

STRUCTURE

⇒ INTERFACE (ensemble de méthodes)

Personne

prénom }
 nom } structure
 age } Personne
 code sexe }
 n° de ligne
 Statut
 A-faire
 Canal de lecture

initialise
travaille
vers-string
donne statut

Création
Producteur

p: ∅
 n: ∅
 a: ∅
 cs: ∅
 # : Random (taille
Fichier)
 s: "V"

Todo: Tableau de p
 Vide
 Canal de lecture (L)

Fonction
 Initialiser
 Travailler
 +
 Lecteur

Conseiller
Municipale

Processus lecteur

p: Lit fichier
 m: conseil.municipal.txt
 a: Réception
 cs: infos
 # : ligne #
 s: rempli
 s: "R"

Todo: [Choisit jusqu'à
 6 travaux au hasard]
 (utiliser un Travail())

Fonction
 Travailler
 Travailler

Si récupère une action
 en ①, alors
 on applique l'action
 à la personne et
 modifie le champ
 work pendant:
 Si Todo=[], "C" ②
 passe au
 collateur
 ① récupère une action
 à faire dans la Todo liste

Client

Serveur

Table d'assoc Identifiant → Personne_Serveur

Personne_Serveur

prénom }
 nom } structure
 age } Personne
 code sexe }
 n° de ligne
 Statut
 A-faire
 Canal de lecture

Fonction
 Création
 Epaveant
 Fonction
 Initialiser
 Epaveant
 Fonction
 Travailler
 Epaveant

• crée personne serveur + associa° id → Pers
 • récupère la pers à partir de l'identifiant
 • crée la Todo liste
 • Statut "R" (rempli)
 • récupère la pers à partir de l'identifiant
 • récupère le 1er elt Todo liste
 • Si travaille
 • Si Todo Vide renvoie le message "terminé"

Personne_dist

identifiant (int)
 URL-propy
 (canal de
 message-propy)
 Canal de retour (chan-string)
 méthode (string)
 identifiant de la personne

Création
 Producteur distant

Interface: donne statut, travaille, initialise, testing
 Méthode
 local crée un canal
 message (identifiant de p, "Nécess", canal de lecture)
 p-propy ← envoie le message au p-propy
 ← local attend le résultat

