **association unitaire (récursive)**



Employé(EID, Nom, Adresse, salaire, date\_naiss, EID)

- \*\*associations ternaires



Fournisseur(FID, Nom, Adresse)  
 Projet(Code, nom, Budget)  
 Partie(PartID, nom\_partie, couleur)  
 Fournit(FID, Code, PartID, Date, Quantité)

- Exemple de mappage d'un type de association ternaire
  - Pour chaque association n-aire (>2), créez une nouvelle association pour représenter l'association.
  - La clé primaire de la nouvelle association est une combinaison des clés primaires des entités participantes qui détiennent le côté N (plusieurs).
  - Dans la plupart des cas de associations n-aires, toutes les entités participantes ont plusieurs côtés.

## 2- Types d'entités

### 2-1 Entités indépendantes

- Les entités indépendantes, également appelées noyaux, sont l'épine dorsale de la base de données.
- Elles constituent la base sur laquelle reposent les autres tables.
- Les noyaux présentent les caractéristiques suivantes :
  - Elles sont les éléments constitutifs d'une base de données.
  - La clé primaire peut être simple ou composite.
  - La clé primaire n'est pas une clé étrangère.
  - Elles ne dépendent pas d'une autre entité pour leur existence.



## 2-2 Entités dépendantes:

- Les entités dépendantes, également appelées entités dérivées, dépendent d'autres tables pour leur signification.
- Ces entités présentent les caractéristiques suivantes :
  - Les entités dépendantes sont utilisées pour relier deux noyaux entre eux.
  - Elles sont dites dépendantes d'existence par rapport à deux ou plusieurs tables.
  - Les associations plusieurs à plusieurs deviennent des tables associatives avec au moins deux clés étrangères.
  - Elles peuvent contenir d'autres attributs.
  - La clé étrangère identifie chaque table associée.
  - Il existe trois options pour la clé primaire :
    1. Utiliser un composite de clés étrangères de tables associées si elles sont uniques.
    2. Utiliser un composite de clés étrangères et d'une colonne de qualification.
    3. Créer une nouvelle clé primaire simple.

## 2-3 Entités caractéristiques:

- Les entités caractéristiques fournissent plus d'informations sur une autre table.
- Ces entités présentent les caractéristiques suivantes :
  - Elles représentent des attributs multivalués.
  - Elles décrivent d'autres entités.
  - Elles ont généralement une association un à plusieurs.
  - La clé étrangère est utilisée pour identifier davantage la table caractérisée.
  - Les options pour la clé primaire sont les suivantes :
    2. Utiliser un composite de clé étrangère plus une colonne de qualification.
    4. Créer une nouvelle clé primaire simple.
- exemple:
  - Employee (EID, Nom, Adresse, Age, Salaire) – EID est la clé primaire simple
  - EmployeePhone (EID, Téléphone) – EID fait partie d'une clé primaire composée. EID est aussi une clé étrangère.

## Références Bibliographiques

- Watt, Adrienne, and Nelson Eng. *Database design*. 2nd Edition. BCcampus, 2014