

Les Méthodes de relation entre classes :

Classe ClientRégulier :

```
// methodes de la relation avec commande
public void ajouterCommande(Commande commande){
    if(!this.getCommandes().contains(commande)){
        commandes.add(commande);
        commande.setClientRégulier(this);
        nbCommande++;
    }
}

public void AnullerCommande(Commande commande){
    this.commandes.remove(commande);
    commande.setClientRégulier(null);
    nbCommande--;
}
```

Classe Employé :

```
//Methodes Livreur
public void ajouterLivraison(Livraison livraison){
    if(!this.livraisons.contains(livraison)){
        this.livraisons.add(livraison);
        livraison.setEmployé(this);
    }
}

public void anulerLivraison(Livraison livraison){
    if(this.livraisons.remove(livraison)){
        livraison.setEmployé(null);
    }
}
```

Classe Commande :

```
//ClientRégulier:
public void ajouterClient(ClientRégulier clientRégulier){
    this.clientRégulier=clientRégulier;
    if(!clientRégulier.getCommandes().contains(this)){
        clientRégulier.getCommandes().add(this);
    }
}
public void supprimerClient(){
    if(this.clientRégulier.getCommandes().remove(0: this)){
        this.clientRégulier=null;
    }
}
```

Classe DétailCom :

```
public void AjouterCon(Consomable consomable,int i){
    this.consomable=consomable;
    this.setNbportion(i);
    this.montant=this.Nbportion*consomable.getPrixCon();
}
public void supprimerCon(){
    this.consomable=null;
    this.setNbportion(0);
    this.setMontant(0);
}
```

Classe Livraison:

```
// Methode pour la relation avec un Employé:
public void setEmployé(Employé employé) { this.employé = employé; }
public void ajouterLivreur(Employé employé){
    this.setEmployé(employé);
    if (!employé.getLivraisons().contains(this))employé.getLivraisons().add(this);
}
public void anulerLivraison(){
    this.employé.getLivraisons().remove(o: this);
    this.employé=null;
}
// Methode pour la relation avec un Commande:
public void ajouterCommande(Commande commande) { this.setCommande(commande); }
public void supprimerCommande(){
    this.setCommande(null);
    this.setEmployé(null);
}
```

Classe FicheTechCon :

```
//Methode pour la relation avec une methode de préparation
public void modifierMethode(MethodePréparation methodePréparation){
    this.methodePréparation=methodePréparation;
}
//Methodes pour la relation avec un Ingrédient
public void ajouterIngrédient(Ingrédient ingrédient){
    if(!this.getIngrédients().contains(ingrédient)){
        this.getIngrédients().add(ingrédient);
    }
}
public void supprimerIngrédient(Ingrédient ingrédient) { this.getIngrédients().remove(ingrédient); }
```