

## TD5 – Patrons de conception

### Exercice 1 : Composite et l'application Document

En vue de construire une application d'édition automatique, on se propose de définir un ensemble d'entités qui modélisent la composition d'un ensemble de pages imprimées. Les règles qui définissent les relations entre ces entités sont les suivantes :

1. un document est *composé* de pages,
2. chaque page est *organisée* en colonnes et/ou cadres,
3. une colonne peut *contenir* des cadres, des images ou des lignes de texte,
4. un cadre peut *contenir* des colonnes ou des images,
5. une ligne de texte est *composée* de caractères et d'images,
6. les caractères et les images n'ont qu'une valeur, ils ne peuvent pas contenir d'autres éléments.

#### Question 1

En vous appuyant sur les termes en italique des règles ci-dessus, identifiez et nommez les classes entités du modèle de document spécifié et dessinez le diagramme de classes du modèle en traduisant les relations de composition.

#### Question 2

En vous appuyant sur le pattern Composite, proposez une autre structure d'implantation pour ce modèle

---

### Exercice 2 : Découverte d'un patron

1. Réaliser le diagramme de classes de l'application ci-dessous.
  2. Mettre en relief le patron utilisé.
  3. Donner le résultat de l'exécution de la classe ServeurWeb.
- 

```
public interface Document {  
    public void setContenu(String contenu);  
    public void dessine();  
    public void imprime();  
}
```

---

```
public class DocumentHtml implements Document {  
    protected String contenu;  
  
    public void setContenu(String contenu) {  
        this.contenu = contenu;  
    }  
  
    public void dessine() {  
        System.out.println("Dessine document HTML : " +  
            contenu);  
    }  
  
    public void imprime() {  
        System.out.println("Imprime document HTML : " +  
            contenu);  
    }  
}
```

```
}
```

---

```
public class DocumentPdf implements Document {
    protected ComposantPdf outilPdf = new ComposantPdf();

    public void setContenu(String contenu) {
        outilPdf.pdfFixeContenu(contenu);
    }

    public void dessine() {
        outilPdf.pdfPrepareAffichage();
        outilPdf.pdfRafraichit();
        outilPdf.pdfTermineAffichage();
    }

    public void imprime() {
        outilPdf.pdfEnvoieImprimante();
    }
}
```

---

```
public class ComposantPdf {
    protected String contenu;

    public void pdfFixeContenu(String contenu) {
        this.contenu = contenu;
    }

    public void pdfPrepareAffichage() {
        System.out.println("Affichage PDF : Debut");
    }

    public void pdfRafraichit() {
        System.out.println("Affichage contenu PDF : " +
            contenu);
    }

    public void pdfTermineAffichage() {
        System.out.println("Affichage PDF : Fin");
    }

    public void pdfEnvoieImprimante() {
        System.out.println("Impression PDF : " + contenu);
    }
}
```

---

```
public class ServeurWeb{

    public static void main(String[] args) {
        Document document1, document2;
        document1 = new DocumentHtml();
        document1.setContenu("Hello");
        document1.dessine();
        System.out.println();
        document2 = new DocumentPdf();
        document2.setContenu("Bonjour");
        document2.dessine();
    }
}
```