---------------------------------------------------------

TRI PERSONNALISE :

---------------------------------------------------------  
def tri\_personnalise(e): # e -> on récupère un élément  
 return e  
 return len(e)---------------------------------------------------------

CREER COLLECTION :

---------------------------------------------------------  
def afficher(collection, n\_premiers\_elements=-1): # Implémenter la collection  
 # collection.sort(reverse=True, key=tri\_personnalise) # Trier par ordre alphabétique / reverse -> inversé  
  
 # UTILISATION DUN SLICE POUR AFFICHER LES PREMIERS ELEMENTS :  
 c = collection  
 if not n\_premiers\_elements == -1:  
 c = collection[0:n\_premiers\_elements]

# PROTEGER LA FONCTION -> CONDITION DE VIDE :  
 nb\_pizzas = len(c)  
 if nb\_pizzas == 0:  
 print("AUCUNE PIZZA")  
 else: # Sinon, on déroule la suite de la fonction.  
  
 # TITRE  
 print(f"----- LISTE DES PIZZAS ({nb\_pizzas}) -----")  
 for i in c: # afficher les pizzas : 1 par ligne  
 print(i)  
 print()  
 # Afficher le nom de la première pizza  
 print("Première pizza : " + c[0])  
 # Afficher le nom de la dernière pizza  
 print("Dernière pizza : " + c[-1])

# AJOUTER UNE PIZZA TAPEE PAR L'UTILISATEUR :  
def ajouter\_pizza\_utilisateur(c):  
 p = input("Pizza à ajouter : ") # Ici on utilise input car on demande qqch à l'utilisateur  
 if p.lower() in c: # Si p existe deja dans la collection -> méthode + rapide  
 print("ERREUR : Cette pizza existe déjà !")  
 else:  
 c.append(p) # p pour récupérer la réponse de l'utilisateur demandée ci-dessus.

ERREUR SI LA PIZZA DE L'UTILISATEUR EXISTE DEJA :  
def pizza\_existe(e, collection): # Mettre en paramètre e la pizza rentrée par l'utilisateur.  
 for i in collection: # Boucle for pour regarder si l'élément existe  
 if i == e:  
 return True  
 return False  
  
  
  
pizzas = ["4 fromages", "végétarienne", "hawai", "calzone"]  
# vide = () # Tuple avec aucun élément  
ajouter\_pizza\_utilisateur(pizzas)  
afficher(pizzas, 2)

RESPECT DE LA CASSE DES CARACTERES :  
# lower() -> passer en minuscules  
# upper() -> passer en majuscules