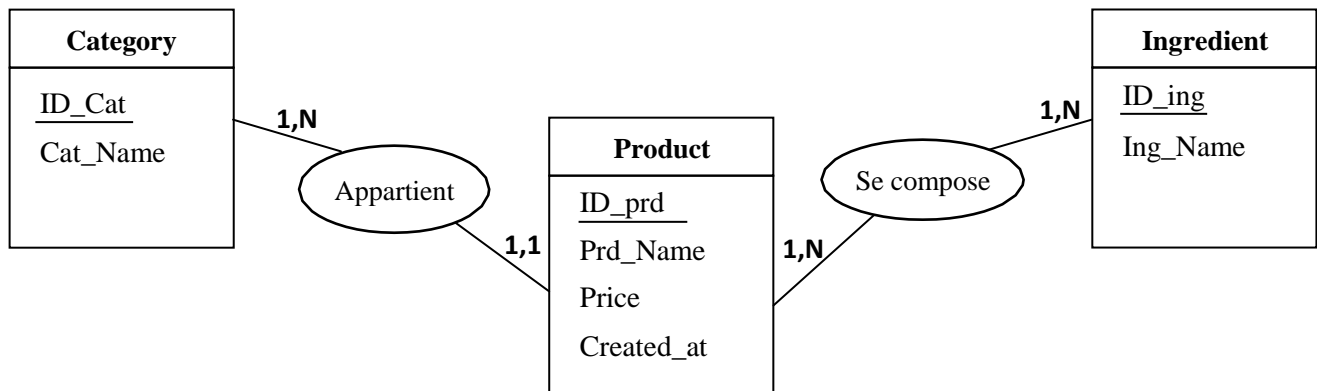




TP3 : Implémentation de votre premier MCD avec Django

Objectif : Le but de ce TP est d'implémenter votre premier MCD construit à partir d'un cahier des charges, l'afficher dans votre espace d'administration et faire de manipulations sur les items de vos classes (Modèles) pour tester les contraintes et les relations entre ces tables. Les deux fichiers que nous allons manipuler sont : *models.py* et *admin.py*.

Exemple : Lors de la conception on a construit le mini MCD suivant :



Created_at : La date de création du produit, générée automatiquement lors de l'insertion d'un produit.
Il s'agit de la date à laquelle le produit a été ajouté à votre BDD.

```
created_at = models.DateTimeField(auto_now=True)
```

*

Indication : (Les relations entre les classes)

Les relations se déclarent dans la classe qui contient la clé étrangère.

1 - * (1 à plusieurs): **name** = models.ForeignKey(class2_name, on_delete=models.CASCADE)

1 - 1 (1 à 1): **name** = models.OneToOneField(class2_name, on_delete=models.CASCADE)

*** - *** (plusieurs à plusieurs): **name**=models.ManyToManyField(class2_name, related_name="name2")

Le nom utilisé pour la relation inverse

Travail demandé :

En considérant le mini MCD décrit ci-dessus

- 1- Faire le passage au modèle relationnel.
- 2- Créer une nouvelle application Django nommée : **produit** dans votre même projet.
- 3- Modifier le fichier *models.py* pour permettre l'implémentation de la conception proposée.
- 4- Modifier le fichier *admin.py* pour permettre l'affichage de vos modèles dans l'interface d'administration Django.
- 5- Faire des manipulations de données sur vos tables (insertion, modification, suppression) pour tester le fonctionnement des relations et des contraintes.