# Practical Exercises

1. Using HTML, CSS create a styled checkbox with animation on state change.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a styled checkbox with animation on state change</title>  </head>  <body>  <div class="w3rcontainer"><strong>Preview:</strong>  <div class="checkbox-container">  <input class="checkbox-input" id="oranges" type="checkbox" />  <label class="checkbox" for="oranges">  <span>  <svg width="12px" height="10px">  <use xlink:href="#check"></use>  </svg>  </span>  <span>Oranges</span>  </label>  <input class="checkbox-input" id="bananas" type="checkbox" />  <label class="checkbox" for="bananas">  <span>  <svg width="12px" height="10px">  <use xlink:href="#check"></use>  </svg>  </span>  <span>Bananas</span>  </label>  </div>  </div>  </body>  </html> | <style>  .w3rcontainer{  border: 1px solid #cccfdb;  border-radius: 2px;  }  .hover-underline-animation {  display: inline-block;  position: relative;  color: #0087ca;  }  .hover-underline-animation:after {  content: '';  position: absolute;  width: 100%;  transform: scaleX(0);  height: 2px;  bottom: 0;  left: 0;  background-color: #0087ca;  transform-origin: bottom right;  transition: transform 0.25s ease-out;  }  .hover-underline-animation:hover:after {  transform: scaleX(1);  transform-origin: bottom left;  }  </style> |

1. Using HTML, CSS create a styled checkbox with animation on state change.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS change the styling of text selection</title>  </head>  <body>  <div class="w3rcontainer"><strong>Preview:</strong>  <p class="custom-text-selection">Select some of this text.</p>  </div>  </body>  </html> | <style>  ::selection {  background: orange;  color: black;  }  .custom-text-selection::selection {  background: green;  color: white;  }  </style> |

1. Using HTML, CSS create a styled checkbox with animation on state change.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create display an image overlay effect on hover</title>  </head>  <body>  <figure class="hover-img"><strong>Preview:</strong><br>  <img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-2.jpeg"/>  <figcaption>  <h3>Pansy <br/>Flower</h3>  </figcaption>  </figure>  </body>  </html> | <style>  .hover-img {  background-color: #000;  color: #fff;  display: inline-block;  margin: 8px;  max-width: 250px;  min-width: 240px;  overflow: hidden;  position: relative;  text-align: center;  width: 100%;  }  .hover-img \* {  box-sizing: border-box;  transition: all 0.45s ease;  }  .hover-img:before,  .hover-img:after {  background-color: rgba(0, 0, 0, 0.5);  border-top: 32px solid rgba(0, 0, 0, 0.5);  border-bottom: 32px solid rgba(0, 0, 0, 0.5);  position: absolute;  top: 0;  bottom: 0;  left: 0;  right: 0;  content: '';  transition: all 0.3s ease;  z-index: 1;  opacity: 0;  transform: scaleY(2);  }  .hover-img img {  vertical-align: top;  max-width: 100%;  backface-visibility: hidden;  }  .hover-img figcaption {  position: absolute;  top: 0;  bottom: 0;  left: 0;  right: 0;  align-items: center;  z-index: 1;  display: flex;  flex-direction: column;  justify-content: center;  line-height: 1.1em;  opacity: 0;  z-index: 2;  transition-delay: 0.1s;  font-size: 24px;  font-family: sans-serif;  font-weight: 400;  letter-spacing: 1px;  text-transform: uppercase;  }  .hover-img:hover:before,  .hover-img:hover:after {  transform: scale(1);  opacity: 1;  }  .hover-img:hover > img {  opacity: 0.7;  }  .hover-img:hover figcaption {  opacity: 1;  }  </style> |

1. Using HTML, CSS create a styled checkbox with animation on state change.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create display a text on top of an image using an overlay</title>  </head>  <body>  <div>  <h3 class="text-overlay">Hello, World</h3>  <img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-1.jpeg" height="200" width="200"></div>  </body>  </html> | <style>    div {  position: relative;  }  .text-overlay {  position: absolute;  top: 0;  left: 0;  padding: 1rem;  font-size: 2rem;  font-weight: 100;  color: white;  backdrop-filter: blur(8px) brightness(80%);  }  </style> |

1. Using HTML, CSS create a list with floating headings for each section

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a list with floating headings for each section</title>  </head>  <body>  <strong>Preview:</strong>  <div class="container">  <div class="floating-stack">  <dl>  <dt>A</dt>  <dd>Aaban</dd>  <dd>Aabel</dd>  <dd>Aabheer</dd>  <dd>Aadam</dd>  <dt>B</dt>  <dd>Baabul</dd>  <dd>Baalaji</dd>  <dd>Baalkrishan</dd>  <dd>Baanke Bihaari</dd>  <dt>C</dt>  <dd>Caddam</dd>  <dd>Cameroon</dd>  <dd>Campbell</dd>  <dd>Cane</dd>    <dt>D</dt>  <dd>Daanesh</dd>  <dd>Dadvar</dd>  <dd>Daghan</dd>  <dd>Daivya</dd>  <dd>Daamini</dd>  <dt>E</dt>  <dd>Eadmer</dd>  <dd>Earnest</dd>  <dd>Eddward</dd>  <dd>Edmond</dd>  </dl>  </div>  </div>  </body>  </html> | <style>    .container {  display: grid;  place-items: center;  min-height: 400px;  }  .floating-stack {  background: #3365a4;  color: #fff;  height: 80vh;  width: 320px;  border-radius: 1rem;  overflow-y: auto;  }  .floating-stack > dl {  margin: 0 0 1rem;  display: grid;  grid-template-columns: 2.5rem 1fr;  align-items: center;  }  .floating-stack dt {  position: sticky;  top: 0.5rem;  left: 0.5rem;  font-weight: bold;  background: #263238;  color: #cfd8dc;  height: 2rem;  width: 2rem;  border-radius: 50%;  padding: 0.25rem 1rem;  grid-column: 1;  display: inline-flex;  align-items: center;  justify-content: center;  box-sizing: border-box;  }  .floating-stack dd {  grid-column: 2;  margin: 0;  padding: 0.75rem;  }  .floating-stack > dl:first-of-type > dd:first-of-type {  margin-top: 0.25rem;  }  </style> |

1. Using HTML, CSS create a custom hover and focus effect for navigation items, using CSS transformations.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a custom hover and focus effect for navigation items, using CSS transformations</title>  </head>  <body>  <div class="w3rcontainer"><strong>Preview:</strong>  <div class="hover-div">  <ul>  <li><a href="#">Home</a></li>  <li><a href="#">About</a></li>  <li><a href="#">Contact</a></li>  </ul>  </div>  </div>  </body>  </html> | <style>  .w3rcontainer{  border: 1px solid #cccfdb;  border-radius: 2px;  }  .hover-div ul {  list-style: none;  margin: 0;  padding: 0;  overflow: hidden;  }  .hover-div li {  float: left;  }  .hover-div li a {  position: relative;  display: block;  color: #fff;  text-align: center;  padding: 8px 12px;  text-decoration: none;  z-index: 0;  }  li a:before {  position: absolute;  content: "";  width: 100%;  height: 100%;  bottom: 0;  left: 0;  background-color: #F68026;  z-index: -1;  transform: scale(0);  transition: transform 0.5s ease-in-out;  }  li a:hover:before,  li a:focus:before {  transform: scale(1);  }  </style> |

1. Using HTML, CSS create a scrollable container that will snap on elements when scrolling.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a scrollable container that will snap on elements when scrolling</title>  </head>  <body>  <strong>Preview:</strong>  <div class="vertical-snap">  <a href="#"><img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-1.jpeg"></a>  <a href="#"><img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-2.jpeg"></a>  <a href="#"><img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-3.jpeg"></a>  <a href="#"><img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-4.jpeg"></a>  <a href="#"><img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-5.jpeg"></a>  <a href="#"><img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-6.jpeg"></a>  </div>  </body>  </html> | <style>  .w3rcontainer{  border: 1px solid #cccfdb;  border-radius: 2px;  }  .vertical-snap {  margin: 0 auto;  display: grid;  grid-auto-flow: row;  gap: 1rem;  width: calc(180px + 1rem);  padding: 1rem;  height: 480px;  overflow-y: auto;  overscroll-behavior-y: contain;  scroll-snap-type: y mandatory;  }  .vertical-snap > a {  scroll-snap-align: center;  }  .vertical-snap img {  width: 180px;  object-fit: contain;  border-radius: 1rem;  }  </style> |

1. Using HTML, CSS create a scrollable container that will snap on elements when scrolling.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS, JavaScript create a hover effect where the gradient follows the mouse cursor</title>  </head>  <body>  <strong>Preview:</strong><br>  <button class="mouse-cursor-gradient-tracking">  <span>Hover me</span></button>  </body>  </html> | <style>  .mouse-cursor-gradient-tracking {  position: relative;  background: #7983ff;  padding: 0.5rem 1rem;  font-size: 1.2rem;  border: none;  color: white;  cursor: pointer;  outline: none;  overflow: hidden;  }  .mouse-cursor-gradient-tracking span {  position: relative;  }  .mouse-cursor-gradient-tracking:before {  --size: 0;  content: '';  position: absolute;  left: var(--x);  top: var(--y);  width: var(--size);  height: var(--size);  background: radial-gradient(circle closest-side, red, transparent);  transform: translate(-50%, -50%);  transition: width 0.2s ease, height 0.2s ease;  }  .mouse-cursor-gradient-tracking:hover:before {  --size: 200px;  }  }  </style> |

1. Using HTML, CSS apply a perspective transform with a hover animation to an element.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS apply a perspective transform with a hover animation to an element</title>  </head>  <body>  <strong>Preview:</strong><br>  <div class="card-container">  <div class="image-card perspective-left"></div>  <div class="image-card perspective-right"></div>  </div>  </body>  </html> | <style>  .w3rcontainer{  border: 1px solid #cccfdb;  border-radius: 2px;  }  .image-card {  display: inline-block;  box-sizing: border-box;  margin: 1rem;  width: 240px;  height: 320px;  padding: 8px;  border-radius: 1rem;  background: url("https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-2.jpeg");  box-shadow: rgba(0, 0, 0, 0.25) 0px 25px 50px -12px;  }  .perspective-left {  transform: perspective(1500px) rotateY(15deg);  transition: transform 1s ease 0s;  }  .perspective-left:hover {  transform: perspective(3000px) rotateY(5deg);  }  .perspective-right {  transform: perspective(1500px) rotateY(-15deg);  transition: transform 1s ease 0s;  }  .perspective-right:hover {  transform: perspective(3000px) rotateY(-5deg);  }  </style> |

1. Using HTML, CSS create a staggered animation for the elements of a list.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a staggered animation for the elements of a list.</title>  </head>  <body>  <strong>Preview:</strong><br>  <div class="container">  <input type="checkbox" name="menu" id="menu" class="menu-toggler">  <label for="menu" class="menu-toggler-label">Menu</label>  <ul class="stagger-menu">  <li style="--i: 0">Home</li>  <li style="--i: 1">Privacy</li>  <li style="--i: 2">About</li>  <li style="--i: 3">Contact</li>  <li style="--i: 4">Feedback</li>  </ul>  </div>  </body>  </html> | <style>  .container {  overflow-x: hidden;  width: 100%;  }  .menu-toggler {  display: none;  }  .menu-toggler-label {  cursor: pointer;  font-size: 20px;  font-weight: bold;  }  .stagger-menu {  list-style-type: none;  margin: 11px 0;  padding: 0;  }  .stagger-menu li {  margin-bottom: 8px;  font-size: 18px;  opacity: 0;  transform: translateX(100%);  transition-property: opacity, transform;  transition-duration: 0.3s;  transition-timing-function: cubic-bezier(0.750, -0.015, 0.565, 1.055);  }  .menu-toggler:checked ~ .stagger-menu li {  opacity: 1;  transform: translateX(0);  transition-delay: calc(0.055s \* var(--i));  }  </style> |

1. Using HTML, CSS, JavaScript create a typewriter effect animation.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS, JavaScript create a typewriter effect animation</title>  </head>  <body>  <strong>Preview:</strong><br>  <div class="typewriter-effect">  <div class="text" id="typewriter-text"></div>  </body>  </html> | <style>  .typewriter-effect {  display: flex;  justify-content: center;  font-family: monospace;  }  .typewriter-effect > .text {  max-width: 0;  animation: typing 3s steps(var(--characters)) infinite;  white-space: nowrap;  overflow: hidden;  }  .typewriter-effect:after {  content: " |";  animation: blink 1s infinite;  animation-timing-function: step-end;  }  @keyframes typing {  75%,  100% {  max-width: calc(var(--characters) \* 1ch);  }  }  @keyframes blink {  0%,  75%,  100% {  opacity: 1;  }  25% {  opacity: 0;  }  }  </style> |

1. Using HTML, CSS create a pulse effect loader animation using the animation-delay property.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a pulse effect loader animation using the animation-delay property</title>  </head>  <body>  <strong>Preview:</strong><br>  <div class="ripple-loader">  <div>w3r</div>  <div>w3r</div>  </div>  </body>  </html> | <style>  .ripple-loader {  position: relative;  width: 64px;  height: 64px;  }  .ripple-loader div {  position: absolute;  border: 4px solid #AA4A44;  border-radius: 50%;  animation: ripple-loader 1s ease-out infinite;  }  .ripple-loader div:nth-child(2) {  animation-delay: -0.5s;  }  @keyframes ripple-loader {  0% {  top: 32px;  left: 32px;  width: 0;  height: 0;  opacity: 1;  }  100% {  top: 0;  left: 0;  width: 64px;  height: 64px;  opacity: 0;  }  }  </style> |

1. Using HTML, CSS create a bouncing loader animation.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a bouncing loader animation</title>  </head>  <body>  <strong>Preview:</strong><br>  <div class="bouncing-loader">  <div> w</div>  <div> 3</div>  <div> r</div>  <div> e</div>  <div> s</div>  <div> o</div>  <div> u</div>  <div> r</div>  <div> c</div>  <div> e</div>  </div>  </body>  </html> | @keyframes bouncing-loader {  to {  opacity: 0.1;  transform: translate3d(0, -16px, 0);  }  }  .bouncing-loader {  display: flex;  justify-content: center;  }  .bouncing-loader > div {  width: 16px;  height: 16px;  margin: 3rem 0.2rem;  background: #8385aa;  border-radius: 50%;  text: center;  animation: bouncing-loader 0.6s infinite alternate;  }  .bouncing-loader > div:nth-child(2) {  animation-delay: 0.2s;  }  .bouncing-loader > div:nth-child(3) {  animation-delay: 0.4s;  }  .bouncing-loader > div:nth-child(4) {  animation-delay: 0.6s;  }  .bouncing-loader > div:nth-child(5) {  animation-delay: 0.8s;  }  .bouncing-loader > div:nth-child(6) {  animation-delay: 0.10s;  }  .bouncing-loader > div:nth-child(7) {  animation-delay: 0.2s;  }  .bouncing-loader > div:nth-child(8) {  animation-delay: 0.4s;  }  .bouncing-loader > div:nth-child(9) {  animation-delay: 0.5s;  } |

1. Using HTML, CSS create a rotate effect for the image on hover.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create a rotate effect for the image on hover</title>  </head>  <body>  <strong>Preview:</strong><br>  <figure class="hover-rotate">  <img src="https://www.w3resource.com/html-css-exercise/html-css-practical-exercises/flower-2.jpeg" height="100" width="100"/></figure>  </body>  </html> | <style>  .hover-rotate {  overflow: hidden;  margin: 8px;  min-width: 240px;  max-width: 320px;  width: 100%;  }  .hover-rotate img {  transition: all 0.3s;  box-sizing: border-box;  max-width: 100%;  }  .hover-rotate:hover img {  transform: scale(1.3) rotate(5deg);  }  </style> |

1. Using HTML, CSS create an animated underline effect when the user hovers over the text.

|  |  |
| --- | --- |
| <!--License: https://bit.ly/3GjrtVF-->  <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width">  <title>Using HTML, CSS create an animated underline effect when the user hovers over the text</title>  </head>  <body>  <strong>Preview:</strong><br>  <p class="hover-underline-animation">Hover w3resource text to see the effect!</p>  </body>  </html> | <style>  .w3rcontainer{  border: 1px solid #cccfdb;  border-radius: 2px;  }  .hover-underline-animation {  display: inline-block;  position: relative;  color: #0087ca;  }  .hover-underline-animation:after {  content: '';  position: absolute;  width: 100%;  transform: scaleX(0);  height: 2px;  bottom: 0;  left: 0;  background-color: #0087ca;  transform-origin: bottom right;  transition: transform 0.25s ease-out;  }  .hover-underline-animation:hover:after {  transform: scaleX(1);  transform-origin: bottom left;  }  </style> |