

```
cd /home/event/discovery_piscine
mkdir cell31
cd cell31
mkdir ex01
cd ex01
```

```
cp /home/event/discovery_piscine/cell30/ex00/background.html balloon.html
```

```
vim balloon.html
i
```

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Balloon</title>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>

      // hauteur de 100% pour html, body et le container
      // (pour pouvoir centrer le ballon verticalement dans son container) (voir plus bas)
      html, body, .container {
        height: 100%;
      }

      // supprime les marges externes appliquées par défaut par le navigateur sur body
      body {
        margin: 0;
      }

      // pour centrer verticalement le ballon sur la page
      .container {
        display: flex;
        justify-content: center;
        align-items: center;
      }
    </style>
  </head>
  <body>
```

```

        #balloon {                // règles pour le ballon
            height: 200px; // hauteur initiale de 200px
            width: 200px;  // largeur initiale de 200px
            background-color: red; // couleur initiale rouge
            border-radius: 50%;    // pour que les bords de la div soit arrondis :
                                   // le bloc se transformera alors en cercle
                                   // une valeur de 50% suffit (thanks Emma)
            cursor: pointer;       // pour transformer le curseur en main
                                   // lors du survol de la souris sur le ballon
        }
    </style>
</head>
<body>
    <div class="container">        // container du ballon
        <div id="balloon"><div>    // ballon
        </div>
    <script>
        // on sélectionne notre élément ballon (l'élément d'id ballon) et on le stocke dans une
        // constante
        const balloon = document.getElementById("balloon");
        // tableau de couleurs que l'on appliquera au ballon
        const balloonBackgroundColors = ['red', 'green', 'blue'];
        // valeur de la taille à additionner à la taille du ballon pour le faire grossir
        const growValue = 10;
        // valeur de la taille à soustraire à la taille du ballon pour le faire rétrécir
        const shrinkValue = 5;
        // taille minimale du ballon
        const balloonMinSize = 200;
        // taille maximale du ballon
        const balloonMaxSize = 420;

        // on initialise la taille du ballon à sa taille minimale
        let balloonSizeValue = balloonMinSize;

        // indice pour parcourir le tableau de couleurs, initialisé à 0
        // car la couleur initiale est 'red'
        // donc l'élément d'indice 0 dans le tableau balloonBackgroundColors
        let i = 0;
    </script>
</body>
</html>

```

```

// on "écoute" le clic sur le ballon
balloon.addEventListener("click", function() {
    // quand le ballon est cliqué, on exécute la fonction changeSize
    // avec la valeur "click"
    changeSize("click");
    // et la fonction changeColor avec la valeur "click"
    changeColor("click");
});

// on écoute la sortie de souris du ballon
balloon.addEventListener("mouseleave", function() {
    // idem, avec la valeur "mouseleave"
    changeSize("mouseleave");
    // idem
    changeColor("mouseleave");
});

// fonction de changement de taille (on lui passe le paramètre action, qui contiendra
// soit "click", soit "mouseleave"
// la variable action contiendra soit "click", soit "mouseleave"
function changeSize(action) {
    // si l'action est un clic
    if (action === "click") {
        // ATTENTION : LES TERNAIRES SONT INTERDITS A 42 !
        // UTILISER PLUTOT CE QUI EST INDIQUE PLUS BAS (IF ELSE)
        // si la taille du ballon après augmentation de sa taille
        // est supérieure à la
        // taille maximum (balloonSizeValue + growValue) > balloonMaxSize) ?
        // la variable balloonSizeValue prendra la valeur de balloonMinSize
        // (on revient à 200)
        // sinon (:), elle prendra la valeur de balloonSizeValue + growValue
        // (on ajoute 10)
        balloonSizeValue = ((balloonSizeValue + growValue) > balloonMaxSize) ?
        balloonMinSize : balloonSizeValue + growValue;

        // se traduit par :
        // if ((balloonSizeValue + growValue) > balloonMaxSize) < balloonMinSize)

```

```

        //      balloonSizeValue = balloonMinSize;
        // else
        //      balloonSizeValue = balloonSizeValue + growValue;
    }

    // sinon, si c'est une sortie de souris
    else if (action === "mouseleave") {
        // ATTENTION : LES TERNAIRES SONT INTERDITS A 42 !
        // UTILISER PLUTOT CE QUI EST INDIQUE PLUS BAS (IF ELSE)

        // si la taille du ballon après réduction de sa taille est inférieure à
        // la taille minimum du ballon, on revient à la taille minimum
        // sinon, on diminue sa taille
        balloonSizeValue = ((balloonSizeValue - shrinkValue) < balloonMinSize) ?
        balloonMinSize : balloonSizeValue - shrinkValue;

        // se traduit par :
        // if ((balloonSizeValue - shrinkValue) < balloonMinSize)
        //      balloonSizeValue = balloonMinSize;
        // else
        //      balloonSizeValue = balloonSizeValue - shrinkValue;
    }

    // on applique la nouvelle taille (en pixels) à la hauteur du ballon
    balloon.size.height = balloonSizeValue + "px";
    // et à la largeur du ballon
    balloon.size.width = balloonSizeValue + "px";
}

// fonction de changement de couleur
function changeColor(action) {
    // si c'est un clic
    if (action === "click")
        // on augmente de 1 l'indice, en l'ajustant pour ne pas qu'il sorte des
        // limites du tableau (limite supérieure ici)
        i = adjustIndex(i + 1, balloonBackgroundColors.length);
    // sinon, si c'est une sortie de souris
    else if (action === "mouseleave")

```

```

        // on diminue de 1 l'indice, en l'ajustant pour ne pas qu'il sorte des
        // limites du tableau (limite inférieure ici)
        i = adjustIndex(i - 1, balloonBackgroundColors.length);

        // on applique la nouvelle couleur à la couleur du fond du ballon
        // en allant chercher la couleur correspondante dans le tableau, à l'indice
        // indiqué
        balloon.style.backgroundColor = balloonBackgroundColors[i];
    }

    // pour ajuster l'indice (permet de faire boucler l'indice dans les limites du tableau)
    function adjustIndex(index, arrayLength) {
        // on ajoute la longueur du tableau à l'indice incrémenté ou décrémenté
        // (index vaudra i + 1 ou i - 1)
        // ici, arrayLength = 3 (3 éléments dans le tableau, 'red', 'green' et 'blue')

        // index 0 = red ; index 1 = green ; index 2 = blue

        // avec index = 2 (dernier index du tableau) :
        // après incrémentation, index + arrayLength = 2 + 1 + 3 = 6
        // => le reste de la division de 6 par 3 est 0
        // donc si on sort du tableau par incrémentation, on revient à 0
        // (1er index du tableau)

        // avec index = 0 (premier index du tableau) :
        // après décrémentation, index + 3 = 0 - 1 + 3 = 2
        // => le reste de la division de 2 par 3 est 2
        // (car 2 est inférieur à 3, il ne peut donc
        // pas être divisé par 3)
        // donc si on sort du tableau par décrémentation,
        // on revient à 2 (dernier index du tableau)
        return (index + arrayLength) % arrayLength;
    }
}
</script>
</body>
</html>

```

echap  
:wq  
enter

firefox balloon.html