

# SAE JAVA :

## Phase 3 :

Pour améliorer la robustesse du code, on peut identifier plusieurs situations qui peuvent provoquer des erreurs.

Les erreurs possibles :

### **Vérification des valeurs d'entrée :**

Lors de la création d'une forme, on peut vérifier si les points passés en paramètre sont valides (non null, même valeurs avec les autres points).

```
Code : try {  
    if ((p1.equals(p2) || p2.equals(p3) ||  
p3.equals(p1)) || (p1==null || p2==null || p3==null) ) {  
        throw new IllegalArgumentException("Les points  
fournis ne forment pas un chapeau valide.");  
    } catch (IllegalArgumentException e) {  
        System.out.println("Erreur lors de la création d'une  
forme : " + e.getMessage());  
    }  
}
```

---

Dans la méthode `deplacer()`, nous pouvons vérifier si les valeurs de déplacement sont valides (non null, valeurs acceptables).

```
Code : try {  
    if (deplacementX < 0 || deplacementY < 0) {  
        throw new IllegalArgumentException("Les  
valeurs de déplacement doivent être positives.");  
    } catch (IllegalArgumentException e) {  
        System.out.println("Erreur lors du déplacement du  
Chapeau : " + e.getMessage());  
    }  
}
```

### **Erreurs lors de l'ajout de formes composantes :**

Dans la méthode `ajouterForme()`, nous pouvons vérifier si la forme passée en paramètre n'est pas null avant de l'ajouter à la liste.

```

Code: try {
    if (forme != null) {
    } else {
        throw new IllegalArgumentException("La forme à
ajouter ne peut pas être null.");
    }
} catch (IllegalArgumentException e) {
    System.out.println("Erreur lors de l'ajout d'une forme
à l'ardoise : " + e.getMessage());
}

```

### Erreurs lors du dessin des formes :

Dans la méthode dessiner() , nous pouvons gérer les cas où une forme ne retourne aucun segment ou renvoie des segments invalides.

```

Code: try {
    // ...
} catch (Exception e) {
    System.out.println("Erreur lors du dessin d'une
forme composante : " + e.getMessage());
}

```

### Erreurs de déplacement :

Dans la méthode déplacer() , nous pouvons gérer les exceptions potentielles qui pourraient être levées lors du déplacement des formes composantes.

```

Code: try {
    // ...
} catch (Exception e) {
    System.out.println("Erreur lors du déplacement
d'une forme composante : " + e.getMessage());
}

```