

Généricité

Cours sur la généricité en java

```
public static Object troisieme(Object[] tab) {  
    return tab[2];  
}  
  
public static void main(String[] args) {  
    String argument = (String) troisieme(args);  
    ...  
}
```

```
// La même fonction que la précédente  
public static <T> T troisieme(T[] tab) {  
    return tab[2];  
}  
  
public static void main(String[] args) {  
    String argument = troisieme(args);  
    ...  
}
```

```
public class Paire<E> {  
    private E elementUn;  
    private E elementDeux;  
    public Paire(E e1, E e2) {  
        this.elementUn = e1;  
        this.elementDeux = e2;  
    }  
    public E get(int index) throws IndexOutOfBoundsException {  
        if (index == 0) {  
            return this.elementUn;  
        } else if (index == 1) {  
            return this.elementDeux;  
        } else {  
            throw new IndexOutOfBoundsException("La valeur "+index+" n'est pas  
acceptable !");  
        }  
    }  
}
```

```
public class Main {
    public static void main(String[] args) {
        Paire<Integer> p;
        p = new Paire<Integer>(12, 7);
        System.out.println(p.get(1));
    }
}
```

```
//
public static void <T extends JComponent> ajouterComposants(LinkedList<T>
liste, Container conteneur) {
    conteneur.setLayout(new GridLayout(liste.size(), 1));
    for(JComponent composant : liste) {
        conteneur.add(composant);
    }
}
```

```
public static void ajouterComposants(LinkedList<? extends JComponent> liste,
Container conteneur) {
    conteneur.setLayout(new GridLayout(liste.size(), 1));
    for(JComponent composant : liste) {
        conteneur.add(composant);
    }
}
```

```
public static <T> void remplirPaire(Paire<? super T> destination, T e1, T e2)
{
    destination.set(0, e1);
    destination.set(1, e2);
}
```