

ACTIVIDAD 5.1

ALFONSO RINCÓN CUERVA
2º DAM
DESARROLLO DE INTERFACES

ÍNDICE

ACTIVIDAD 1	1
ACTIVIDAD 2	2
ACTIVIDAD 3	3
ACTIVIDAD 4	4
ACTIVIDAD 5	5
ACTIVIDAD 6	7
ACTIVIDAD 7	7
ACTIVIDAD 8	8
BIBLIOGRAFÍA.....	10

ACTIVIDAD 1

Mediante NetBeans, utiliza XHTML para crear tu propio CV.

```

1 <!DOCTYPE html>
2 <html xmlns="http://www.w3.org/1999/xhtml">
3   <head>
4     <title>CV Actividad 1</title>
5     <meta charset="UTF-8"/>
6     <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
7     <style>
8       {
9         padding: 0;
10        margin: 0;
11        box-sizing: border-box;
12        font-family: Arial;
13      }
14      body {
15        background-color: #eee;
16        display: flex;
17        justify-content: center;
18        padding: 3%;
19      }
20      main {
21        background-color: #fff;
22        width: 900px;
23        border-radius: 5px;
24        box-shadow: 0px 2px 4px rgba(0,0,0,0.2);
25        padding: 4%;
26      }
27      h1 {
28        color: #c00000;
29        margin-bottom: 15px;
30        font-size: 2.4em;
31        text-align: center;
32      }
33      h2 {
34        color: #b32e2e;
35        margin: 20px 0px 5px 0px;
36      }
37      h3 {
38        margin-bottom: 7px;
39      }
40      p {
41        margin-bottom: 25px;
42      }
43      ul {
44        padding-left: 30px;
45      }
46      div {
47        margin-bottom: 40px;
48      }
49    </style>
50  </head>
51  <body>
52    <main>
53      <!--Primera sección-->
54      <div>
55        <h1>Alfonso Rincón Cuerva</h1>
56        <h3>Dirección</h3>
57        <p>Calle Falsa 9, Talavera de la Reina, España</p>
58        <h3>Email</h3>
59        <p>alfonso.rincondev@riberadelatajo.es</p>
60      </div>
61      <!--Segunda sección-->
62      <div>
63        <h2>Perfil Profesional</h2>
64        <p style="text-align: justify;">Desarrollador de Aplicaciones Multiplataforma (DAM) con experiencia en el diseño
65      </div>
66      <!--Tercera sección-->
67      <div>
68        <h2>Experiencia Laboral</h2>
69        <h3>Programador</h3>
70        <p>Empresa, S.L. | Marzo 2025 - Abril 2025</p>
71      </div>
72      <!--Cuarta sección-->
73      <div>
74        <h2>Educación</h2>
75        <h3>Instituto IES Ribera del Tajo</h3>
76        <p>Estudios en Grado Superior de DAM | 2023 - 2025</p>
77      </div>
78      <!--Quinta sección-->
79      <div>
80        <h2>Habilidades</h2>
81        <ul>
82          <li>Desarrollo de software</li>
83          <li>Desarrollo web</li>
84          <li>Dominio de Adobe Photoshop</li>
85          <li>Desarrollo de aplicaciones móviles</li>
86          <li>Dominio del paquete Office</li>
87        </ul>
88      </div>
89    </main>
90  </body>
91 </html>

```

```

91      <li>Desarrollo web</li>
92      <li>Dominio de Adobe Photoshop</li>
93      <li>Desarrollo de aplicaciones móviles</li>
94      <li>Dominio del paquete Office</li>
95    </ul>
96  </div>
97
98  <!-- Sexta sección -->
99  <div>
100    <h2>Certificaciones</h2>
101    <h3>Curso de HTML5</h3>
102    <p>2024</p>
103
104    <h3>Curso de CSS3</h3>
105    <p>2024</p>
106  </div>
107
108  <!-- Séptima sección -->
109  <div>
110    <h2>Idiomas</h2>
111    <h3>Castellano</h3>
112    <p>Mativo</p>
113
114    <h3>Inglés</h3>
115    <p>Nivel B1</p>
116  </div>
117 </main>
118 </body>
119 </html>

```

Este XHTML cumple la estructura de dicho formato, cerrando todas las etiquetas que utiliza, y determinando en una etiqueta **<html>** que es un formato XHTML.

Para poder realizar el CV, he utilizado varios **<div>** para separar las distintas secciones de la página, y diferentes tipos de títulos, como **<h1>**, **<h2>** y **<h3>** para organizar la información.

También, he agregado diversos estilos, tanto de color como de espacio, para dar un mejor diseño a la página web.



ACTIVIDAD 2

Investiga como utilizar MathML en NetBeans 8.2 e implementa la fórmula matemática utilizando el lenguaje MathML.

$$a = \sqrt{c^2 - b^2}$$

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <math>
3   <mrow>
4     <mi>a</mi>
5     <mo>=</mo>
6     <msqrt>
7       <mrow>
8         <!--Superindices-->
9         <msup>
10          <mi>c</mi>
11          <mn>2</mn>
12        </msup>
13        <mo>-</mo>
14        <msup>
15          <mi>b</mi>
16          <mn>2</mn>
17        </msup>
18      </mrow>
19    </msqrt>
20  </mrow>
21 </math>

```

1. Para crear esta función, primero de todo utilizamos la etiqueta **<math>**, con la cual podremos crear funciones matemáticas.
2. Utilizamos **<mrow>** para poder agrupar elementos matemáticos, y así poder agrupar la función.
3. Usamos **<mi>** para representar caracteres matemáticos (a, x, y...) y **<mo>** para representar símbolos (=, +, -...).
4. Ahora vamos a la parte de crear la raíz cuadrada. Utilizamos **<msqrt>** para crearla, y volvemos a crear una agrupación con **<mrow>**.
5. Ahora usamos **<msup>** para agregar superíndices, como es el 2. Los números se agregan con la etiqueta **<mn>**.

ACTIVIDAD 3

Crear una cara sonriente utilizando SVG.

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>SVG Actividad 3</title>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   </head>
8   <body>
9     <svg width="150" height="150">
10      <circle cx="70" cy="70" r="50" fill="#f5d93b" stroke="#000" stroke-width="5px"/>
11      <!--Ojos-->
12      <circle cx="50" cy="60" r="10" fill="#000"/>
13      <circle cx="90" cy="60" r="10" fill="#000"/>
14      <!--Sonrisa-->
15      <path d="M 45 85 Q 70 110 95 85" stroke="#000" stroke-width="5px" fill="none" />
16    </svg>
17  </body>
18 </html>

```

Se utilizan las etiquetas de XML **<svg>** para poder empezar a crear la figura. Dentro de esta, agrego las figuras que voy a crear. Con el primer **<circle>** creo la forma de la cabeza. Los otros dos **<circle>** los utilizo para ponerle ojos a la cara, junto con sus respectivos atributos (fill, stroke...). Finalmente, para la

sonrisa, utilizo un `<path>`. Este tendrá `fill="none"` para evitar que se rellene la figura.



ACTIVIDAD 4

Establece el formato que debería de tener una estructura xml para realizar una factura con los campos:

- Factura
- Número
- Fecha
- Nombre
- Cliente
- Teléfono cliente
- Dirección cliente
- Concepto
- Total

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <facturas>
3    <factura numero="1">
4      <fecha>20-11-2024</fecha>
5      <nombre>Juan Pérez</nombre>
6      <!--Datos del cliente-->
7      <cliente>
8        <telefono>612345678</telefono>
9        <direccion>Calle Magia 123, Toledo, España</direccion>
10       <concepto>Compra de productos electrónicos</concepto>
11     </cliente>
12     <total>350.50</total>
13   </factura>
14 </facturas>

```

En este XML, podemos definir diferentes facturas, las cuales almacenarán tanto sus datos como los de su cliente. El número se guardará como atributo dentro de la etiqueta `<factura>`.

ACTIVIDAD 5

Implementa una interfaz en NetBeans que se utilicen todos los eventos mencionados en este apartado de eventos.

```

19  */
20  public class PantallaPrincipal extends javax.swing.JFrame {
21
22      /**
23       * Creates new form PantallaPrincipal
24       */
25      public PantallaPrincipal() {
26          initComponents();
27
28          // Estilos para la ventana
29          this.setSize(370, 390);
30          this.setLocationRelativeTo(null);
31          this.setResizable(false);
32          this.setIconImage(new ImageIcon(getClass().getResource("/actividad5interfaz/images/icono.png")).getImage());
33
34          // Padding izquierdo para el jTextField
35          Border borde=BorderFactory.createLineBorder(new Color(51,51,51),3);
36          EmptyBorder padding=(EmptyBorder) BorderFactory.createEmptyBorder(0,10,0,0);
37          CompoundBorder bordeFinal=BorderFactory.createCompoundBorder(borde, padding);
38          jTextFieldNombre.setBorder(bordeFinal);
39          jPasswordField1.setBorder(bordeFinal);
40          jPasswordField1.setEchoChar('*');
41
42          BotonEmpezar.requestFocusInWindow();
43
44      }
45
46      /**
47       * This method is called from within the constructor to initialize the form.
48       * WARNING: Do NOT modify this code. The content of this method is always
49       * regenerated by the Form Editor.
50       */
51      @SuppressWarnings("unchecked")
52      // Generated Code
53
54      private void jTextFieldNombreCaretUpdate(javax.swing.event.CaretEvent evt) {
55          String texto=jTextFieldNombre.getText().toString();
56          if(!texto.equals("Ingresa un nombre...") && !texto.equals("")) {
57              jLabelNombre.setText(texto);
58          }
59      }
60
61      // EVENTOS PARA EL HOVER
62      private void jLabel1MouseEntered(java.awt.event.MouseEvent evt) {
63          BotonEmpezar.setBackground(new Color(90,90,90));
64      }
65
66      private void jLabel1MouseExited(java.awt.event.MouseEvent evt) {
67          BotonEmpezar.setBackground(new Color(60,63,65));
68      }
69
70      // EVENTOS PARA EL MOUSEDOWN DEL BOTÓN
71      private void jLabel1MousePressed(java.awt.event.MouseEvent evt) {
72          BotonEmpezar.setBackground(new Color(30,33,35));
73      }
74
75      private void jLabel1MouseReleased(java.awt.event.MouseEvent evt) {
76          BotonEmpezar.setBackground(new Color(90,90,90));
77      }
78
79      // EVENTOS DE FOCUS PARA EL TEXTFIELD
80      private void jTextFieldNombreFocusGained(java.awt.event.FocusEvent evt) {
81
82      }
83
84      private void jTextFieldNombreFocusLost(java.awt.event.FocusEvent evt) {
85          String texto=jTextFieldNombre.getText().toString();
86          if(texto.equals("")) {
87              jTextFieldNombre.setText("Ingresa un nombre...");
88          }
89      }
90
91      private void jPasswordField1FocusGained(java.awt.event.FocusEvent evt) {
92          String texto=new String(jPasswordField1.getPassword());
93          if(texto.equals("password")) {
94              jPasswordField1.setText("");
95          }
96      }
97
98      private void jPasswordField1FocusLost(java.awt.event.FocusEvent evt) {
99          String texto=new String(jPasswordField1.getPassword());
100         if(texto.equals("")) {
101             jPasswordField1.setText("password");
102         }
103     }
104
105     private void jLabel1MouseClicked(java.awt.event.MouseEvent evt) {
106         String nombre=jTextFieldNombre.getText().toString();
107         String contra=new String(jPasswordField1.getPassword());
108
109         if(!nombre.equals("") && !contra.equals("") && !nombre.equals("Ingresa un nombre...") && !contra.equals("password")) {
110             DialogSeguir d=new DialogSeguir(this, true);
111             d.setVisible(true);
112         }
113     }
114 }

```

```

31  L
32  L
33  L
34  L
35  L
36  L
37  L
38  L
39  L
40  L
41  L
42  L
43  L
44  L
45  L
46  L
47  L
48  L
49  L
50  L
51  L
52  L
53  L
54  L
55  L
56  L
57  L
58  L
59  L
60  L
61  L
62  L
63  L
64  L
65  L
66  L
67  L
68  L
69  L
70  L
71  L
72  L
73  L
74  L
75  L
76  L
77  L
78  L
79  L
80  L
81  L
82  L
83  L
84  L
85  L
86  L
87  L
88  L
89  L
90  L
91  L
92  L
93  L
94  L
95  L
96  L
97  L
98  L
99  L
100 L
101 L
102 L
103 L
104 L
105 L
106 L
107 L
108 L
109 L
110 L
111 L
112 L
113 L
114 L
115 L
116 L
117 L
118 L
119 L
120 L
121 L
122 L
123 L
124 L
125 L
126 L
127 L
128 L
129 L
130 L
131 L
132 L
133 L
134 L
135 L
136 L
137 L
138 L
139 L
140 L
141 L
142 L
143 L
144 L
145 L
146 L
147 L
148 L
149 L
150 L
151 L
152 L
153 L
154 L
155 L
156 L
157 L
158 L
159 L
160 L
161 L
162 L
163 L
164 L
165 L
166 L
167 L
168 L
169 L
170 L
171 L
172 L
173 L
174 L
175 L
176 L
177 L
178 L
179 L
180 L
181 L
182 L
183 L
184 L
185 L
186 L
187 L
188 L
189 L
190 L
191 L
192 L
193 L
194 L
195 L
196 L
197 L
198 L
199 L
200 L
201 L
202 L
203 L
204 L
205 L
206 L
207 L
208 L
209 L
210 L
211 L
212 L
213 L
214 L
215 L
216 L
217 L
218 L
219 L
220 L
221 L
222 L
223 L
224 L
225 L
226 L
227 L
228 L
229 L
230 L
231 L
232 L
233 L
234 L
235 L
236 L
237 L
238 L
239 L
240 L
241 L
242 L
243 L
244 L
245 L
246 L
247 L
248 L
249 L
250 L
251 L
252 L
253 L
254 L
255 L
256 L
257 L
258 L
259 L
260 L
261 L
262 L
263 L
264 L
265 L
266 L
267 L
268 L
269 L
270 L
271 L
272 L
273 L
274 L
275 L
276 L
277 L
278 L
279 L
280 L
281 L
282 L
283 L
284 L
285 L
286 L
287 L
288 L
289 L
290 L
291 L
292 L
293 L
294 L
295 L
296 L
297 L
298 L
299 L
300 L
301 L
302 L
303 L
304 L
305 L
306 L
307 L
308 L
309 L
310 L
311 L
312 L
313 L
314 L
315 L
316 L
317 L
318 L
319 L
320 L
321 L
322 L
323 L
324 L
325 L
326 L
327 L
328 L
329 L
330 L
331 L
332 L
333 L
334 L
335 L
336 L
337 L
338 L
339 L
340 L
341 L
342 L
343 L
344 L
345 L
346 L
347 L
348 L
349 L
350 L
351 L
352 L
353 L
354 L
355 L
356 L
357 L
358 L
359 L
360 L
361 L
362 L
363 L
364 L
365 L
366 L
367 L
368 L
369 L
370 L
371 L
372 L
373 L
374 L
375 L
376 L
377 L
378 L
379 L
380 L
381 L
382 L
383 L
384 L
385 L
386 L
387 L
388 L
389 L
390 L
391 L
392 L
393 L
394 L
395 L
396 L
397 L
398 L
399 L
400 L
401 L
402 L
403 L
404 L
405 L
406 L
407 L
408 L
409 L
410 L
411 L
412 L
413 L
414 L
415 L
416 L
417 L
418 L
419 L
420 L
421 L
422 L
423 L
424 L
425 L
426 L
427 L
428 L
429 L
430 L
431 L
432 L
433 L
434 L
435 L
436 L
437 L
438 L
439 L
440 L
441 L
442 L
443 L
444 L
445 L
446 L
447 L
448 L
449 L
450 L
451 L
452 L
453 L
454 L
455 L
456 L
457 L
458 L
459 L
460 L
461 L
462 L
463 L
464 L
465 L
466 L
467 L
468 L
469 L
470 L
471 L
472 L
473 L
474 L
475 L
476 L
477 L
478 L
479 L
480 L
481 L
482 L
483 L
484 L
485 L
486 L
487 L
488 L
489 L
490 L
491 L
492 L
493 L
494 L
495 L
496 L
497 L
498 L
499 L
500 L
501 L
502 L
503 L
504 L
505 L
506 L
507 L
508 L
509 L
510 L
511 L
512 L
513 L
514 L
515 L
516 L
517 L
518 L
519 L
520 L
521 L
522 L
523 L
524 L
525 L
526 L
527 L
528 L
529 L
530 L
531 L
532 L
533 L
534 L
535 L
536 L
537 L
538 L
539 L
540 L
541 L
542 L
543 L
544 L
545 L
546 L
547 L
548 L
549 L
550 L
551 L
552 L
553 L
554 L
555 L
556 L
557 L
558 L
559 L
560 L
561 L
562 L
563 L
564 L
565 L
566 L
567 L
568 L
569 L
570 L
571 L
572 L
573 L
574 L
575 L
576 L
577 L
578 L
579 L
580 L
581 L
582 L
583 L
584 L
585 L
586 L
587 L
588 L
589 L
590 L
591 L
592 L
593 L
594 L
595 L
596 L
597 L
598 L
599 L
600 L
601 L
602 L
603 L
604 L
605 L
606 L
607 L
608 L
609 L
610 L
611 L
612 L
613 L
614 L
615 L
616 L
617 L
618 L
619 L
620 L
621 L
622 L
623 L
624 L
625 L
626 L
627 L
628 L
629 L
630 L
631 L
632 L
633 L
634 L
635 L
636 L
637 L
638 L
639 L
640 L
641 L
642 L
643 L
644 L
645 L
646 L
647 L
648 L
649 L
650 L
651 L
652 L
653 L
654 L
655 L
656 L
657 L
658 L
659 L
660 L
661 L
662 L
663 L
664 L
665 L
666 L
667 L
668 L
669 L
670 L
671 L
672 L
673 L
674 L
675 L
676 L
677 L
678 L
679 L
680 L
681 L
682 L
683 L
684 L
685 L
686 L
687 L
688 L
689 L
690 L
691 L
692 L
693 L
694 L
695 L
696 L
697 L
698 L
699 L
700 L
701 L
702 L
703 L
704 L
705 L
706 L
707 L
708 L
709 L
710 L
711 L
712 L
713 L
714 L
715 L
716 L
717 L
718 L
719 L
720 L
721 L
722 L
723 L
724 L
725 L
726 L
727 L
728 L
729 L
730 L
731 L
732 L
733 L
734 L
735 L
736 L
737 L
738 L
739 L
740 L
741 L
742 L
743 L
744 L
745 L
746 L
747 L
748 L
749 L
750 L
751 L
752 L
753 L
754 L
755 L
756 L
757 L
758 L
759 L
760 L
761 L
762 L
763 L
764 L
765 L
766 L
767 L
768 L
769 L
770 L
771 L
772 L
773 L
774 L
775 L
776 L
777 L
778 L
779 L
780 L
781 L
782 L
783 L
784 L
785 L
786 L
787 L
788 L
789 L
790 L
791 L
792 L
793 L
794 L
795 L
796 L
797 L
798 L
799 L
800 L
801 L
802 L
803 L
804 L
805 L
806 L
807 L
808 L
809 L
810 L
811 L
812 L
813 L
814 L
815 L
816 L
817 L
818 L
819 L
820 L
821 L
822 L
823 L
824 L
825 L
826 L
827 L
828 L
829 L
830 L
831 L
832 L
833 L
834 L
835 L
836 L
837 L
838 L
839 L
840 L
841 L
842 L
843 L
844 L
845 L
846 L
847 L
848 L
849 L
850 L
851 L
852 L
853 L
854 L
855 L
856 L
857 L
858 L
859 L
860 L
861 L
862 L
863 L
864 L
865 L
866 L
867 L
868 L
869 L
870 L
871 L
872 L
873 L
874 L
875 L
876 L
877 L
878 L
879 L
880 L
881 L
882 L
883 L
884 L
885 L
886 L
887 L
888 L
889 L
890 L
891 L
892 L
893 L
894 L
895 L
896 L
897 L
898 L
899 L
900 L
901 L
902 L
903 L
904 L
905 L
906 L
907 L
908 L
909 L
910 L
911 L
912 L
913 L
914 L
915 L
916 L
917 L
918 L
919 L
920 L
921 L
922 L
923 L
924 L
925 L
926 L
927 L
928 L
929 L
930 L
931 L
932 L
933 L
934 L
935 L
936 L
937 L
938 L
939 L
940 L
941 L
942 L
943 L
944 L
945 L
946 L
947 L
948 L
949 L
950 L
951 L
952 L
953 L
954 L
955 L
956 L
957 L
958 L
959 L
960 L
961 L
962 L
963 L
964 L
965 L
966 L
967 L
968 L
969 L
970 L
971 L
972 L
973 L
974 L
975 L
976 L
977 L
978 L
979 L
980 L
981 L
982 L
983 L
984 L
985 L
986 L
987 L
988 L
989 L
990 L
991 L
992 L
993 L
994 L
995 L
996 L
997 L
998 L
999 L
1000 L

```

EVENTO DE MOUSEMOVE: este está implementado en el Dialog que se despliega al pulsar el botón. Lo que este hará es obtener la posición X e Y del ratón, y hacer que un conjunto de jPanels lo siga. Gracias a la clase **Timer**, se pondrá un delay, para darle un mejor efecto.

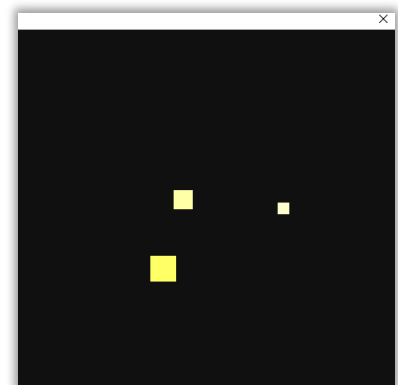
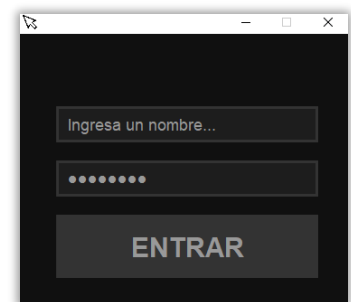
EVENTO DE MOUSEDOWN: este se producirá al presionar el botón de ENTRAR, ya que al hacer esto, se cambiará el color de fondo de dicho botón, para hacer entender al usuario que está interactuando con el botón en cuestión.

CHANGE: se produce al escribir texto en el Campo de texto de nombre. Cada vez que se modifique el contenido de este campo, se podrá visualizar en un JLabel que está situado encima suyo.

CLICK: al hacer Click en el cubo amarillo grande, se cambiará el fondo por 20 milisegundos, para que el usuario entienda que su acción ha sido ejecutada.

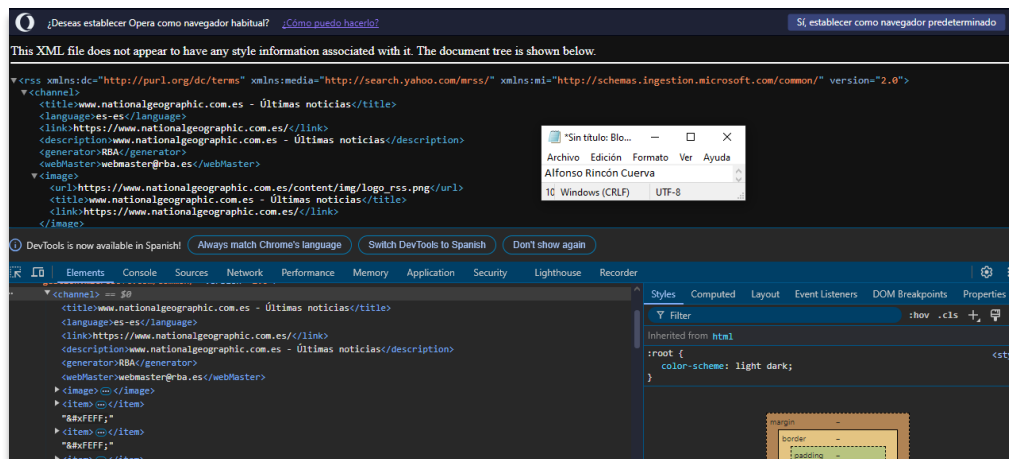
GETFOCUS: este evento se produce en los campos de texto, para darle un efecto de placeholder. Una vez que el campo de texto obtiene el foco, se elimina el contenido por defecto que tiene.

LOSTFOCUS: al contrario que el anterior, una vez que el campo de texto pierde el foco, si no se ha insertado contenido, se restaurará el contenido que estos campos tienen por defecto.



ACTIVIDAD 6

Para difundir información a los usuarios que se han suscrito a una fuente de contenidos, es recomendable utilizar el lenguaje de sindicación realmente simple (RSS). Busca en la web (por ejemplo, en cualquier página de Google) un ejemplo de este tipo de lenguaje que incluya, al menos, las etiquetas `<title>`, `<link>` y `<description>`. ¿Qué tipo de contenido se especifica en cada una de estas etiquetas?



Como podemos ver, la página de National Geographic tiene sus valores RSS para determinar su información. Podemos encontrar las siguientes etiquetas:

- **<title>**: es el título de la web.
- **<language>**: determina el lenguaje en el que se muestra la web. En este caso, es español.
- **<link>**: es la URL en la que nos encontramos actualmente.
- **<generator>**: indica el nombre del software utilizado para crearla.

ACTIVIDAD 7

El siguiente código XML contiene 3 errores. Localiza cada uno de ellos e indica dónde se encuentra y cómo resolverlo.

El código corregido es el siguiente:

```

<?xml version="1.0" encoding="UTF-8" ?>
<capitales>
  <capital>
    <nombre>Pekín</nombre>
    <pais>China</pais>
  </capital>
  <capital>
    <nombre>Roma</nombre>
  </capital>
</capitales>

```

```
<pais>Italia</pais>
</capital>
<capital>
  <nombre>Lisboa</nombre>
  <pais>Portugal</pais>
</capital>
<capital>
  <nombre>Lima</nombre>
  <pais>Perú</pais>
</capital>
</capitales>
```

Los 3 errores que se encontraron en el XML original, y que han sido solucionados fueron los siguientes:

1. El primer error estaba en la primera línea:

```
<?xml version="1.0" encoding="UTF-8"?>
```

El carácter que produce este error es la `¡`, ya que al definir un formato de tipo XML, no puede seguir esa estructura.

2. El segundo error está en la línea 13.

```
</pais>Portugal</pais>
```

Como podemos ver, tanto la etiqueta de apertura como la de cierre contienen el `/`. Solo las etiquetas de cierre o las etiquetas vacías pueden tenerlo.

3. El tercer error se encuentra en la última línea de capital:

```
<capital>
```

```
....
```

```
<capital>
```

Esta etiqueta no se cierra en ningún momento, lo cual es un error y debería cerrarse, al igual que el resto de etiquetas.

También hay que destacar algo que no es un error, pero visualmente empeora el aspecto del código, y es que hay algunas líneas que están mal tabuladas.

ACTIVIDAD 8.

Una nueva empresa de suministro eléctrico nos ha solicitado el diseño de un logo que represente su marca. Esta multinacional se caracteriza por su sencillez a la hora de realizar los contratos y la rapidez a la hora de proporcionar el suministro. Desea una imagen representativa que esté formada por un círculo con un rayo en su interior. Esta imagen debe ser escalable y de gran calidad, por lo que se realizará utilizando el lenguaje SVG.

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>SVG Rayo Actividad 8</title>
5    <meta charset="UTF-8">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7  </head>
8  <body>
9    <svg width="200" height="200" xmlns="http://www.w3.org/2000/svg">
10     <circle fill="none" stroke="#888" stroke-width="8px" cy="100" cx="100" r="80"></circle>
11     <!-- Rayo -->
12     <polygon points="95,50 85,90 105,90 80,130 110,130 95,160 125,110 110,110 125,50" fill="#FFA500" />
13   </svg>
14 </body>
15 </html>
16

```

En este código utilizo la etiqueta **<circle>** para crear el círculo que cubre al logo. Este será de color gris oscuro. Para dibujar el rayo, utilizo la etiqueta **<polygon>**, dibujando varios puntos hasta crear una silueta parecida a un rayo. Se rellena de color amarillo con **fill**.



BIBLIOGRAFÍA

National Geographic. (2024, 20 noviembre). www.nationalgeographic.com.es.
<https://www.nationalgeographic.com.es/>