Gestion d’association :

* **appartenir (1->n) :** Foreign Key côté Logements : Logements(logeur⇒Logeurs)
* **loger (1->n) :** Foreign Key côté Etudiant : Etudiants(logeur⇒Logeurs)

La Foreign Key Etudiants(logeur⇒Logeurs) est inutile car on est dans un modèle en triangle fermé. On a pas besoin d’avoir logeur dans Etudiants car on a logement dans Etudiants et logeur dans Logements donc on peut retrouver logeur à l’aide de logement dans Etudiants.

Ainsi, on ne va pas prendre en compte cette Foreign Key.

* **occuper (1->n) :** Foreign Key côté Etudiants : Etudiants(logement⇒Logements)

Identification des clés :

On a 3 tables :

* **Logeurs :** la clé primaire est une clé artificielle (idlogeur) : on aurait pu prendre (nom,prenom) comme clé primaire mais il y aurait eu des risques d’homonymie, et en rajoutant les attributs correspondant à l’adresse, la clé primaire serait devenue trop complexe.
* **Logements :** la clé primaire est une clé artificielle (idlogement) : pour caractériser correctement le logement, sans qu’il y ait de risque d’avoir un autre logement avec des caractéristiques identiques, il aurait fallu prendre (numero\_rue,nom\_rue,ville, code\_postal) ce qui aurait rendu la clé primaire trop complexe.
* **Étudiants :** la clé primaire est une clé artificielle (idetudiant) : on aurait pu prendre (nom,prenom) comme clé primaire mais il y aurait eu des risques d’homonymie, et en rajoutant l’attribut semestre, la clé primaire serait devenue trop complexe.

Modèle relationnel (représentation textuelle) :

* **Logeurs**(#idlogeur : int, nom : str, prenom : str, numero\_rue : str, nom\_rue : str, code\_postal : str, ville : str) avec nom, prenom, numero\_rue, nom\_rue, code\_postal, ville non-nuls

On considère que numero\_rue, nom\_rue, code\_postal, ville doivent être non nuls, car l’étudiant doit pouvoir contacter le logeur, et le seul moyen de contact fourni est l’adresse. De plus, le nom et le prénom du logeur doivent être non-nuls car, d’un point de vue légal, on doit savoir qui est le logeur, à qui appartient le logement. (nom, prenom) et (numero\_rue, nom\_rue, code\_postal, ville) ne sont pas forcément unique car il peut y avoir plusieurs logeurs avec un même nom, prénom et il peut également y avoir des logeurs différents ayant la même adresse.

* **Logements**(#idlogement : int, numero\_rue : str, nom\_rue : str, code\_postal : str, ville : str, label : int, logeur⇒Logeur) avec numero\_rue, nom\_rue, ville, code\_postal, logeur non-nuls / (numero\_rue, nom\_rue, ville, code\_postal), logeur unique

numero\_rue, nom\_rue, ville, code\_postal doivent être non-nuls car ce sont les caractéristiques intrinsèques d’un logement. logeur doit être non-nul car on doit savoir à qui appartient le logement. (numero\_rue, nom\_rue, ville, code\_postal) doit être unique car deux logements ne peuvent pas avoir la même adresse et logeur doit être unique car un logement n’a qu’un seul propriétaire. label n’est ni non-nul, ni unique car plusieurs logements peuvent avoir le même label et un logement peut ne pas avoir de label si il est nouveau à la location.

* **Étudiants**(#idetudiant : int, nom : str, prenom : str, semestre : int, logement⇒Logement) avec nom, prenom non-nuls / logement unique et non-nul

nom, prenom car d’un point de vue légal on doit connaître le nom et le prénom du locataire, mais (nom, prenom) n’est pas unique car il peut y avoir des homonymes. Le semestre n’est ni non nul ni unique car la connaissance du semestre d’un étudiant n’est pas obligatoire dans la location d’un appartement, et n’est pas unique car plusieurs étudiants ont le même semestre. Le logement doit être unique car un étudiant a un unique logement, et non-nul car on est dans la base de données de l’ALESC donc tout les étudiants ont un logement.