

**Exercice1 :** Ecrivez un programme demandant de l'utilisateur de saisir les données d'un tableau dont la taille est fixée au début.

-Ensuite le programme demande de l'utilisateur d'introduire un indice pour qu'il divise sa valeur correspondante par un nombre saisi par l'utilisateur.

-Si l'indice saisi par l'utilisateur dépasse la taille du tableau ,une exception de type `ArgumentOutOfRangeException` est levée et traitée.

-Si le diviseur égale à 0, une exception de type `DivideByZeroException` est levée et traitée.

1- Capturer l'exception dans un bloc try catch

2- Propager l'exception vers la méthode appelante.

**Exercice2 :** On souhaite gérer les exceptions afin de s'assurer que la longueur du rayon d'un cercle est bien un réel positif. Dans le cas où l'on donnera une valeur négative, la longueur du rayon du cercle sera initialisée à son opposé .Les étapes à suivre sont les suivantes :

1- Écrire la déclaration de la classe `ValRayonValideException` qui permettra d'instancier une exception dès qu'une longueur de rayon négative sera donnée.

2- Écrire la déclaration de la classe `CouleurValideException` qui permettra d'instancier une exception dès que la couleur est un objet nul ou de longueur nul ou de couleur rouge.

3- Construire la classe `Cercle` en gérant correctement le code de ses méthodes qui sont à risque.

4- Tester la classe `Cercle` dans la fonction main. Le squelette de la classe `Cercle` est donné par : **public class Cercle {**

<b>private double x, y, rayon;</b>	<b>(x, y) est le centre du cercle dont le rayon est donné par rayon.</b>
<b>private String couleur;</b>	<b>La couleur du cercle</b>
<b>public Cercle(double x, double y, double rayon, String couleur)</b>	<b>Constructeur de la classe</b>
<b>public double getRayon() public String getCouleur()</b>	
<b>public void setRayon(double newRayon)</b>	<b>Fonction permettant de modifier la valeur du rayon du cercle. Si la valeur du nouveau rayon donnée par newRayon est négative alors une exception de type <code>ValRayonValideException</code> est levée et traitée</b>

<b>public void setCouleur(String newCouleur)</b>	<b>Fonction permettant de modifier la couleur du cercle courant, en gérant des exceptions de type <code>NullPointerException</code> . si le paramètre <code>newCouleur</code> est nul ou s'il est une chaîne de caractères vide ou si <code>newCouleur</code> est de couleur rouge ceci en utilisant la classe <code>CouleurValideException</code></b>
<b>public String toString()</b>	<b>Fonction permettant d'afficher les caractéristiques du cercle courant</b>

}