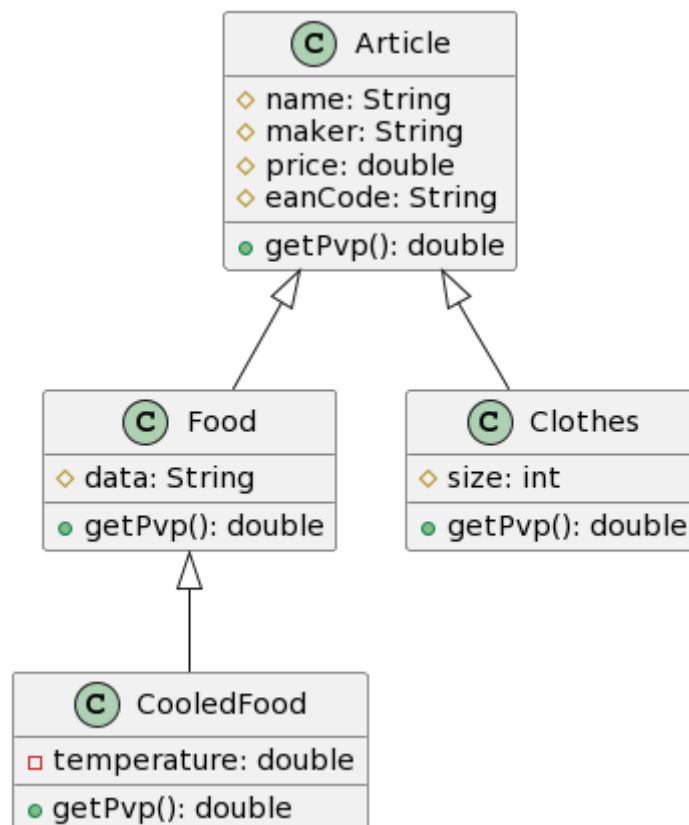


# Exercici Supermercat

Crear l'estructura del model de classes per als articles d'un supermercat.



Crear les classes indicades al diagrama de classes UML, amb els constructors pertinents i els altres mètodes (accessors, equals, ...)

Dos articles són iguals i els seus codis EAN són iguals.

El mètode `getPvp()` retorna el preu final de venda al públic, un cop aplicats els impostos i descomptes pertinents:

- Article: IVA general del 21%
- Food: IVA del 4%
- Clothes: Talla entre 30 i 40, el preu base serà el 80%. Talla entre 20 i 30, preu base del 90%. Altres casos, preu íntegre de l'article. En tots els casos, cal aplicar després l'IVA del 21%.

Per a la classe *CooledFood*, cal crear també un mètode *rotten()*, el qual rep com a paràmetre un valor de temperatura ambient i retornarà un booleà (true si el producte no té garantida la qualitat perquè la temperatura ambient és superior a l'establerta per a la conservació de l'article).

Crear la classe *Supermarket*, la qual contindrà com a atributs un nom (*name*) i un array d'articles.

Per a la gestió de l'array també necessita una variable o constant *MAX\_ARTICLES* (per exemple, 100) i una altra *numArticles* (per controlar quants articles hi ha en cada moment a l'array).

La classe Supermarket ha de disposar de mètodes per a la gestió de la llista d'articles. Com a mínim:

- `searchArticle(eanCode: String): Article`
- `addArticle(article: Article): boolean`

No s'ha de permetre duplicitats a l'EAN.

A la classe principal cal crear un objecte *Supermarket* amb els seus articles (mínim 10).

També crear un menú d'usuari amb les opcions:

- Sortir
- Llistar tots els articles
- Afegir Article
- Buscar article per codi EAN