<script>

//detectar cara cercana

function closestEdge(x,y,w,h) {

var topEdgeDist = distMetric(x,y,w/2,0);

var bottomEdgeDist = distMetric(x,y,w/2,h);

var leftEdgeDist = distMetric(x,y,0,h/2);

var rightEdgeDist = distMetric(x,y,w,h/2);

var min = Math.min(topEdgeDist,bottomEdgeDist,leftEdgeDist,rightEdgeDist);

switch (min) {

case leftEdgeDist:

return "left";

case rightEdgeDist:

return "right";

case topEdgeDist:

return "top";

case bottomEdgeDist:

return "bottom";

}

}

function closestEdge2(x,y,w,h){

}

//Distance Formula

function distMetric(x,y,x2,y2) {

var xDiff = x - x2;

var yDiff = y - y2;

return (xDiff \* xDiff) + (yDiff \* yDiff);

}

var boxes = document.querySelectorAll(".noticiaItem");

for(var i = 0; i < boxes.length; i++){

boxes[i].onmouseenter = function(e){

//alert(this.offsetTop);

var x = e.pageX - $(this).offset().left;

var y = e.pageY - $(this).offset().top;

//alert("valores: ".concat(x," ",y," ",this.clientWidth," ",this.clientHeight));

var edge = closestEdge(x,y,this.clientWidth, this.clientHeight);

var overlay = this.childNodes[1];

var image = this.childNodes[0];

switch(edge){

case "left":

//tween overlay from the left

overlay.style.top = "0%";

overlay.style.left = "-100%";

TweenMax.to(overlay, .5, {left: '0%'});

TweenMax.to(image, .5, {scale: 1.2});

break;

case "right":

overlay.style.top = "0%";

overlay.style.left = "100%";

//tween overlay from the right

TweenMax.to(overlay, .5, {left: '0%'});

TweenMax.to(image, .5, {scale: 1.2});

break;

case "top":

overlay.style.top = "-100%";

overlay.style.left = "0%";

//tween overlay from the right

TweenMax.to(overlay, .5, {top: '0%'});

TweenMax.to(image, .5, {scale: 1.2});

break;

case "bottom":

overlay.style.top = "100%";

overlay.style.left = "0%";

//tween overlay from the right

TweenMax.to(overlay, .5, {top: '0%'});

TweenMax.to(image, .5, {scale: 1.2});

break;

}

};

boxes[i].onmouseleave = function(e){

var x = e.pageX - $(this).offset().left;

var y = e.pageY - $(this).offset().top;

var edge = closestEdge(x,y,this.clientWidth, this.clientHeight);

var overlay = this.childNodes[1];

var image = this.childNodes[0];

switch(edge){

case "left":

TweenMax.to(overlay, .5, {left: '-100%'});

TweenMax.to(image, .5, {scale: 1.0});

break;

case "right":

TweenMax.to(overlay, .5, {left: '100%'});

TweenMax.to(image, .5, {scale: 1.0});

break;

case "top":

TweenMax.to(overlay, .5, {top: '-100%'});

TweenMax.to(image, .5, {scale: 1.0});

break;

case "bottom":

TweenMax.to(overlay, .5, {top: '100%'});

TweenMax.to(image, .5, {scale: 1.0});

break;

}

};

}

</script>