# Chương 1: Cấu tạo nguyên tử

**Bài 1: Thành phần của nguyên tử:** https://phet.colorado.edu/sims/html/build-an-atom/latest/build-an-atom\_all.html?locale=vi

**Bài 2: Nguyên tố hoá học:** https://phet.colorado.edu/sims/html/isotopes-and-atomic-mass/latest/isotopes-and-atomic-mass\_all.html?locale=vi

**Bài 3: Cấu trúc lớp vỏ electron nguyên tử:** https://phet.colorado.edu/sims/html/atomic-interactions/latest/atomic-interactions\_all.html?locale=vi

**Bài 4: Ôn tập chương 1:** https://phet.colorado.edu/sims/html/build-an-atom/latest/build-an-atom\_all.html?locale=vi

# Chương 2: Bảng tuần hoàn và định luật tuần hoàn

**Bài 5: Cấu tạo bảng tuần hoàn:**

Bảng tuần hoàn 3D:

Source code :

Html : <!DOCTYPE html>

<html lang="en" >

<head>

<meta charset="UTF-8">

<title>CodePen - Periodic Table of Elements</title>

<link rel="stylesheet" href="./style.css">

</head>

<body>

<!-- partial:index.partial.html -->

<div id="container"></div>

<div id="info">Bảng Tuần Hoàn Hóa Học 3D LimVA</div>

<div id="menu">

<button id="table">Bảng Cổ Điển</button>

<button id="sphere">Hình Cầu</button>

<button id="helix">Đường Xoắn Ốc</button>

<button id="grid">Dạng Lưới</button>

</div>

<!-- partial -->

<script src='https://cdnjs.cloudflare.com/ajax/libs/three.js/r67/three.min.js'></script><script src="./script.js"></script>

</body>

</html>

Css: html, body {

height: 100%;

}

body {

background-color: #000000;

margin: 0;

font-family: Helvetica, sans-serif;

overflow: hidden;

}

a {

color: #ffffff;

}

#info {

position: absolute;

width: 100%;

color: #ffffff;

padding: 5px;

font-family: Monospace;

font-size: 13px;

font-weight: bold;

text-align: center;

z-index: 1;

}

#menu {

position: absolute;

bottom: 20px;

width: 100%;

text-align: center;

font-family: verdana,Tahoma,Arial,Hei,"Microsoft Yahei",SimHei;

}

.element {

width: 120px;

height: 160px;

box-shadow: 0px 0px 12px rgba(0,255,255,0.5);

border: 1px solid rgba(127,255,255,0.25);

text-align: center;

cursor: default;

}

.element:hover {

box-shadow: 0px 0px 12px rgba(0,255,255,0.75);

border: 1px solid rgba(127,255,255,0.75);

}

.element .number {

position: absolute;

top: 20px;

right: 20px;

font-size: 12px;

color: rgba(127,255,255,0.75);

}

.element .symbol {

position: absolute;

top: 40px;

left: 0px;

right: 0px;

font-size: 60px;

font-weight: bold;

color: rgba(255,255,255,0.75);

text-shadow: 0 0 10px rgba(0,255,255,0.95);

}

.element .details {

position: absolute;

bottom: 15px;

left: 0px;

right: 0px;

font-size: 12px;

color: rgba(127,255,255,0.75);

}

button {

color: rgba(127,255,255,0.75);

background: transparent;

outline: 1px solid rgba(127,255,255,0.75);

border: 0px;

padding: 5px 10px;

cursor: pointer;

}

button:hover {

background-color: rgba(0,255,255,0.5);

}

button:active {

color: #000000;

background-color: rgba(0,255,255,0.75);

}

JS:

// tween.js - https://github.com/sole/tween.js

'use strict';

var TWEEN = TWEEN || function() {

var a = [];

return {

REVISION: "7",

getAll: function() {

return a

},

removeAll: function() {

a = []

},

add: function(c) {

a.push(c)

},

remove: function(c) {

c = a.indexOf(c);

- 1 !== c && a.splice(c, 1)

},

update: function(c) {

if (0 === a.length)

return !1;

for (var b = 0, d = a.length, c = void 0 !== c ? c : Date.now(); b < d;)

a[b].update(c) ? b++ : (a.splice(b, 1), d--);

return !0

}

}

}();

TWEEN.Tween = function(a) {

var c = {}, b = {}, d = 1E3, e = 0, f = null, h = TWEEN.Easing.Linear.None, r = TWEEN.Interpolation.Linear, k = [], l = null, m=!1, n = null, p = null;

this.to = function(a, c) {

null !== c && (d = c);

b = a;

return this

};

this.start = function(d) {

TWEEN.add(this);

m=!1;

f = void 0 !== d ? d : Date.now();

f += e;

for (var g in b)

if (null !== a[g]) {

if (b[g]instanceof Array) {

if (0 === b[g].length)

continue;

b[g] = [a[g]].concat(b[g])

}

c[g] = a[g]

}

return this

};

this.stop = function() {

TWEEN.remove(this);

return this

};

this.delay = function(a) {

e = a;

return this

};

this.easing =

function(a) {

h = a;

return this

};

this.interpolation = function(a) {

r = a;

return this

};

this.chain = function() {

k = arguments;

return this

};

this.onStart = function(a) {

l = a;

return this

};

this.onUpdate = function(a) {

n = a;

return this

};

this.onComplete = function(a) {

p = a;

return this

};

this.update = function(e) {

if (e < f)

return !0;

!1 === m && (null !== l && l.call(a), m=!0);

var g = (e - f) / d, g = 1 < g ? 1: g, i = h(g), j;

for (j in c) {

var s = c[j], q = b[j];

a[j] = q instanceof Array ? r(q, i) : s + (q - s) \* i

}

null !== n && n.call(a, i);

if (1 == g) {

null !== p && p.call(a);

g = 0;

for (i = k.length; g <

i; g++)

k[g].start(e);

return !1

}

return !0

}

};

TWEEN.Easing = {

Linear: {

None: function(a) {

return a

}

},

Quadratic: {

In: function(a) {

return a \* a

},

Out: function(a) {

return a \* (2 - a)

},

InOut: function(a) {

return 1 > (a\*=2) ? 0.5 \* a \* a : - 0.5 \* (--a \* (a - 2) - 1)

}

},

Cubic: {

In: function(a) {

return a \* a \* a

},

Out: function(a) {

return --a \* a \* a + 1

},

InOut: function(a) {

return 1 > (a\*=2) ? 0.5 \* a \* a \* a : 0.5 \* ((a -= 2) \* a \* a + 2)

}

},

Quartic: {

In: function(a) {

return a \* a \* a \* a

},

Out: function(a) {

return 1 - --a \* a \* a \* a

},

InOut: function(a) {

return 1 > (a\*=2) ? 0.5 \* a \* a \* a \* a : - 0.5 \* ((a -= 2) \* a \* a \* a - 2)

}

},

Quintic: {

In: function(a) {

return a \* a \* a \*

a \* a

},

Out: function(a) {

return --a \* a \* a \* a \* a + 1

},

InOut: function(a) {

return 1 > (a\*=2) ? 0.5 \* a \* a \* a \* a \* a : 0.5 \* ((a -= 2) \* a \* a \* a \* a + 2)

}

},

Sinusoidal: {

In: function(a) {

return 1 - Math.cos(a \* Math.PI / 2)

},

Out: function(a) {

return Math.sin(a \* Math.PI / 2)

},

InOut: function(a) {

return 0.5 \* (1 - Math.cos(Math.PI \* a))

}

},

Exponential: {

In: function(a) {

return 0 === a ? 0 : Math.pow(1024, a - 1)

},

Out: function(a) {

return 1 === a ? 1 : 1 - Math.pow(2, - 10 \* a)

},

InOut: function(a) {

return 0 === a ? 0 : 1 === a ? 1 : 1 > (a\*=2) ? 0.5 \* Math.pow(1024, a - 1) : 0.5 \* ( - Math.pow(2, - 10 \* (a - 1)) + 2)

}

},

Circular: {

In: function(a) {

return 1 -

Math.sqrt(1 - a \* a)

},

Out: function(a) {

return Math.sqrt(1 - --a \* a)

},

InOut: function(a) {

return 1 > (a\*=2)?-0.5 \* (Math.sqrt(1 - a \* a) - 1) : 0.5 \* (Math.sqrt(1 - (a -= 2) \* a) + 1)

}

},

Elastic: {

In: function(a) {

var c, b = 0.1;

if (0 === a)

return 0;

if (1 === a)

return 1;

!b || 1 > b ? (b = 1, c = 0.1) : c = 0.4 \* Math.asin(1 / b) / (2 \* Math.PI);

return - (b \* Math.pow(2, 10 \* (a -= 1)) \* Math.sin((a - c) \* 2 \* Math.PI / 0.4))

},

Out: function(a) {

var c, b = 0.1;

if (0 === a)

return 0;

if (1 === a)

return 1;

!b || 1 > b ? (b = 1, c = 0.1) : c = 0.4 \* Math.asin(1 / b) / (2 \* Math.PI);

return b \* Math.pow(2, - 10 \* a) \* Math.sin((a - c) \*

2 \* Math.PI / 0.4) + 1

},

InOut: function(a) {

var c, b = 0.1;

if (0 === a)

return 0;

if (1 === a)

return 1;

!b || 1 > b ? (b = 1, c = 0.1) : c = 0.4 \* Math.asin(1 / b) / (2 \* Math.PI);

return 1 > (a\*=2)?-0.5 \* b \* Math.pow(2, 10 \* (a -= 1)) \* Math.sin((a - c) \* 2 \* Math.PI / 0.4) : 0.5 \* b \* Math.pow(2, - 10 \* (a -= 1)) \* Math.sin((a - c) \* 2 \* Math.PI / 0.4) + 1

}

},

Back: {

In: function(a) {

return a \* a \* (2.70158 \* a - 1.70158)

},

Out: function(a) {

return --a \* a \* (2.70158 \* a + 1.70158) + 1

},

InOut: function(a) {

return 1 > (a\*=2) ? 0.5 \* a \* a \* (3.5949095 \* a - 2.5949095) : 0.5 \* ((a -= 2) \* a \* (3.5949095 \* a + 2.5949095) + 2)

}

},

Bounce: {

In: function(a) {

return 1 -

TWEEN.Easing.Bounce.Out(1 - a)

},

Out: function(a) {

return a < 1 / 2.75 ? 7.5625 \* a \* a : a < 2 / 2.75 ? 7.5625 \* (a -= 1.5 / 2.75) \* a + 0.75 : a < 2.5 / 2.75 ? 7.5625 \* (a -= 2.25 / 2.75) \* a + 0.9375 : 7.5625 \* (a -= 2.625 / 2.75) \* a + 0.984375

},

InOut: function(a) {

return 0.5 > a ? 0.5 \* TWEEN.Easing.Bounce.In(2 \* a) : 0.5 \* TWEEN.Easing.Bounce.Out(2 \* a - 1) + 0.5

}

}

};

TWEEN.Interpolation = {

Linear: function(a, c) {

var b = a.length - 1, d = b \* c, e = Math.floor(d), f = TWEEN.Interpolation.Utils.Linear;

return 0 > c ? f(a[0], a[1], d) : 1 < c ? f(a[b], a[b - 1], b - d) : f(a[e], a[e + 1 > b ? b: e + 1], d - e)

},

Bezier: function(a, c) {

var b = 0, d = a.length - 1, e = Math.pow, f = TWEEN.Interpolation.Utils.Bernstein, h;

for (h = 0; h <= d; h++)

b += e(1 - c, d - h) \* e(c, h) \* a[h] \* f(d, h);

return b

},

CatmullRom: function(a, c) {

var b = a.length - 1, d = b \* c, e = Math.floor(d), f = TWEEN.Interpolation.Utils.CatmullRom;

return a[0] === a[b] ? (0 > c && (e = Math.floor(d = b \* (1 + c))), f(a[(e -

1 + b)%b], a[e], a[(e + 1)%b], a[(e + 2)%b], d - e)) : 0 > c ? a[0] - (f(a[0], a[0], a[1], a[1], - d) - a[0]) : 1 < c ? a[b] - (f(a[b], a[b], a[b - 1], a[b - 1], d - b) - a[b]) : f(a[e ? e - 1: 0], a[e], a[b < e + 1 ? b: e + 1], a[b < e + 2 ? b: e + 2], d - e)

},

Utils: {

Linear: function(a, c, b) {

return (c - a) \* b + a

},

Bernstein: function(a, c) {

var b = TWEEN.Interpolation.Utils.Factorial;

return b(a) / b(c) / b(a - c)

},

Factorial: function() {

var a = [1];

return function(c) {

var b = 1, d;

if (a[c])

return a[c];

for (d = c; 1 < d; d--)

b\*=d;

return a[c] = b

}

}(),

CatmullRom: function(a, c, b, d, e) {

var a = 0.5 \* (b - a), d = 0.5 \* (d - c), f =

e \* e;

return (2 \* c - 2 \* b + a + d) \* e \* f + ( - 3 \* c + 3 \* b - 2 \* a - d) \* f + a \* e + c

}

}

};

/\*\*

\* @author Eberhard Graether / http://egraether.com/

\* @author Mark Lundin / http://mark-lundin.com

\*/

THREE.TrackballControls = function ( object, domElement ) {

var \_this = this;

var STATE = { NONE: -1, ROTATE: 0, ZOOM: 1, PAN: 2, TOUCH\_ROTATE: 3, TOUCH\_ZOOM: 4, TOUCH\_PAN: 5 };

this.object = object;

this.domElement = ( domElement !== undefined ) ? domElement : document;

// API

this.enabled = true;

this.screen = { left: 0, top: 0, width: 0, height: 0 };

this.rotateSpeed = 1.0;

this.zoomSpeed = 1.2;

this.panSpeed = 0.3;

this.noRotate = false;

this.noZoom = false;

this.noPan = false;

this.noRoll = false;

this.staticMoving = false;

this.dynamicDampingFactor = 0.2;

this.minDistance = 0;

this.maxDistance = Infinity;

this.keys = [ 65 /\*A\*/, 83 /\*S\*/, 68 /\*D\*/ ];

// internals

this.target = new THREE.Vector3();

var EPS = 0.000001;

var lastPosition = new THREE.Vector3();

var \_state = STATE.NONE,

\_prevState = STATE.NONE,

\_eye = new THREE.Vector3(),

\_rotateStart = new THREE.Vector3(),

\_rotateEnd = new THREE.Vector3(),

\_zoomStart = new THREE.Vector2(),

\_zoomEnd = new THREE.Vector2(),

\_touchZoomDistanceStart = 0,

\_touchZoomDistanceEnd = 0,

\_panStart = new THREE.Vector2(),

\_panEnd = new THREE.Vector2();

// for reset

this.target0 = this.target.clone();

this.position0 = this.object.position.clone();

this.up0 = this.object.up.clone();

// events

var changeEvent = { type: 'change' };

var startEvent = { type: 'start'};

var endEvent = { type: 'end'};

// methods

this.handleResize = function () {

if ( this.domElement === document ) {

this.screen.left = 0;

this.screen.top = 0;

this.screen.width = window.innerWidth;

this.screen.height = window.innerHeight;

} else {

var box = this.domElement.getBoundingClientRect();

// adjustments come from similar code in the jquery offset() function

var d = this.domElement.ownerDocument.documentElement;

this.screen.left = box.left + window.pageXOffset - d.clientLeft;

this.screen.top = box.top + window.pageYOffset - d.clientTop;

this.screen.width = box.width;

this.screen.height = box.height;

}

};

this.handleEvent = function ( event ) {

if ( typeof this[ event.type ] == 'function' ) {

this[ event.type ]( event );

}

};

this.getMouseOnScreen = function ( pageX, pageY, vector ) {

return vector.set(

( pageX - \_this.screen.left ) / \_this.screen.width,

( pageY - \_this.screen.top ) / \_this.screen.height

);

};

this.getMouseProjectionOnBall = (function(){

var objectUp = new THREE.Vector3(),

mouseOnBall = new THREE.Vector3();

return function ( pageX, pageY, projection ) {

mouseOnBall.set(

( pageX - \_this.screen.width \* 0.5 - \_this.screen.left ) / (\_this.screen.width\*.5),

( \_this.screen.height \* 0.5 + \_this.screen.top - pageY ) / (\_this.screen.height\*.5),

0.0

);

var length = mouseOnBall.length();

if ( \_this.noRoll ) {

if ( length < Math.SQRT1\_2 ) {

mouseOnBall.z = Math.sqrt( 1.0 - length\*length );

} else {

mouseOnBall.z = .5 / length;

}

} else if ( length > 1.0 ) {

mouseOnBall.normalize();

} else {

mouseOnBall.z = Math.sqrt( 1.0 - length \* length );

}

\_eye.copy( \_this.object.position ).sub( \_this.target );

projection.copy( \_this.object.up ).setLength( mouseOnBall.y )

projection.add( objectUp.copy( \_this.object.up ).cross( \_eye ).setLength( mouseOnBall.x ) );

projection.add( \_eye.setLength( mouseOnBall.z ) );

return projection;

}

}());

this.rotateCamera = (function(){

var axis = new THREE.Vector3(),

quaternion = new THREE.Quaternion();

return function () {

var angle = Math.acos( \_rotateStart.dot( \_rotateEnd ) / \_rotateStart.length() / \_rotateEnd.length() );

if ( angle ) {

axis.crossVectors( \_rotateStart, \_rotateEnd ).normalize();

angle \*= \_this.rotateSpeed;

quaternion.setFromAxisAngle( axis, -angle );

\_eye.applyQuaternion( quaternion );

\_this.object.up.applyQuaternion( quaternion );

\_rotateEnd.applyQuaternion( quaternion );

if ( \_this.staticMoving ) {

\_rotateStart.copy( \_rotateEnd );

} else {

quaternion.setFromAxisAngle( axis, angle \* ( \_this.dynamicDampingFactor - 1.0 ) );

\_rotateStart.applyQuaternion( quaternion );

}

}

}

}());

this.zoomCamera = function () {

if ( \_state === STATE.TOUCH\_ZOOM ) {

var factor = \_touchZoomDistanceStart / \_touchZoomDistanceEnd;

\_touchZoomDistanceStart = \_touchZoomDistanceEnd;

\_eye.multiplyScalar( factor );

} else {

var factor = 1.0 + ( \_zoomEnd.y - \_zoomStart.y ) \* \_this.zoomSpeed;

if ( factor !== 1.0 && factor > 0.0 ) {

\_eye.multiplyScalar( factor );

if ( \_this.staticMoving ) {

\_zoomStart.copy( \_zoomEnd );

} else {

\_zoomStart.y += ( \_zoomEnd.y - \_zoomStart.y ) \* this.dynamicDampingFactor;

}

}

}

};

this.panCamera = (function(){

var mouseChange = new THREE.Vector2(),

objectUp = new THREE.Vector3(),

pan = new THREE.Vector3();

return function () {

mouseChange.copy( \_panEnd ).sub( \_panStart );

if ( mouseChange.lengthSq() ) {

mouseChange.multiplyScalar( \_eye.length() \* \_this.panSpeed );

pan.copy( \_eye ).cross( \_this.object.up ).setLength( mouseChange.x );

pan.add( objectUp.copy( \_this.object.up ).setLength( mouseChange.y ) );

\_this.object.position.add( pan );

\_this.target.add( pan );

if ( \_this.staticMoving ) {

\_panStart.copy( \_panEnd );

} else {

\_panStart.add( mouseChange.subVectors( \_panEnd, \_panStart ).multiplyScalar( \_this.dynamicDampingFactor ) );

}

}

}

}());

this.checkDistances = function () {

if ( !\_this.noZoom || !\_this.noPan ) {

if ( \_eye.lengthSq() > \_this.maxDistance \* \_this.maxDistance ) {

\_this.object.position.addVectors( \_this.target, \_eye.setLength( \_this.maxDistance ) );

}

if ( \_eye.lengthSq() < \_this.minDistance \* \_this.minDistance ) {

\_this.object.position.addVectors( \_this.target, \_eye.setLength( \_this.minDistance ) );

}

}

};

this.update = function () {

\_eye.subVectors( \_this.object.position, \_this.target );

if ( !\_this.noRotate ) {

\_this.rotateCamera();

}

if ( !\_this.noZoom ) {

\_this.zoomCamera();

}

if ( !\_this.noPan ) {

\_this.panCamera();

}

\_this.object.position.addVectors( \_this.target, \_eye );

\_this.checkDistances();

\_this.object.lookAt( \_this.target );

if ( lastPosition.distanceToSquared( \_this.object.position ) > EPS ) {

\_this.dispatchEvent( changeEvent );

lastPosition.copy( \_this.object.position );

}

};

this.reset = function () {

\_state = STATE.NONE;

\_prevState = STATE.NONE;

\_this.target.copy( \_this.target0 );

\_this.object.position.copy( \_this.position0 );

\_this.object.up.copy( \_this.up0 );

\_eye.subVectors( \_this.object.position, \_this.target );

\_this.object.lookAt( \_this.target );

\_this.dispatchEvent( changeEvent );

lastPosition.copy( \_this.object.position );

};

// listeners

function keydown( event ) {

if ( \_this.enabled === false ) return;

window.removeEventListener( 'keydown', keydown );

\_prevState = \_state;

if ( \_state !== STATE.NONE ) {

return;

} else if ( event.keyCode === \_this.keys[ STATE.ROTATE ] && !\_this.noRotate ) {

\_state = STATE.ROTATE;

} else if ( event.keyCode === \_this.keys[ STATE.ZOOM ] && !\_this.noZoom ) {

\_state = STATE.ZOOM;

} else if ( event.keyCode === \_this.keys[ STATE.PAN ] && !\_this.noPan ) {

\_state = STATE.PAN;

}

}

function keyup( event ) {

if ( \_this.enabled === false ) return;

\_state = \_prevState;

window.addEventListener( 'keydown', keydown, false );

}

function mousedown( event ) {

if ( \_this.enabled === false ) return;

event.preventDefault();

event.stopPropagation();

if ( \_state === STATE.NONE ) {

\_state = event.button;

}

if ( \_state === STATE.ROTATE && !\_this.noRotate ) {

\_this.getMouseProjectionOnBall( event.pageX, event.pageY, \_rotateStart );

\_rotateEnd.copy(\_rotateStart)

} else if ( \_state === STATE.ZOOM && !\_this.noZoom ) {

\_this.getMouseOnScreen( event.pageX, event.pageY, \_zoomStart );

\_zoomEnd.copy(\_zoomStart);

} else if ( \_state === STATE.PAN && !\_this.noPan ) {

\_this.getMouseOnScreen( event.pageX, event.pageY, \_panStart );

\_panEnd.copy(\_panStart)

}

document.addEventListener( 'mousemove', mousemove, false );

document.addEventListener( 'mouseup', mouseup, false );

\_this.dispatchEvent( startEvent );

}

function mousemove( event ) {

if ( \_this.enabled === false ) return;

event.preventDefault();

event.stopPropagation();

if ( \_state === STATE.ROTATE && !\_this.noRotate ) {

\_this.getMouseProjectionOnBall( event.pageX, event.pageY, \_rotateEnd );

} else if ( \_state === STATE.ZOOM && !\_this.noZoom ) {

\_this.getMouseOnScreen( event.pageX, event.pageY, \_zoomEnd );

} else if ( \_state === STATE.PAN && !\_this.noPan ) {

\_this.getMouseOnScreen( event.pageX, event.pageY, \_panEnd );

}

}

function mouseup( event ) {

if ( \_this.enabled === false ) return;

event.preventDefault();

event.stopPropagation();

\_state = STATE.NONE;

document.removeEventListener( 'mousemove', mousemove );

document.removeEventListener( 'mouseup', mouseup );

\_this.dispatchEvent( endEvent );

}

function mousewheel( event ) {

if ( \_this.enabled === false ) return;

event.preventDefault();

event.stopPropagation();

var delta = 0;

if ( event.wheelDelta ) { // WebKit / Opera / Explorer 9

delta = event.wheelDelta / 40;

} else if ( event.detail ) { // Firefox

delta = - event.detail / 3;

}

\_zoomStart.y += delta \* 0.01;

\_this.dispatchEvent( startEvent );

\_this.dispatchEvent( endEvent );

}

function touchstart( event ) {

if ( \_this.enabled === false ) return;

switch ( event.touches.length ) {

case 1:

\_state = STATE.TOUCH\_ROTATE;

\_rotateEnd.copy( \_this.getMouseProjectionOnBall( event.touches[ 0 ].pageX, event.touches[ 0 ].pageY, \_rotateStart ));

break;

case 2:

\_state = STATE.TOUCH\_ZOOM;

var dx = event.touches[ 0 ].pageX - event.touches[ 1 ].pageX;

var dy = event.touches[ 0 ].pageY - event.touches[ 1 ].pageY;

\_touchZoomDistanceEnd = \_touchZoomDistanceStart = Math.sqrt( dx \* dx + dy \* dy );

break;

case 3:

\_state = STATE.TOUCH\_PAN;

\_panEnd.copy( \_this.getMouseOnScreen( event.touches[ 0 ].pageX, event.touches[ 0 ].pageY, \_panStart ));

break;

default:

\_state = STATE.NONE;

}

\_this.dispatchEvent( startEvent );

}

function touchmove( event ) {

if ( \_this.enabled === false ) return;

event.preventDefault();

event.stopPropagation();

switch ( event.touches.length ) {

case 1:

\_this.getMouseProjectionOnBall( event.touches[ 0 ].pageX, event.touches[ 0 ].pageY, \_rotateEnd );

break;

case 2:

var dx = event.touches[ 0 ].pageX - event.touches[ 1 ].pageX;

var dy = event.touches[ 0 ].pageY - event.touches[ 1 ].pageY;

\_touchZoomDistanceEnd = Math.sqrt( dx \* dx + dy \* dy )

break;

case 3:

\_this.getMouseOnScreen( event.touches[ 0 ].pageX, event.touches[ 0 ].pageY, \_panEnd );

break;

default:

\_state = STATE.NONE;

}

}

function touchend( event ) {

if ( \_this.enabled === false ) return;

switch ( event.touches.length ) {

case 1:

\_rotateStart.copy( \_this.getMouseProjectionOnBall( event.touches[ 0 ].pageX, event.touches[ 0 ].pageY, \_rotateEnd ));

break;

case 2:

\_touchZoomDistanceStart = \_touchZoomDistanceEnd = 0;

break;

case 3:

\_panStart.copy( \_this.getMouseOnScreen( event.touches[ 0 ].pageX, event.touches[ 0 ].pageY, \_panEnd ));

break;

}

\_state = STATE.NONE;

\_this.dispatchEvent( endEvent );

}

this.domElement.addEventListener( 'contextmenu', function ( event ) { event.preventDefault(); }, false );

this.domElement.addEventListener( 'mousedown', mousedown, false );

this.domElement.addEventListener( 'mousewheel', mousewheel, false );

this.domElement.addEventListener( 'DOMMouseScroll', mousewheel, false ); // firefox

this.domElement.addEventListener( 'touchstart', touchstart, false );

this.domElement.addEventListener( 'touchend', touchend, false );

this.domElement.addEventListener( 'touchmove', touchmove, false );

window.addEventListener( 'keydown', keydown, false );

window.addEventListener( 'keyup', keyup, false );

this.handleResize();

// force an update at start

this.update();

};

THREE.TrackballControls.prototype = Object.create( THREE.EventDispatcher.prototype );

/\*\*

\* Based on http://www.emagix.net/academic/mscs-project/item/camera-sync-with-css3-and-webgl-threejs

\* @author mrdoob / http://mrdoob.com/

\*/

THREE.CSS3DObject = function ( element ) {

THREE.Object3D.call( this );

this.element = element;

this.element.style.position = 'absolute';

this.addEventListener( 'removed', function ( event ) {

if ( this.element.parentNode !== null ) {

this.element.parentNode.removeChild( this.element );

for ( var i = 0, l = this.children.length; i < l; i ++ ) {

this.children[ i ].dispatchEvent( event );

}

}

} );

};

THREE.CSS3DObject.prototype = Object.create( THREE.Object3D.prototype );

THREE.CSS3DSprite = function ( element ) {

THREE.CSS3DObject.call( this, element );

};

THREE.CSS3DSprite.prototype = Object.create( THREE.CSS3DObject.prototype );

//

THREE.CSS3DRenderer = function () {

console.log( 'THREE.CSS3DRenderer', THREE.REVISION );

var \_width, \_height;

var \_widthHalf, \_heightHalf;

var matrix = new THREE.Matrix4();

var domElement = document.createElement( 'div' );

domElement.style.overflow = 'hidden';

domElement.style.WebkitTransformStyle = 'preserve-3d';

domElement.style.MozTransformStyle = 'preserve-3d';

domElement.style.oTransformStyle = 'preserve-3d';

domElement.style.transformStyle = 'preserve-3d';

this.domElement = domElement;

var cameraElement = document.createElement( 'div' );

cameraElement.style.WebkitTransformStyle = 'preserve-3d';

cameraElement.style.MozTransformStyle = 'preserve-3d';

cameraElement.style.oTransformStyle = 'preserve-3d';

cameraElement.style.transformStyle = 'preserve-3d';

domElement.appendChild( cameraElement );

this.setClearColor = function () {

};

this.setSize = function ( width, height ) {

\_width = width;

\_height = height;

\_widthHalf = \_width / 2;

\_heightHalf = \_height / 2;

domElement.style.width = width + 'px';

domElement.style.height = height + 'px';

cameraElement.style.width = width + 'px';

cameraElement.style.height = height + 'px';

};

var epsilon = function ( value ) {

return Math.abs( value ) < 0.000001 ? 0 : value;

};

var getCameraCSSMatrix = function ( matrix ) {

var elements = matrix.elements;

return 'matrix3d(' +

epsilon( elements[ 0 ] ) + ',' +

epsilon( - elements[ 1 ] ) + ',' +

epsilon( elements[ 2 ] ) + ',' +

epsilon( elements[ 3 ] ) + ',' +

epsilon( elements[ 4 ] ) + ',' +

epsilon( - elements[ 5 ] ) + ',' +

epsilon( elements[ 6 ] ) + ',' +

epsilon( elements[ 7 ] ) + ',' +

epsilon( elements[ 8 ] ) + ',' +

epsilon( - elements[ 9 ] ) + ',' +

epsilon( elements[ 10 ] ) + ',' +

epsilon( elements[ 11 ] ) + ',' +

epsilon( elements[ 12 ] ) + ',' +

epsilon( - elements[ 13 ] ) + ',' +

epsilon( elements[ 14 ] ) + ',' +

epsilon( elements[ 15 ] ) +

')';

};

var getObjectCSSMatrix = function ( matrix ) {

var elements = matrix.elements;

return 'translate3d(-50%,-50%,0) matrix3d(' +

epsilon( elements[ 0 ] ) + ',' +

epsilon( elements[ 1 ] ) + ',' +

epsilon( elements[ 2 ] ) + ',' +

epsilon( elements[ 3 ] ) + ',' +

epsilon( - elements[ 4 ] ) + ',' +

epsilon( - elements[ 5 ] ) + ',' +

epsilon( - elements[ 6 ] ) + ',' +

epsilon( - elements[ 7 ] ) + ',' +

epsilon( elements[ 8 ] ) + ',' +

epsilon( elements[ 9 ] ) + ',' +

epsilon( elements[ 10 ] ) + ',' +

epsilon( elements[ 11 ] ) + ',' +

epsilon( elements[ 12 ] ) + ',' +

epsilon( elements[ 13 ] ) + ',' +

epsilon( elements[ 14 ] ) + ',' +

epsilon( elements[ 15 ] ) +

')';

};

var renderObject = function ( object, camera ) {

if ( object instanceof THREE.CSS3DObject ) {

var style;

if ( object instanceof THREE.CSS3DSprite ) {

// http://swiftcoder.wordpress.com/2008/11/25/constructing-a-billboard-matrix/

matrix.copy( camera.matrixWorldInverse );

matrix.transpose();

matrix.copyPosition( object.matrixWorld );

matrix.scale( object.scale );

matrix.elements[ 3 ] = 0;

matrix.elements[ 7 ] = 0;

matrix.elements[ 11 ] = 0;

matrix.elements[ 15 ] = 1;

style = getObjectCSSMatrix( matrix );

} else {

style = getObjectCSSMatrix( object.matrixWorld );

}

var element = object.element;

element.style.WebkitTransform = style;

element.style.MozTransform = style;

element.style.oTransform = style;

element.style.transform = style;

if ( element.parentNode !== cameraElement ) {

cameraElement.appendChild( element );

}

}

for ( var i = 0, l = object.children.length; i < l; i ++ ) {

renderObject( object.children[ i ], camera );

}

};

this.render = function ( scene, camera ) {

var fov = 0.5 / Math.tan( THREE.Math.degToRad( camera.fov \* 0.5 ) ) \* \_height;

domElement.style.WebkitPerspective = fov + "px";

domElement.style.MozPerspective = fov + "px";

domElement.style.oPerspective = fov + "px";

domElement.style.perspective = fov + "px";

scene.updateMatrixWorld();

if ( camera.parent === undefined ) camera.updateMatrixWorld();

camera.matrixWorldInverse.getInverse( camera.matrixWorld );

var style = "translate3d(0,0," + fov + "px)" + getCameraCSSMatrix( camera.matrixWorldInverse ) +

" translate3d(" + \_widthHalf + "px," + \_heightHalf + "px, 0)";

cameraElement.style.WebkitTransform = style;

cameraElement.style.MozTransform = style;

cameraElement.style.oTransform = style;

cameraElement.style.transform = style;

renderObject( scene, camera );

};

};

var table = [

"H", "Hydrogen", "1.00794", 1, 1,

"He", "Helium", "4.002602", 18, 1,

"Li", "Lithium", "#6.941", 1, 2,

"Be", "Beryllium", "9.012182", 2, 2,

"B", "Boron", "#10.811", 13, 2,

"C", "Carbon", "#12.0107", 14, 2,

"N", "Nitrogen", "#14.0067", 15, 2,

"O", "Oxygen", "#15.9994", 16, 2,

"F", "Fluorine", "18.9984032", 17, 2,

"Ne", "Neon", "#20.1797", 18, 2,

"Na", "Sodium", "22.98976...", 1, 3,

"Mg", "Magnesium", "#24.305", 2, 3,

"Al", "Aluminium", "26.9815386", 13, 3,

"Si", "Silicon", "#28.0855", 14, 3,

"P", "Phosphorus", "30.973762", 15, 3,

"S", "Sulfur", "#32.065", 16, 3,

"Cl", "Chlorine", "#35.453", 17, 3,

"Ar", "Argon", "#39.948", 18, 3,

"K", "Potassium", "#39.948", 1, 4,

"Ca", "Calcium", "#40.078", 2, 4,

"Sc", "Scandium", "44.955912", 3, 4,

"Ti", "Titanium", "#47.867", 4, 4,

"V", "Vanadium", "#50.9415", 5, 4,

"Cr", "Chromium", "#51.9961", 6, 4,

"Mn", "Manganese", "54.938045", 7, 4,

"Fe", "Iron", "#55.845", 8, 4,

"Co", "Cobalt", "58.933195", 9, 4,

"Ni", "Nickel", "#58.6934", 10, 4,

"Cu", "Copper", "#63.546", 11, 4,

"Zn", "Zinc", "#65.38", 12, 4,

"Ga", "Gallium", "#69.723", 13, 4,

"Ge", "Germanium", "#72.63", 14, 4,

"As", "Arsenic", "#74.9216", 15, 4,

"Se", "Selenium", "#78.96", 16, 4,

"Br", "Bromine", "#79.904", 17, 4,

"Kr", "Krypton", "#83.798", 18, 4,

"Rb", "Rubidium", "#85.4678", 1, 5,

"Sr", "Strontium", "#87.62", 2, 5,

"Y", "Yttrium", "88.90585", 3, 5,

"Zr", "Zirconium", "#91.224", 4, 5,

"Nb", "Niobium", "92.90628", 5, 5,

"Mo", "Molybdenum", "#95.96", 6, 5,

"Tc", "Technetium", "(98)", 7, 5,

"Ru", "Ruthenium", "#101.07", 8, 5,

"Rh", "Rhodium", "#102.9055", 9, 5,

"Pd", "Palladium", "#106.42", 10, 5,

"Ag", "Silver", "#107.8682", 11, 5,

"Cd", "Cadmium", "#112.411", 12, 5,

"In", "Indium", "#114.818", 13, 5,

"Sn", "Tin", "#118.71", 14, 5,

"Sb", "Antimony", "#121.76", 15, 5,

"Te", "Tellurium", "127.6", 16, 5,

"I", "Iodine", "126.90447", 17, 5,

"Xe", "Xenon", "#131.293", 18, 5,

"Cs", "Caesium", "#132.9054", 1, 6,

"Ba", "Barium", "#132.9054", 2, 6,

"La", "Lanthanum", "138.90547", 4, 9,

"Ce", "Cerium", "#140.116", 5, 9,

"Pr", "Praseodymium", "140.90765", 6, 9,

"Nd", "Neodymium", "#144.242", 7, 9,

"Pm", "Promethium", "(145)", 8, 9,

"Sm", "Samarium", "#150.36", 9, 9,

"Eu", "Europium", "#151.964", 10, 9,

"Gd", "Gadolinium", "#157.25", 11, 9,

"Tb", "Terbium", "158.92535", 12, 9,

"Dy", "Dysprosium", "162.5", 13, 9,

"Ho", "Holmium", "164.93032", 14, 9,

"Er", "Erbium", "#167.259", 15, 9,

"Tm", "Thulium", "168.93421", 16, 9,

"Yb", "Ytterbium", "#173.054", 17, 9,

"Lu", "Lutetium", "#174.9668", 18, 9,

"Hf", "Hafnium", "#178.49", 4, 6,

"Ta", "Tantalum", "180.94788", 5, 6,

"W", "Tungsten", "#183.84", 6, 6,

"Re", "Rhenium", "#186.207", 7, 6,

"Os", "Osmium", "#190.23", 8, 6,

"Ir", "Iridium", "#192.217", 9, 6,

"Pt", "Platinum", "#195.084", 10, 6,

"Au", "Gold", "196.966569", 11, 6,

"Hg", "Mercury", "#200.59", 12, 6,

"Tl", "Thallium", "#204.3833", 13, 6,

"Pb", "Lead", "207.2", 14, 6,

"Bi", "Bismuth", "#208.9804", 15, 6,

"Po", "Polonium", "(209)", 16, 6,

"At", "Astatine", "(210)", 17, 6,

"Rn", "Radon", "(222)", 18, 6,

"Fr", "Francium", "(223)", 1, 7,

"Ra", "Radium", "(226)", 2, 7,

"Ac", "Actinium", "(227)", 4, 10,

"Th", "Thorium", "232.03806", 5, 10,

"Pa", "Protactinium", "#231.0588", 6, 10,

"U", "Uranium", "238.02891", 7, 10,

"Np", "Neptunium", "(237)", 8, 10,

"Pu", "Plutonium", "(244)", 9, 10,

"Am", "Americium", "(243)", 10, 10,

"Cm", "Curium", "(247)", 11, 10,

"Bk", "Berkelium", "(247)", 12, 10,

"Cf", "Californium", "(251)", 13, 10,

"Es", "Einstenium", "(252)", 14, 10,

"Fm", "Fermium", "(257)", 15, 10,

"Md", "Mendelevium", "(258)", 16, 10,

"No", "Nobelium", "(259)", 17, 10,

"Lr", "Lawrencium", "(262)", 18, 10,

"Rf", "Rutherfordium", "(267)", 4, 7,

"Db", "Dubnium", "(268)", 5, 7,

"Sg", "Seaborgium", "(271)", 6, 7,

"Bh", "Bohrium", "(272)", 7, 7,

"Hs", "Hassium", "(270)", 8, 7,

"Mt", "Meitnerium", "(276)", 9, 7,

"Ds", "Darmstadium", "(281)", 10, 7,

"Rg", "Roentgenium", "(280)", 11, 7,

"Cn", "Copernicium", "(285)", 12, 7,

"Uut", "Unutrium", "(284)", 13, 7,

"Fl", "Flerovium", "(289)", 14, 7,

"Uup", "Ununpentium", "(288)", 15, 7,

"Lv", "Livermorium", "(293)", 16, 7,

"Uus", "Ununseptium", "(294)", 17, 7,

"Uuo", "Ununoctium", "(294)", 18, 7

];

var camera, scene, renderer;

var controls;

var objects = [];

var targets = { table: [], sphere: [], helix: [], grid: [] };

init();

animate();

function init() {

camera = new THREE.PerspectiveCamera( 40, window.innerWidth / window.innerHeight, 1, 10000 );

camera.position.z = 3000;

scene = new THREE.Scene();

// table

for ( var i = 0; i < table.length; i += 5 ) {

var element = document.createElement( 'div' );

element.className = 'element';

element.style.backgroundColor = 'rgba(0,127,127,' + ( Math.random() \* 0.5 + 0.25 ) + ')';

var number = document.createElement( 'div' );

number.className = 'number';

number.textContent = (i/5) + 1;

element.appendChild( number );

var symbol = document.createElement( 'div' );

symbol.className = 'symbol';

symbol.textContent = table[ i ];

element.appendChild( symbol );

var details = document.createElement( 'div' );

details.className = 'details';

details.innerHTML = table[ i + 1 ] + '<br>' + table[ i + 2 ];

element.appendChild( details );

var object = new THREE.CSS3DObject( element );

object.position.x = Math.random() \* 4000 - 2000;

object.position.y = Math.random() \* 4000 - 2000;

object.position.z = Math.random() \* 4000 - 2000;

scene.add( object );

objects.push( object );

//

var object = new THREE.Object3D();

object.position.x = ( table[ i + 3 ] \* 140 ) - 1330;

object.position.y = - ( table[ i + 4 ] \* 180 ) + 990;

targets.table.push( object );

}

// sphere

var vector = new THREE.Vector3();

for ( var i = 0, l = objects.length; i < l; i ++ ) {

var phi = Math.acos( -1 + ( 2 \* i ) / l );

var theta = Math.sqrt( l \* Math.PI ) \* phi;

var object = new THREE.Object3D();

object.position.x = 800 \* Math.cos( theta ) \* Math.sin( phi );

object.position.y = 800 \* Math.sin( theta ) \* Math.sin( phi );

object.position.z = 800 \* Math.cos( phi );

vector.copy( object.position ).multiplyScalar( 2 );

object.lookAt( vector );

targets.sphere.push( object );

}

// helix

var vector = new THREE.Vector3();

for ( var i = 0, l = objects.length; i < l; i ++ ) {

var phi = i \* 0.175 + Math.PI;

var object = new THREE.Object3D();

object.position.x = 900 \* Math.sin( phi );

object.position.y = - ( i \* 8 ) + 450;

object.position.z = 900 \* Math.cos( phi );

vector.x = object.position.x \* 2;

vector.y = object.position.y;

vector.z = object.position.z \* 2;

object.lookAt( vector );

targets.helix.push( object );

}

// grid

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

var object = new THREE.Object3D();

object.position.x = ( ( i % 5 ) \* 400 ) - 800;

object.position.y = ( - ( Math.floor( i / 5 ) % 5 ) \* 400 ) + 800;

object.position.z = ( Math.floor( i / 25 ) ) \* 1000 - 2000;

targets.grid.push( object );

}

//

renderer = new THREE.CSS3DRenderer();

renderer.setSize( window.innerWidth, window.innerHeight );

renderer.domElement.style.position = 'absolute';

document.getElementById( 'container' ).appendChild( renderer.domElement );

//

controls = new THREE.TrackballControls( camera, renderer.domElement );

controls.rotateSpeed = 0.5;

controls.minDistance = 500;

controls.maxDistance = 6000;

controls.addEventListener( 'change', render );

var button = document.getElementById( 'table' );

button.addEventListener( 'click', function ( event ) {

transform( targets.table, 2000 );

}, false );

var button = document.getElementById( 'sphere' );

button.addEventListener( 'click', function ( event ) {

transform( targets.sphere, 2000 );

}, false );

var button = document.getElementById( 'helix' );

button.addEventListener( 'click', function ( event ) {

transform( targets.helix, 2000 );

}, false );

var button = document.getElementById( 'grid' );

button.addEventListener( 'click', function ( event ) {

transform( targets.grid, 2000 );

}, false );

transform( targets.table, 5000 );

//

window.addEventListener( 'resize', onWindowResize, false );

}

function transform( targets, duration ) {

TWEEN.removeAll();

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

var object = objects[ i ];

var target = targets[ i ];

new TWEEN.Tween( object.position )

.to( { x: target.position.x, y: target.position.y, z: target.position.z }, Math.random() \* duration + duration )

.easing( TWEEN.Easing.Exponential.InOut )

.start();

new TWEEN.Tween( object.rotation )

.to( { x: target.rotation.x, y: target.rotation.y, z: target.rotation.z }, Math.random() \* duration + duration )

.easing( TWEEN.Easing.Exponential.InOut )

.start();

}

new TWEEN.Tween( this )

.to( {}, duration \* 2 )

.onUpdate( render )

.start();

}

function onWindowResize() {

camera.aspect = window.innerWidth / window.innerHeight;

camera.updateProjectionMatrix();

renderer.setSize( window.innerWidth, window.innerHeight );

render();

}

function animate() {

requestAnimationFrame( animate );

TWEEN.update();

controls.update();

}

function render() {

renderer.render( scene, camera );

}

Bảng tuần hoàn cổ điển :

**Bài 6: Xu hướng biến đổi theo chu kì và nhóm:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi

**Bài 7: Xu hướng biến đổi thành phần & tính chất hợp chất:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi

**Bài 8: Định luật tuần hoàn, ý nghĩa bảng tuần hoàn:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi

**Bài 9: Ôn tập chương 2:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi

# Chương 3: Liên kết hóa học

**Bài 10: Quy tắc Octet:** https://phet.colorado.edu/sims/html/ionic-covalent-bonds/latest/ionic-covalent-bonds\_all.html?locale=vi

**Bài 11: Liên kết ion:** https://phet.colorado.edu/sims/html/ionic-covalent-bonds/latest/ionic-covalent-bonds\_all.html?locale=vi

**Bài 12: Liên kết cộng hoá trị:** https://phet.colorado.edu/sims/html/ionic-covalent-bonds/latest/ionic-covalent-bonds\_all.html?locale=vi

**Bài 13: Liên kết Hydrogen và Van der Waals:** https://phet.colorado.edu/sims/html/atomic-interactions/latest/atomic-interactions\_all.html?locale=vi

**Bài 14: Ôn tập chương 3:** https://phet.colorado.edu/sims/html/ionic-covalent-bonds/latest/ionic-covalent-bonds\_all.html?locale=vi

# Chương 4: Phản ứng oxi hóa - khử

**Bài 15: Phản ứng oxi hóa – khử:** https://phet.colorado.edu/sims/html/reactions-and-rates/latest/reactions-and-rates\_all.html?locale=vi

**Bài 16: Ôn tập chương 4:** https://phet.colorado.edu/sims/html/reactions-and-rates/latest/reactions-and-rates\_all.html?locale=vi

# Chương 5: Năng lượng hóa học

**Bài 17: Biến thiên Enthalpy:** https://phet.colorado.edu/sims/html/energy-forms-and-changes/latest/energy-forms-and-changes\_all.html?locale=vi

**Bài 18: Ôn tập chương 5:** https://phet.colorado.edu/sims/html/energy-forms-and-changes/latest/energy-forms-and-changes\_all.html?locale=vi

# Chương 6: Tốc độ phản ứng

**Bài 19: Tốc độ phản ứng:** https://phet.colorado.edu/sims/html/reactions-and-rates/latest/reactions-and-rates\_all.html?locale=vi

**Bài 20: Ôn tập chương 6:** https://phet.colorado.edu/sims/html/reactions-and-rates/latest/reactions-and-rates\_all.html?locale=vi

# Chương 7: Nguyên tố nhóm Halogen

**Bài 21: Nhóm Halogen:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi

**Bài 22: Hydrogen Halide, Muối Halide:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi

**Bài 23: Ôn tập chương 7:** https://phet.colorado.edu/sims/html/periodic-trends/latest/periodic-trends\_all.html?locale=vi