

TD02:

- Exercice:
- 1- les attributs de Produit : identité, nom, et sa qualité
 - 2- les méthodes de Produit : augmenter Quantité() et vendre()()
 - 3- Encapsulation → getter

```
public class produit {  
    private int id;  
    private String nom;  
    private int qt = 0;  
    public produit(int id, String nom) {  
        this.id = id;  
        this.nom = nom;  
    }  
    public void augmenter_qt(int qt) {  
        if (qt > 0) {  
            this.qt += qt;  
        }  
        else {  
            System.out.println("erreur");  
        }  
    }  
    public void vendre() {  
        if (qt > 0) {  
            qt --;  
        }  
        else {  
            System.out.println("stock épuisé");  
        }  
    }  
}
```

```

public int getId() {
    return id;
}

public String getNom() {
    return nom;
}

public int getQl() {
    return ql;
}

```

```

import java.util.List;
class GestionStock {

```

```

    private List <Produit> lp = new ArrayList <Produit> ();

```

```

    public void ajouterProduit(Produit P) {
        lp.add(P);
    }

```

```

    public void AfficherDetailProduit() {
        for (int i = 0; i < lp.size(); i++) {
            System.out.println(lp.get(i).toString());
        }
    }

```

```

    public void trouverProduit(int idP) {
        if (Produit P: lp) {
            if (P.getId() == idP) {
                return P;
            }

```

lp.get(i) ←

```

            else {
                return null;
            }

```

```

        }
    }

    public int nombreEnstock(int id) {
        Produit P = this.trouverProduit(id);
        if (P != null) {
            return P.getQl();
        }
        else {
            return 0;
        }
    }

```

for (int i = 0; i < lp.size(); i++)
if (lp.get(i).getId() == idP)

```

public void livraison (int id, int qt) {
    for (int i = 0; i < p.size(); i++)
    if (product p = this.trouverProduit(id)) {
        Product p = null;
        p = this.trouverProduit(id);
        if (p != null) {
            p.setqt (p.getqt + qt);
        }
    }
}

```

```

for (Product P : lp) {
    if (P.getId() == id) {
    }
}

```

```

6. c. public Product trouverProduit (String n) {
    for (Product P : lp) {
        if (P.getNom().equals(n))
            return P;
    }
    return null;
}

```

- 7. a -

```

public class Date {
    private int j, m, a;
}

```

```

public class ProduitFrais extends Produits {
    private Date dp;
}

```

```

public ProduitFrais (int id, String Nom, int qt, Date dp) {
    super (id, Nom, qt);
    this.dp = dp;
}

```