

# Interfaces Graphiques en Java

---

Semestre 4

## 0.1 Généralités

Pour concevoir une interface graphique il faut :

- Disposer de composants graphiques (Fenêtres, boutons, champs, ...)
- Disposer d'un mécanisme pour agencer les composants graphiques
- Associer un traitement à un composant

## 0.2 Les composants de l'AWT

- Package `java.awt`
- 3 ensemble de classes
  - Les composants simples (Héritant de `Component`)
  - Les composants de réceptacle (Héritant de `Container`)
  - Les composants de menu (Héritant de `MenuComponent`)
- Possibilité d'imbriquer des composants de réceptacle
  - Notion de parenté

### 0.2.1 Exemple de composant graphique : `TextField`

Méthode	Description
<code>String getSelectedText()</code>	Renvoie le texte sélectionné
<code>int getSelectionStart()</code>	Renvoie la position de début de sélection
<code>int getSelectionEnd()</code>	Renvoie la position de fin de sélection
<code>String getText()</code>	Renvoie le texte contenu dans l'objet
<code>boolean isEditable()</code>	Indique si le texte est modifiable
<code>void setEditable(boolean b)</code>	Autoriser ou interdire la modification de texte

## 0.3 Les gestionnaires de mise en page

Un gestionnaire de mise en page est un `LayoutManager`, c'est un objet associé à un `Container`.

### 0.3.1 Différents gestionnaires

- `FlowLayout` : Ajout les composants les uns après les autres
- `BorderLayout` : Ajoute les composants en fonction d'une position géographique
- `GridLayout` : Ajoute les composants les uns après les autres dans un quadrillage de n lignes et m colonnes<sup>1</sup>
- `GridBagLayout` : Ajoute les composants les uns après les autres dans un quadrillage de n lignes et m colonnes<sup>2</sup>
- `CardLayout` : Permet de positionner des onglets pour visualiser plusieurs feuilles

---

1. Taille identique

2. Taille non identique

**0.4 La gestion des événements**

**0.5 Les composants SWING**

**0.6 Le contexte graphique**

**0.7 Les appels**

**0.8 Les environnements de développements intégrés**