<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8">

    <meta name="authoring-tool" content="Adobe\_Animate\_CC">

    <title>SPACE\_WAR</title>

    <script src="https://code.createjs.com/1.0.0/createjs.min.js"></script>

    <script src="SPACE\_WAR.js?1592651286108"></script>

    <!-- 引入css -->

    <link rel="stylesheet" href="./SPACE\_WAR.css">

    <!-- 引入碰撞函數 -->

    <script src="./ndgmr.Collision.js"></script>

    <script>

        var canvas, stage, exportRoot, anim\_container, dom\_overlay\_container, fnStartAnimation;

        function init() {

            canvas = document.getElementById("canvas");

            anim\_container = document.getElementById("animation\_container");

            dom\_overlay\_container = document.getElementById("dom\_overlay\_container");

            var comp = AdobeAn.getComposition("0356A0DF8E8F4448B3C6AC8B88E2FDE9");

            var lib = comp.getLibrary();

            var loader = new createjs.LoadQueue(false);

            loader.addEventListener("fileload", function (evt) {

                handleFileLoad(evt, comp)

            });

            loader.addEventListener("complete", function (evt) {

                handleComplete(evt, comp)

            });

            var lib = comp.getLibrary();

            loader.loadManifest(lib.properties.manifest);

        }

        function handleFileLoad(evt, comp) {

            var images = comp.getImages();

            if (evt && (evt.item.type == "image")) {

                images[evt.item.id] = evt.result;

            }

        }

        function handleComplete(evt, comp) {

            var lib = comp.getLibrary();

            var ss = comp.getSpriteSheet();

            var queue = evt.target;

            var ssMetadata = lib.ssMetadata;

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

                ss[ssMetadata[i].name] = new createjs.SpriteSheet({

                    "images": [queue.getResult(ssMetadata[i].name)],

                    "frames": ssMetadata[i].frames

                })

            }

            var set\_volume = 0.6;

            //判定是否暫停

            var is\_pause = false;

            //判定是否勝利

            var is\_win = false;

            //判定是否開始

            var is\_start = false;

            //判定是否失敗

            var is\_gameover = false;

            //判定boss是否出現

            var boss\_show = false;

            //建立背景

            exportRoot = new lib.SPACE\_WAR();

            stage = new lib.Stage(canvas);

            //螢幕上顯示背景

            stage.addChild(exportRoot);

            //建立飛船

            var ship = new lib.sh();

            //設定飛船初始位置

            ship.x = 512;

            ship.y = 660;

            //螢幕上顯示飛船

            exportRoot.addChild(ship);

            //預設控制水平與垂直移動變數為0

            var H = 0;

            var V = 0;

            //設定每次移動的距離

            var D = 15;

            //判定鍵盤是否按下

            var is\_KD = false;

            //預設分數變數為0分

            var score = 0;

            //預設血量100

            var Hp = 100;

            //預設藍量100

            var Mp = 100;

            //for a1

            //預設a1子彈數為0

            var a1\_num = 0;

            //裝a1子彈的陣列

            var a1\_array = [0];

            //產生a1子彈的function

            var a1\_generate = function () {

                //建立a1

                let a1 = new lib.a1();

                //回傳建立的物件a1

                return a1;

            }

            //for a2

            //預設a2雷球數為0

            var a2\_num = 0;

            //裝a2雷球的陣列

            var a2\_array = [0];

            //產生a2雷球的function

            var a2\_generate = function () {

                let a2 = new lib.a2();

                return a2;

            }

            //for a4

            //預設a4脈衝波數為0

            var a4\_num = 0;

            //裝a4脈衝波的陣列

            var a4\_array = [0];

            //產生a4脈衝波的function

            var a4\_generate = function () {

                let a4 = new lib.a4();

                return a4;

            }

            //for a7

            //預設a7月牙天衝數為0

            var a7\_num = 0;

            //裝a7月牙天衝的陣列

            var a7\_array = [0];

            //產生a7月牙天衝的function

            var a7\_generate = function () {

                let a7 = new lib.a7();

                return a7;

            }

            //抓取使用音效的位置同時給定id方便之後調用

            const sounds = [{

                    src: "./音效/game\_start.mp3",

                    id: "game\_start"

                },

                {

                    src: "./音效/attack1.mp3",

                    id: "attack1"

                },

                {

                    src: "./音效/boom.mp3",

                    id: "boom"

                },

                {

                    src: "./音效/hit1.mp3",

                    id: "hit1"

                },

                {

                    src: "./音效/doong.mp3",

                    id: "doong"

                },

                {

                    src: "./音效/lazer.mp3",

                    id: "lazer"

                },

                {

                    src: "./音效/ping.mp3",

                    id: "ping"

                },

                {

                    src: "./音效/devil\_scared1.mp3",

                    id: "devil\_scared1"

                },

                {

                    src: "./音效/devil\_scared2.mp3",

                    id: "devil\_scared2"

                },

                {

                    src: "./音效/harmed.mp3",

                    id: "harmed"

                },

                {

                    src: "./音效/lightning.mp3",

                    id: "lightning"

                },

                {

                    src: "./音效/attack7.mp3",

                    id: "attack7"

                },

                {

                    src: "./音效/striking.mp3",

                    id: "striking"

                },

                {

                    src: "./音效/impact.mp3",

                    id: "impact"

                },

                {

                    src: "./音效/game\_over.mp3",

                    id: "game\_over"

                },

                {

                    src: "./音效/hp\_up.mp3",

                    id: "hp\_up"

                },

                {

                    src: "./音效/win.mp3",

                    id: "win"

                },

                {

                    src: "./音效/boss.mp3",

                    id: "boss"

                },

            ];

            createjs.Sound.alternateExtensions = ["mp3"];

            createjs.Sound.registerSounds(sounds);

            createjs.Sound.addEventListener("fileload", (e) => {

                console.log(e.src);

                console.log(e.id);

            });

            //啟動(遊戲開始)鍵盤事件監聽

            window.addEventListener("keydown", start);

            //遊戲開始function

            function start(e) {

                //按空白鍵開始遊戲

                if (e.keyCode === 32) {

                    //遊戲開始為真

                    is\_start = true;

                    //播放遊戲開始音效

                    createjs.Sound.play("game\_start").volume = set\_volume;

                    //將開始按鈕改為jquery版本

                    $('#start').css("display", "none");

                    //物件導向版本如下

                    // document.querySelector("#start").style.display = "none";

                    //啟動鍵盤事件監聽

                    window.addEventListener("keydown", KDF);

                    window.addEventListener("keyup", KUF);

                    //啟動定時更新遊戲事件變化

                    createjs.Ticker.addEventListener("tick", change);

                    //關掉(遊戲開始)鍵盤事件監聽

                    window.removeEventListener("keydown", start);

                    //刪除遊戲說明

                    $("#intro").remove();

                    //暫停

                    $('#pause').click(() => {

                        if (!is\_pause) {

                            createjs.Sound.stop();

                            createjs.Ticker.paused = true;

                            is\_pause = true;

                        } else {

                            createjs.Ticker.paused = false;

                            is\_pause = false;

                        }

                    })

                }

            }

            //遊戲再玩一次按鈕

            //改成jquery版本

            $('#restart').click(() => {

                window.location.reload();

            })

            //物件導向版本如下

            // document.querySelector("#restart").addEventListener("click", () => {

            //  window.location.reload()

            // })

            //音量控制

            $('#volume').click(() => {

                if (set\_volume == 0.6) {

                    $('#V').attr('src', "./img/S1.png");

                    set\_volume = 1;

                } else if (set\_volume == 1) {

                    $('#V').attr('src', "./img/S0.png");

                    set\_volume = 0;

                } else if (set\_volume == 0) {

                    $('#V').attr('src', "./img/S0.3.png");

                    set\_volume = 0.3;

                } else if (set\_volume == 0.3) {

                    $('#V').attr('src', "./img/S0.6.png");

                    set\_volume = 0.6;

                }

            })

            //重新

            $('#restart1').click(() => {

                window.location.reload();

            })

            // 怪物1

            //設定間隔時間執行生成monster1

            let monster1 = setInterval(() => {

                    //當分數條件滿足時執行

                    if (score >= 0 && score <= 50) {

                        //當遊戲開始為真且非暫停時執行

                        if (is\_start && !is\_pause) {

                            //建立怪物1(物件導向)

                            let m = new lib.m1();

                            //設定初始位置

                            //(最大值-最小值+1)\*亂數 +最小值

                            //x隨機(random放大後取高斯)

                            m.x = Math.floor(Math.random() \* (924 - 100 + 1) + 100);

                            //y為-50(屏幕後)

                            m.y = -50;

                            //怪物1放大1.5倍

                            m.scaleX = 1.5;

                            m.scaleY = 1.5;

                            //螢幕上顯示怪物1

                            exportRoot.addChild(m);

                            //設定怪物1初始血量為1滴血(區域變數)

                            let HP = 1;

                            //設定區域定數is\_died判定怪物1是否死亡(決定此物件的生命週期)

                            let is\_died = false;

                            //用HTML5canvas動畫工具createjs輔助

                            //createjs.Tween.get(目標)控制怪物1的移動、表現、生命週期、觸發事件...

                            createjs.Tween.get(m)

                                //設定怪物1由起點移動到指定位置(同時給定要達成秒數(毫秒))

                                .to({

                                    y: 768,

                                }, 15000)

                                //設定當怪物1到達指定位置後要執行的指令

                                .call(() => {

                                    is\_died = true;

                                    //將怪物1剩餘血量扣在玩家當前血量身上

                                    Hp -= HP;

                                    //消除怪物1(物件與動畫)

                                    createjs.Tween.removeTweens(m);

                                    exportRoot.removeChild(m);

                                })

                                //建立怪物1觸發事件監聽(當每次怪物位置變更時自動執行)

                                .addEventListener("change", () => {

                                    //從a1\_array抓取每個創立且儲存在陣列內的a1子彈

                                    for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                        //當陣列的i位置為空(0)時跳過這次，並繼續往下執行i+1

                                        if (a1\_array[i] == 0) {

                                            continue;

                                        } else {

                                            //利用引入的第三方函式庫判定物件之間的碰撞

                                            //判定a1子彈是否有與怪物1碰撞，回傳true或false(由區域變數hit\_target接收)

                                            let hit\_target = ndgmr.checkRectCollision(a1\_array[i], m);

                                            //當a1子彈與怪物1碰撞時執行

                                            if (hit\_target) {

                                                //怪物1血量-1

                                                HP -= 1;

                                                //當怪物1血量小於0時執行

                                                if (HP <= 0) {

                                                    //消除怪物1(物件與動畫)的del function

                                                    //等等在setTimeout時調用，目的是為了讓爆炸動畫播放完才執行消除(馬上執行消除怪物1會看不到爆炸動畫)

                                                    function del() {

                                                        createjs.Tween.removeTweens(m);

                                                        exportRoot.removeChild(m);

                                                    }

                                                    //當is\_died為真時break(目的是因為setTimeout所以怪物1會延後消除(但怪物早已死)，此期間避免第二顆子彈與怪物1做重複碰撞判定造成分數累加等...)

                                                    if (is\_died) {

                                                        break;

                                                    } else {

                                                        //遊戲分數+1

                                                        score += 1;

                                                        //怪物1死亡為真(is\_died判定在這邊才轉為真，之後的碰撞會被視為無效碰撞)

                                                        is\_died = true;

                                                        //播放怪物爆炸音效

                                                        createjs.Sound.play("boom").volume = set\_volume;

                                                        //播放怪物爆炸動畫

                                                        m.gotoAndPlay("boom");

                                                        //利用setTimeout延後執行del function

                                                        setTimeout(del, 100);

                                                    }

                                                }

                                                //消除a1子彈(物件與動畫)

                                                createjs.Tween.removeTweens(a1\_array[i]);

                                                exportRoot.removeChild(a1\_array[i]);

                                                //將a1子彈陣列的第i個位置設為0(也就是與怪物1碰撞的那顆子彈)

                                                a1\_array[i] = 0;

                                            }

                                        }

                                    }

                                    //脈衝波碰撞判定

                                    for (let i = 0; i <= (a4\_array.length - 1); i++) {

                                        if (a4\_array[i] == 0) {

                                            continue;

                                        } else {

                                            let hit\_target = ndgmr.checkRectCollision(a4\_array[i], m);

                                            if (hit\_target) {

                                                HP -= 100;

                                                if (HP <= 0) {

                                                    function del() {

                                                        createjs.Tween.removeTweens(m);

                                                        exportRoot.removeChild(m);

                                                    }

                                                    if (is\_died) {

                                                        break;

                                                    } else {

                                                        score += 1;

                                                        is\_died = true;

                                                        createjs.Sound.play("boom").volume = set\_volume;

                                                        m.gotoAndPlay("boom");

                                                        setTimeout(del, 100);

                                                    }

                                                }

                                            }

                                        }

                                    }

                                    //月牙天衝碰撞判定

                                    for (let i = 0; i <= (a7\_array.length - 1); i++) {

                                        if (a7\_array[i] == 0) {

                                            continue;

                                        } else {

                                            let hit\_target = ndgmr.checkRectCollision(a7\_array[i], m);

                                            if (hit\_target) {

                                                HP -= 100;

                                                if (HP <= 0) {

                                                    function del() {

                                                        createjs.Tween.removeTweens(m);

                                                        exportRoot.removeChild(m);

                                                    }

                                                    if (is\_died) {

                                                        break;

                                                    } else {

                                                        score += 1;

                                                        is\_died = true;

                                                        createjs.Sound.play("boom").volume = set\_volume;

                                                        m.gotoAndPlay("boom");

                                                        setTimeout(del, 100);

                                                    }

                                                }

                                            }

                                        }

                                    }

                                    //雷球碰撞判定

                                    for (let i = 0; i <= (a2\_array.length - 1); i++) {

                                        if (a2\_array[i] == 0) {

                                            continue;

                                        } else {

                                            let hit\_target = ndgmr.checkRectCollision(a2\_array[i], m);

                                            if (hit\_target) {

                                                HP -= 2;

                                                if (HP <= 0) {

                                                    function del() {

                                                        createjs.Tween.removeTweens(m);

                                                        exportRoot.removeChild(m);

                                                    }

                                                    if (is\_died) {

                                                        break;

                                                    } else {

                                                        score += 1;

                                                        is\_died = true;

                                                        createjs.Sound.play("boom").volume = set\_volume;

                                                        m.gotoAndPlay("boom");

                                                        setTimeout(del, 100);

                                                    }

                                                }

                                                createjs.Tween.removeTweens(a2\_array[i]);

                                                exportRoot.removeChild(a2\_array[i]);

                                                a2\_array[i] = 0;

                                            }

                                        }

                                    }

                                    //飛船與怪物1碰撞判定

                                    if (ndgmr.checkRectCollision(ship, m)) {

                                        if (is\_died) {

                                            return

                                        } else {

                                            //飛船與怪物1碰撞後，將怪物1剩餘血量扣在玩家當前血量身上

                                            Hp -= HP;

                                            is\_died = true;

                                            createjs.Sound.play("boom").volume = set\_volume;

                                            m.gotoAndPlay("boom");

                                            //播放飛船毀損動畫

                                            ship.gotoAndPlay("hit");

                                            function del() {

                                                createjs.Tween.removeTweens(m);

                                                exportRoot.removeChild(m);

                                                //延後0.2秒後恢復飛船靜止狀態動畫(漂浮)

                                                ship.gotoAndPlay("stop");

                                            }

                                            setTimeout(del, 200);

                                        }

                                    }

                                    //當玩家遊戲失敗或遊戲勝利時關掉setInterval同時清除所有物件

                                    if (is\_gameover || is\_win) {

                                        createjs.Tween.removeTweens(m);

                                        exportRoot.removeChild(m);

                                        clearInterval(monster1);

                                    }

                                })

                        }

                    }

                },

                1000)

            //怪物2

            let monster2 = setInterval(() => {

                if (score >= 25 && score <= 75) {

                    if (is\_start && !is\_pause) {

                        let m = new lib.m2();

                        m.x = Math.floor(Math.random() \* (924 - 100 + 1) + 100);

                        m.y = -100;

                        exportRoot.addChild(m);

                        let HP = 2;

                        let is\_died = false;

                        createjs.Tween.get(m)

                            .to({

                                y: 768

                            }, 15000)

                            .call(() => {

                                is\_died = true;

                                Hp -= HP;

                                createjs.Tween.removeTweens(m);

                                exportRoot.removeChild(m);

                            })

                            .addEventListener("change", () => {

                                for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                    if (a1\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a1\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 1;

                                            if (HP >= 1) {

                                                //播放怪物2專屬碰撞音校

                                                createjs.Sound.play("hit1").volume = set\_volume;

                                            }

                                            if (HP == 1) {

                                                //播放怪物2憤怒動畫

                                                m.gotoAndPlay("run2");

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a1\_array[i]);

                                            exportRoot.removeChild(a1\_array[i]);

                                            a1\_array[i] = 0;

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a4\_array.length - 1); i++) {

                                    if (a4\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a4\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a7\_array.length - 1); i++) {

                                    if (a7\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a7\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a2\_array.length - 1); i++) {

                                    if (a2\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a2\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 2;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a2\_array[i]);

                                            exportRoot.removeChild(a2\_array[i]);

                                            a2\_array[i] = 0;

                                        }

                                    }

                                }

                                if (ndgmr.checkRectCollision(ship, m)) {

                                    if (is\_died) {

                                        return

                                    } else {

                                        Hp -= HP;

                                        is\_died = true;

                                        createjs.Sound.play("boom").volume = set\_volume;

                                        m.gotoAndPlay("boom");

                                        ship.gotoAndPlay("hit");

                                        function del() {

                                            createjs.Tween.removeTweens(m);

                                            exportRoot.removeChild(m);

                                            ship.gotoAndPlay("stop");

                                        }

                                        setTimeout(del, 200);

                                    }

                                }

                                if (is\_gameover || is\_win) {

                                    createjs.Tween.removeTweens(m);

                                    exportRoot.removeChild(m);

                                    clearInterval(monster2);

                                }

                            })

                        //怪物2的火球攻擊

                        setInterval(() => {

                            //當pause時return不往下執行

                            if (is\_pause) {

                                return;

                            }

                            if (is\_died) {

                                return

                            }

                            if (HP == 1) {

                                //隨機決定火球初始方向(左或右)

                                let left = Math.random() >= 0.5 ? false : true;

                                let f1 = new lib.fi1;

                                f1.x = (m.x);

                                f1.y = 0;

                                exportRoot.addChild(f1);

                                let is\_hit = false;

                                createjs.Tween.get(f1)

                                    .to({

                                        y: 768,

                                    }, 6000)

                                    .call(() => {

                                        createjs.Tween.removeTweens(f1);

                                        exportRoot.removeChild(f1);

                                    })

                                    .addEventListener("change", () => {

                                        if (is\_hit) {

                                            return;

                                        }

                                        //控制火球撞牆壁時反彈

                                        if (left) {

                                            f1.x -= 10;

                                        } else {

                                            f1.x += 10;

                                        }

                                        if (f1.x >= 1024) {

                                            left = true;

                                        }

                                        if (f1.x <= 0) {

                                            left = false;

                                        }

                                        let hit = ndgmr.checkRectCollision(ship, f1);

                                        if (hit) {

                                            is\_hit = true;

                                            //當火球擊中飛船時玩家血量-1

                                            Hp -= 1;

                                            //播放飛船受損音效

                                            createjs.Sound.play("harmed").volume = set\_volume;

                                            //播放飛船毀損動畫

                                            ship.gotoAndPlay("hit");

                                            createjs.Tween.removeTweens(f1);

                                            exportRoot.removeChild(f1);

                                            function delay() {

                                                ship.gotoAndPlay("stop");

                                            }

                                            setTimeout(delay, 200);

                                        }

                                        if (is\_gameover || is\_win) {

                                            createjs.Tween.removeTweens(f1);

                                            exportRoot.removeChild(f1);

                                        }

                                    })

                            }

                        }, 1200)

                    }

                }

            }, 1500)

            //怪物3

            let monster3 = setInterval(() => {

                if (score >= 50 && score <= 100) {

                    if (is\_start && !is\_pause) {

                        let m = new lib.m3();

                        m.x = Math.floor(Math.random() \* (874 - 150 + 1) + 150);

                        m.y = -100;

                        exportRoot.addChild(m);

                        let HP = 3;

                        let is\_died = false;

                        createjs.Tween.get(m)

                            .to({

                                y: 768

                            }, 20000)

                            .call(() => {

                                is\_died = true;

                                Hp -= HP;

                                createjs.Tween.removeTweens(m);

                                exportRoot.removeChild(m);

                            })

                            .addEventListener("change", () => {

                                for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                    if (a1\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a1\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 1;

                                            if (HP >= 1) {

                                                createjs.Sound.play("striking").volume = set\_volume;

                                            }

                                            //兩種暴怒型態

                                            if (HP == 2) {

                                                m.gotoAndPlay("run3");

                                            }

                                            if (HP == 1) {

                                                m.gotoAndPlay("run2");

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a1\_array[i]);

                                            exportRoot.removeChild(a1\_array[i]);

                                            a1\_array[i] = 0;

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a4\_array.length - 1); i++) {

                                    if (a4\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a4\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a7\_array.length - 1); i++) {

                                    if (a7\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a7\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a2\_array.length - 1); i++) {

                                    if (a2\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a2\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 2;

                                            if (HP >= 1) {

                                                createjs.Sound.play("striking").volume = set\_volume;

                                            }

                                            if (HP == 1) {

                                                m.gotoAndPlay("run2");

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a2\_array[i]);

                                            exportRoot.removeChild(a2\_array[i]);

                                            a2\_array[i] = 0;

                                        }

                                    }

                                }

                                if (ndgmr.checkRectCollision(ship, m)) {

                                    if (is\_died) {

                                        return;

                                    } else {

                                        is\_died = true;

                                        Hp -= HP;

                                        createjs.Sound.play("boom").volume = set\_volume;

                                        m.gotoAndPlay("boom");

                                        ship.gotoAndPlay("hit");

                                        function del() {

                                            createjs.Tween.removeTweens(m);

                                            exportRoot.removeChild(m);

                                            ship.gotoAndPlay("stop");

                                        }

                                        setTimeout(del, 200);

                                    }

                                }

                                if (is\_gameover || is\_win) {

                                    createjs.Tween.removeTweens(m);

                                    exportRoot.removeChild(m);

                                    clearInterval(monster3);

                                }

                            })

                        setInterval(() => {

                            if (is\_pause) {

                                return;

                            }

                            if (is\_died) {

                                return;

                            }

                            if (HP >= 1 && HP <= 2) {

                                let rot = 0;

                                let T = 2500;

                                if (HP == 1) {

                                    rot = 360;

                                    T = 1500;

                                }

                                let a5 = new lib.a5;

                                a5.x = (m.x - 100);

                                a5.y = 0;

                                exportRoot.addChild(a5);

                                let is\_hit = false;

                                createjs.Tween.get(a5)

                                    .to({

                                        y: 768,

                                        x: ship.x,

                                        rotation: rot,

                                    }, T)

                                    .call(() => {

                                        createjs.Tween.removeTweens(a5);

                                        exportRoot.removeChild(a5);

                                    })

                                    .addEventListener("change", () => {

                                        if (is\_hit) {

                                            return;

                                        }

                                        let hit = ndgmr.checkRectCollision(ship, a5);

                                        if (hit) {

                                            is\_hit = true;

                                            Hp -= 2;

                                            createjs.Sound.play("harmed").volume = set\_volume;

                                            ship.gotoAndPlay("hit");

                                            createjs.Tween.removeTweens(a5);

                                            exportRoot.removeChild(a5);

                                            function delay() {

                                                ship.gotoAndPlay("stop");

                                            }

                                            setTimeout(delay, 200);

                                        }

                                        if (is\_gameover || is\_win) {

                                            createjs.Tween.removeTweens(a5);

                                            exportRoot.removeChild(a5);

                                        }

                                    })

                            }

                        }, 1000)

                    }

                }

            }, 2000)

            //怪物4

            let monster4 = setInterval(() => {

                if (score >= 75 && score <= 125) {

                    if (is\_start && !is\_pause) {

                        let m = new lib.m4();

                        m.x = Math.floor(Math.random() \* (874 - 150 + 1) + 150);

                        m.y = -100;

                        exportRoot.addChild(m);

                        let HP = 6;

                        let is\_died = false;

                        createjs.Tween.get(m)

                            .to({

                                y: 768,

                            }, 20000)

                            .call(() => {

                                is\_died = true;

                                Hp -= HP;

                                createjs.Tween.removeTweens(m);

                                exportRoot.removeChild(m);

                            })

                            .addEventListener("change", () => {

                                for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                    if (a1\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a1\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 1;

                                            if (HP >= 1) {

                                                createjs.Sound.play("ping").volume = set\_volume;

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared1").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a1\_array[i]);

                                            exportRoot.removeChild(a1\_array[i]);

                                            a1\_array[i] = 0;

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a4\_array.length - 1); i++) {

                                    if (a4\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a4\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared1").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a7\_array.length - 1); i++) {

                                    if (a7\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a7\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared1").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a2\_array.length - 1); i++) {

                                    if (a2\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a2\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 2;

                                            if (HP >= 1) {

                                                createjs.Sound.play("ping").volume = set\_volume;

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("boom").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a2\_array[i]);

                                            exportRoot.removeChild(a2\_array[i]);

                                            a2\_array[i] = 0;

                                        }

                                    }

                                }

                                if (ndgmr.checkRectCollision(ship, m)) {

                                    if (is\_died) {

                                        return

                                    } else {

                                        is\_died = true;

                                        Hp -= HP;

                                        createjs.Sound.play("boom").volume = set\_volume;

                                        m.gotoAndPlay("boom");

                                        ship.gotoAndPlay("hit");

                                        function del() {

                                            createjs.Tween.removeTweens(m);

                                            exportRoot.removeChild(m);

                                            ship.gotoAndPlay("stop");

                                        }

                                        setTimeout(del, 200);

                                    }

                                }

                                if (is\_gameover || is\_win) {

                                    createjs.Tween.removeTweens(m);

                                    exportRoot.removeChild(m);

                                    clearInterval(monster4);

                                }

                            })

                        setInterval(() => {

                            if (is\_pause) {

                                return;

                            }

                            if (is\_died) {

                                return;

                            }

                            let r = new lib.ro;

                            r.x = 1124;

                            r.y = Math.floor(Math.random() \* 668);

                            exportRoot.addChild(r);

                            let is\_hit = false;

                            createjs.Tween.get(r)

                                .to({

                                    x: 0,

                                    y: 818,

                                }, (1000 + Math.floor(Math.random() \* 2000)))

                                .call(() => {

                                    createjs.Tween.removeTweens(r);

                                    exportRoot.removeChild(r);

                                })

                                .addEventListener("change", () => {

                                    if (is\_hit) {

                                        return;

                                    }

                                    let hit = ndgmr.checkRectCollision(ship, r);

                                    if (hit) {

                                        is\_hit = true;

                                        Hp -= 3;

                                        createjs.Sound.play("impact").volume = set\_volume;

                                        ship.gotoAndPlay("hit");

                                        function delay() {

                                            ship.gotoAndPlay("stop");

                                        }

                                        setTimeout(delay, 200);

                                    }

                                    if (is\_gameover || is\_win) {

                                        createjs.Tween.removeTweens(r);

                                        exportRoot.removeChild(r);

                                    }

                                })

                        }, Math.floor(Math.random() \* 3500 + 1500))

                    }

                }

            }, 2000)

            //怪物5

            let monster5 = setInterval(() => {

                if (score >= 100 && score <= 150) {

                    if (is\_start && !is\_pause) {

                        let m = new lib.m5();

                        m.x = Math.floor(Math.random() \* (924 - 100 + 1) + 100);

                        m.y = 0;

                        exportRoot.addChild(m);

                        let HP = 10;

                        let is\_died = false;

                        let left = Math.random() >= 0.5 ? false : true;

                        createjs.Tween.get(m)

                            .to({

                                y: 768,

                            }, 25000)

                            .call(() => {

                                Hp -= HP;

                                is\_died = true;

                                createjs.Tween.removeTweens(m);

                                exportRoot.removeChild(m);

                            })

                            .addEventListener("change", () => {

                                if (left) {

                                    m.x -= 8;

                                } else {

                                    m.x += 8;

                                }

                                if (m.x >= 949) {

                                    left = true;

                                }

                                if (m.x <= 0) {

                                    left = false;

                                }

                                for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                    if (a1\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a1\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 1;

                                            if (HP >= 1) {

                                                createjs.Sound.play("doong").volume = set\_volume;

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared2").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a1\_array[i]);

                                            exportRoot.removeChild(a1\_array[i]);

                                            a1\_array[i] = 0;

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a4\_array.length - 1); i++) {

                                    if (a4\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a4\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared2").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a7\_array.length - 1); i++) {

                                    if (a7\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a7\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 100;

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared2").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                        }

                                    }

                                }

                                for (let i = 0; i <= (a2\_array.length - 1); i++) {

                                    if (a2\_array[i] == 0) {

                                        continue;

                                    } else {

                                        let hit\_target = ndgmr.checkRectCollision(a2\_array[i], m);

                                        if (hit\_target) {

                                            HP -= 2;

                                            if (HP >= 1) {

                                                createjs.Sound.play("doong").volume = set\_volume;

                                            }

                                            if (HP <= 0) {

                                                function del() {

                                                    createjs.Tween.removeTweens(m);

                                                    exportRoot.removeChild(m);

                                                }

                                                if (is\_died) {

                                                    break;

                                                } else {

                                                    score += 1;

                                                    is\_died = true;

                                                    createjs.Sound.play("devil\_scared2").volume = set\_volume;

                                                    m.gotoAndPlay("boom");

                                                    setTimeout(del, 100);

                                                }

                                            }

                                            createjs.Tween.removeTweens(a2\_array[i]);

                                            exportRoot.removeChild(a2\_array[i]);

                                            a2\_array[i] = 0;

                                        }

                                    }

                                }

                                if (ndgmr.checkRectCollision(ship, m)) {

                                    if (is\_died) {

                                        return

                                    } else {

                                        is\_died = true;

                                        Hp -= HP;

                                        createjs.Sound.play("boom").volume = set\_volume;

                                        m.gotoAndPlay("boom");

                                        ship.gotoAndPlay("hit");

                                        function del() {

                                            createjs.Tween.removeTweens(m);

                                            exportRoot.removeChild(m);

                                            ship.gotoAndPlay("stop");

                                        }

                                        setTimeout(del, 200);

                                    }

                                }

                                if (is\_gameover || is\_win) {

                                    createjs.Tween.removeTweens(m);

                                    exportRoot.removeChild(m);

                                    clearInterval(monster5);

                                }

                            })

                        setInterval(() => {

                            if (is\_pause) {

                                return;

                            }

                            if (is\_died) {

                                return;

                            }

                            let f2 = new lib.fi2;

                            f2.x = Math.floor(Math.random() \* (924 - 100 + 1) + 100);

                            f2.y = 0;

                            exportRoot.addChild(f2);

                            let enlarge = Math.floor(Math.random() \* 768);

                            let is\_hit = false;

                            let is\_enlarge = false;

                            createjs.Tween.get(f2)

                                .to({

                                    y: 768,

                                }, 1500)

                                .call(() => {

                                    createjs.Tween.removeTweens(f2);

                                    exportRoot.removeChild(f2);

                                })

                                .addEventListener("change", () => {

                                    if (is\_hit) {

                                        return;

                                    }

                                    if (f2.y > enlarge && !is\_enlarge) {

                                        f2.scaleX = 2;

                                        f2.scaleY = 2;

                                        is\_enlarge = true;

                                    }

                                    let hit = ndgmr.checkRectCollision(ship, f2);

                                    if (hit) {

                                        is\_hit = true;

                                        Hp -= 4;

                                        createjs.Sound.play("harmed").volume = set\_volume;

                                        ship.gotoAndPlay("hit");

                                        function delay() {

                                            ship.gotoAndPlay("stop");

                                        }

                                        setTimeout(delay, 200);

                                    }

                                    if (is\_gameover || is\_win) {

                                        createjs.Tween.removeTweens(f2);

                                        exportRoot.removeChild(f2);

                                    }

                                })

                        }, Math.floor(Math.random() \* 2000 + 1500))

                    }

                }

            }, 2500)

            function html() {

                createjs.Sound.play("boss").volume = set\_volume;

                let boss = new lib.hht();

                boss.x = 540;

                boss.y = 100;

                exportRoot.addChild(boss);

                let is\_died = false;

                let is\_hit = false;

                let HP = 100;

                function b1() {

                    ship.gotoAndPlay("stop");

                    is\_hit = false;

                    createjs.Tween.get(boss)

                        .to({

                            x: Math.floor(Math.random() \* (924 - 100 + 1) + 100),

                            y: Math.floor(Math.random() \* (450 - 50 + 1) + 50),

                        }, 1000)

                        .call(() => {

                            if (Math.random() > 0.6 ? true : false) {

                                b3();

                            } else {

                                b2();

                            }

                        })

                        .addEventListener("change", () => {

                            for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                if (a1\_array[i] == 0) {

                                    continue;

                                } else {

                                    let hit\_target = ndgmr.checkRectCollision(a1\_array[i], boss);

                                    if (hit\_target) {

                                        HP -= 1;

                                        if (HP >= 1) {

                                            createjs.Sound.play("ping").volume = set\_volume;

                                        }

                                        if (HP <= 0) {

                                            is\_win = true;

                                            createjs.Sound.play("boom").volume = set\_volume;

                                            createjs.Tween.removeTweens(boss);

                                            exportRoot.removeChild(boss);

                                        }

                                        createjs.Tween.removeTweens(a1\_array[i]);

                                        exportRoot.removeChild(a1\_array[i]);

                                        a1\_array[i] = 0;

                                    }

                                }

                            }

                            if (is\_gameover) {

                                createjs.Tween.removeTweens(boss);

                                exportRoot.removeChild(boss);

                            }

                        })

                }

                function b2() {

                    createjs.Tween.get(boss)

                        .to({

                            x: Math.floor(Math.random() \* (924 - 100 + 1) + 100),

                            y: Math.floor(Math.random() \* (450 - 50 + 1) + 50),

                        }, 1000)

                        .call(() => {

                            b1();

                        })

                        .addEventListener("change", () => {

                            for (let i = 0; i <= (a1\_array.length - 1); i++) {

                                if (a1\_array[i] == 0) {

                                    continue;

                                } else {

                                    let hit\_target = ndgmr.checkRectCollision(a1\_array[i], boss);

                                    if (hit\_target) {

                                        HP -= 1;

                                        if (HP >= 1) {

                                            createjs.Sound.play("ping").volume = set\_volume;

                                        }

                                        if (HP <= 0) {

                                            is\_win = true;

                                            createjs.Sound.play("boom").volume = set\_volume;

                                            createjs.Tween.removeTweens(boss);

                                            exportRoot.removeChild(boss);

                                        }

                                        createjs.Tween.removeTweens(a1\_array[i]);

                                        exportRoot.removeChild(a1\_array[i]);

                                        a1\_array[i] = 0;

                                    }

                                }

                            }

                            if (is\_gameover) {

                                createjs.Tween.removeTweens(boss);

                                exportRoot.removeChild(boss);

                            }

                        })

                }

                function b3() {

                    createjs.Tween.get(boss)

                        .to({

                            x: ship.x,

                            y: ship.y,

                        }, 100)

                        .call(() => {

                            b2();

                        })

                        .addEventListener("change", () => {

                            if (ndgmr.checkRectCollision(ship, boss)) {

                                if (is\_hit) {

                                    return;

                                } else {

                                    is\_hit = true;

                                    Hp -= 10;

                                    ship.gotoAndPlay("hit");

                                    createjs.Sound.play("harmed").volume = set\_volume;

                                }

                            }

                        })

                }

                b1();

            }

            function css() {

                let ccs = new lib.ccs();

                ccs.x = 100;

                ccs.y = 500;

                exportRoot.addChild(ccs);

                createjs.Tween.get(ccs, {

                        loop: true

                    })

                    .to({

                        x: 924,

                    }, (Math.floor(Math.random() \* 3000) + 1000))

                    .to({

                        x: 100,

                    }, (Math.floor(Math.random() \* 3000) + 1000))

                    .addEventListener("change", () => {

                        for (let i = 0; i <= (a1\_array.length - 1); i++) {

                            if (a1\_array[i] == 0) {

                                continue;

                            } else {

                                let hit\_target = ndgmr.checkRectCollision(a1\_array[i], ccs);

                                if (hit\_target) {

                                    createjs.Tween.removeTweens(a1\_array[i]);

                                    exportRoot.removeChild(a1\_array[i]);

                                    a1\_array[i] = 0;

                                }

                            }

                        }

                        if (is\_win || is\_gameover) {

                            createjs.Tween.removeTweens(ccs);

                            exportRoot.removeChild(ccs);

                        }

                    })

            }

            function js() {

                let jjs = new lib.jjs();

                jjs.x = 924;

                jjs.y = 350;

                exportRoot.addChild(jjs);

                createjs.Tween.get(jjs, {

                        loop: true

                    })

                    .to({

                        x: 100,

                    }, (Math.floor(Math.random() \* 3000) + 1000))

                    .to({

                        x: 924,

                    }, (Math.floor(Math.random() \* 3000) + 1000))

                    .addEventListener("change", () => {

                        for (let i = 0; i <= (a1\_array.length - 1); i++) {

                            if (a1\_array[i] == 0) {

                                continue;

                            } else {

                                let hit\_target = ndgmr.checkRectCollision(a1\_array[i], jjs);

                                if (hit\_target) {

                                    createjs.Tween.removeTweens(a1\_array[i]);

                                    exportRoot.removeChild(a1\_array[i]);

                                    a1\_array[i] = 0;

                                }

                            }

                        }

                        if (is\_win || is\_gameover) {

                            createjs.Tween.removeTweens(jjs);

                            exportRoot.removeChild(jjs);

                        }

                    })

            }

            function boss() {

                html();

                css();

                js();

            }

            //鍵盤按下事件

            function KDF(e) {

                //回傳按下鍵盤按鍵的keyCode

                console.log(e.keyCode);

                //鍵盤keyCode紀錄 37左 38上 39右 40下

                //左右方向控制

                if (!is\_pause) {

                    if (e.keyCode === 37 || e.keyCode === 39) {

                        //執行一次後直接return避免重複執行，影像一直重新loading

                        if (is\_KD) return;

                        //鍵盤按下為真

                        is\_KD = true;

                        //水平移動方向控制

                        H = e.keyCode === 39 ? 1 : -1;

                        //移動時飛船播放run狀態(噴火)

                        ship.gotoAndPlay("run");

                    }

                    //上下方向控制

                    if (e.keyCode === 38 || e.keyCode === 40) {

                        //執行一次後直接擋掉避免重複執行，影像一直重新loading

                        if (is\_KD) return;

                        //鍵盤按下為真

                        is\_KD = true;

                        //垂直移動方向控制

                        V = e.keyCode === 38 ? -1 : 1;

                        //移動時飛船播放run狀態(噴火)

                        ship.gotoAndPlay("run");

                    }

                    //按空白鍵發射子彈

                    if (e.keyCode === 32) {

                        console.log("a1子彈數" + a1\_num + "顆");

                        //播放a1子彈發射音效

                        createjs.Sound.play("attack1").volume = set\_volume;

                        //存一個區域變數記錄a1子彈將存放位置

                        let tmp = a1\_num;

                        //a1子彈個數加1

                        a1\_num += 1;

                        //當累積a1子彈數等於100時a1子彈數歸零

                        if (a1\_num > 100) {

                            a1\_num = 0;

                        }

                        //呼叫function a1\_generate()產生a1子彈後存入陣列a1\_array之第tmp的位置

                        a1\_array[tmp] = a1\_generate();

                        //抓取當前飛船的位置

                        let ship\_x = (ship.x + 12);

                        let ship\_y = (ship.y - 23);

                        //設定a1子彈產生的位置

                        a1\_array[tmp].x = ship\_x;

                        a1\_array[tmp].y = ship\_y;

                        //螢幕上顯示a1子彈

                        exportRoot.addChild(a1\_array[tmp]);

                        //用HTML5canvas動畫工具createjs輔助

                        createjs.Tween.get(a1\_array[tmp])

                            //設定a1子彈產生後由起始點移動到y=-500位置(2.5秒內完成)，

                            .to({

                                y: -100

                            }, 1500)

                            //設定當a1子彈到達指定位置後要執行的指令

                            .call(() => {

                                createjs.Tween.removeTweens(a1\_array[tmp]);

                                exportRoot.removeChild(a1\_array[tmp]);

                                a1\_array[tmp] = 0;

                            })

                    }

                    //按Q發射雷球

                    if (e.keyCode === 81 && Mp >= 1) {

                        Mp -= 1;

                        console.log("a2雷球數" + a2\_num + "顆");

                        createjs.Sound.play("lightning").volume = set\_volume;

                        let tmp = a2\_num;

                        a2\_num += 1;

                        if (a2\_num > 300) {

                            a2\_num = 0;

                        }

                        a2\_array[tmp] = a2\_generate();

                        let ship\_x = (ship.x + 7);

                        let ship\_y = (ship.y - 40);

                        a2\_array[tmp].x = ship\_x;

                        a2\_array[tmp].y = ship\_y;

                        a2\_array[tmp].scaleX = 0.5;

                        a2\_array[tmp].scaleY = 0.5;

                        exportRoot.addChild(a2\_array[tmp]);

                        createjs.Tween.get(a2\_array[tmp])

                            .to({

                                rotation: 180,

                            }, 10000)

                        createjs.Tween.get(a2\_array[tmp])

                            .to({

                                y: -100

                            }, 6000)

                            .call(() => {

                                createjs.Tween.removeTweens(a2\_array[tmp]);

                                exportRoot.removeChild(a2\_array[tmp]);

                                a2\_array[tmp] = 0;

                            })

                    }

                    //按W發射月牙天衝

                    if (e.keyCode === 87 && Mp >= 5) {

                        Mp -= 5;

                        console.log("a7月牙天衝數" + a7\_num + "個");

                        createjs.Sound.play("attack7").volume = set\_volume;

                        let tmp = a7\_num;

                        a7\_num += 1;

                        if (a7\_num > 100) {

                            a7\_num = 0;

                        }

                        a7\_array[tmp] = a7\_generate();

                        let ship\_x = (ship.x + 50);

                        let ship\_y = (ship.y + 10);

                        a7\_array[tmp].x = ship\_x;

                        a7\_array[tmp].y = ship\_y;

                        exportRoot.addChild(a7\_array[tmp]);

                        createjs.Tween.get(a7\_array[tmp])

                            .to({

                                y: -100

                            }, 2500)

                            .call(() => {

                                createjs.Tween.removeTweens(a7\_array[tmp]);

                                exportRoot.removeChild(a7\_array[tmp]);

                                a7\_array[tmp] = 0;

                            })

                    }

                    //按R發射脈衝波

                    if (e.keyCode === 82 && Mp >= 10) {

                        Mp -= 10;

                        console.log("a4脈衝波數" + a4\_num + "個");

                        createjs.Sound.play("lazer").volume = set\_volume;

                        let tmp = a4\_num;

                        a4\_num += 1;

                        if (a4\_num > 100) {

                            a4\_num = 0;

                        }

                        a4\_array[tmp] = a4\_generate();

                        let ship\_y = (ship.y - 20);

                        a4\_array[tmp].x = 0;

                        a4\_array[tmp].y = ship\_y;

                        exportRoot.addChild(a4\_array[tmp]);

                        createjs.Tween.get(a4\_array[tmp])

                            .to({

                                y: -100

                            }, 2000)

                            .call(() => {

                                createjs.Tween.removeTweens(a4\_array[tmp]);

                                exportRoot.removeChild(a4\_array[tmp]);

                                a4\_array[tmp] = 0;

                            })

                    }

                }

                //按S回滿血/藍

                if (e.keyCode === 83) {

                    createjs.Sound.play("hp\_up").volume = set\_volume;

                    Hp = 100;

                    Mp = 100;

                }

                //按P遊戲暫停

                if (e.keyCode === 80) {

                    if (!is\_pause) {

                        createjs.Sound.stop();

                        createjs.Ticker.paused = true;

                        is\_pause = true;

                    } else {

                        createjs.Ticker.paused = false;

                        is\_pause = false;

                    }

                }

            }

            //鍵盤放開事件

            function KUF(e) {

                if (e.keyCode === 37 || e.keyCode == 39) {

                    //鍵盤按下為否

                    is\_KD = false;

                    //停止移動時飛船撥放stop狀態(飛船漂浮)

                    ship.gotoAndPlay("stop");

                    //將控制水平移動方向值歸0

                    H = 0;

                }

                if (e.keyCode === 38 || e.keyCode == 40) {

                    is\_KD = false;

                    ship.gotoAndPlay("stop");

                    //將控制垂直移動方向值歸0

                    V = 0;

                }

            }

            //遊戲事件變化

            function change() {

                //控制飛船移動同時限制飛船移動範圍

                //左右移動

                if (H == 1 && ship.x <= 974) {

                    //向右移動

                    ship.x += D \* H;

                }

                if (H == -1 && ship.x >= 25) {

                    //向左移動

                    ship.x += D \* H;

                }

                //上下移動

                if (V == -1) {

                    ship.y += D \* V;

                }

                if (V == 1 && ship.y <= 718) {

                    ship.y += D \* V;

                }

                //遊戲分數大於150分時飛船往前開出遊戲畫面直接勝利

                if (ship.y <= -100 && score >= 150) {

                    is\_win = true;

                }

                //更新遊戲得分、血量、藍量

                //改成jquery版本

                $('.score').text(score);

                $('.hp').css("width", Hp + "%");

                $('.mp').css("width", Mp + "%");

                //物件導向版本如下

                // document.querySelector(".score").innerHTML = score;

                // document.querySelector(".hp").style.width = `${Hp}%`;

                // document.querySelector(".mp").style.width = `${Mp}%`;

                if (Hp <= 0) {

                    //遊戲失敗為真

                    is\_gameover = true;

                    //播放遊戲失敗音效

                    createjs.Sound.play("game\_over").volume = set\_volume;

                    //改成jquery版本

                    $('#game\_over').css("display", "flex");

                    $('#restart').css("display", "flex");

                    //物件導向版本如下

                    // document.querySelector("#game\_over").style.display = "flex";

                    // document.querySelector("#restart").style.display = "flex";

                    //關掉鍵盤事件監聽

                    window.removeEventListener("keydown", KDF);

                    window.removeEventListener("keyup", KUF);

                    //關掉定時更新遊戲事件變化

                    createjs.Ticker.removeEventListener("tick", change);

                    function boom() {

                        createjs.Sound.play("boom").volume = set\_volume;

                        ship.gotoAndPlay("boom");

                        function del() {

                            createjs.Tween.removeTweens(ship);

                            exportRoot.removeChild(ship);

                        }

                        setTimeout(del, 300);

                    }

                    setTimeout(boom, 2500);

                }

                if (is\_win) {

                    createjs.Sound.play("win").volume = set\_volume;

                    //改成jquery版本

                    $('#win\_game').css("display", "flex");

                    $('#restart').css("display", "flex");

                    //物件導向版本如下

                    // document.querySelector("#win\_game").style.display = "flex";

                    // document.querySelector("#restart").style.display = "flex";

                    //關掉鍵盤事件監聽

                    window.removeEventListener("keydown", KDF);

                    window.removeEventListener("keyup", KUF);

                    //關掉定時更新遊戲事件變化

                    createjs.Ticker.removeEventListener("tick", change);

                    if (ship.y >= 0) {

                        ship.gotoAndPlay("run");

                        createjs.Tween.get(ship)

                            .to({

                                y: -100

                            }, 5000)

                            .call(() => {

                                createjs.Tween.removeTweens(ship);

                                exportRoot.removeChild(ship);

                            })

                    }

                    if (ship.y < 0) {

                        createjs.Tween.removeTweens(ship);

                        exportRoot.removeChild(ship);

                    }

                }

                //當遊戲分數達150分時魔王出現

                if (score == 150 && !boss\_show) {

                    boss\_show = true;

                    setTimeout(boss, 500);

                }

            }

            //遊戲畫面更新

            fnStartAnimation = function () {

                createjs.Ticker.framerate = lib.properties.fps;

                createjs.Ticker.addEventListener("tick", stage);

            }

            //支援螢幕縮放

            AdobeAn.makeResponsive(false, 'both', false, 1, [canvas, anim\_container, dom\_overlay\_container]);

            AdobeAn

                .compositionLoaded(lib.properties.id);

            fnStartAnimation();

        }

    </script>

    <!-- 引用jquery -->

    <script src="https://code.jquery.com/jquery-3.5.1.js"

        integrity="sha256-QWo7LDvxbWT2tbbQ97B53yJnYU3WhH/C8ycbRAkjPDc=" crossorigin="anonymous"></script>

</head>

<body onload="init();" style="margin:0px;cursor: default;">

    <!-- 將遊戲畫面包在div.container內，預設置中 -->

    <div class="container">

        <!-- 得分 -->

        <h1 class="score">0</h1>

        <!-- 血條 -->

        <div class="hpBar">

            <h2>HP</h2>

            <div class="hpBox">

                <div class="hp"></div>

            </div>

        </div>

        <!-- 藍條 -->

        <div class="mpBar">

            <h2>MP</h2>

            <div class="mpBox">

                <div class="mp"></div>

            </div>

        </div>

        <!-- 音量 -->

        <div id="volume">

            <img src="./img/S0.6.png" style="position: absolute; left: 100%;top: 2%;width: 8%;cursor: pointer;" id="V">

        </div>

        <!-- 重新 -->

        <div id="restart1">

            <img src="./img/RESTART1.png"

                style="position: absolute; left: 120%;top: 2.5%;width: 6%;cursor: pointer;opacity: 0.8;">

        </div>

        <!-- 暫停 -->

        <div id="pause">

            <img src="./img/PAUSE.png"

                style="position: absolute; left: 110%;top:2.5%;width: 6%;cursor: pointer;opacity: 0.8;" id="P">

        </div>

        <!-- 開始 -->

        <div id="start"><img src="./img/S\_\_4276288.gif"></div>

        <!-- 遊戲失敗 -->

        <div id="game\_over"><img src="./img/GAME\_OVER.png"></div>

        <!-- 再玩一次 -->

        <div id="restart"><img src="./img/RESTART.png"></div>

        <!-- 遊戲勝利 -->

        <div id="win\_game"><img src="./img/WIN\_GAME.jpg"></div>

        <!-- 遊戲版面 -->

        <div id="animation\_container" style="background-color:rgba(255, 255, 255, 1.00); width:1024px; height:768px">

            <canvas id="canvas" width="1024" height="768"

                style="position: absolute; display: block; background-color:rgba(255, 255, 255, 1.00);z-index: 1;"></canvas>

            <div id="dom\_overlay\_container"

                style="pointer-events:none; overflow:hidden; width:1024px; height:768px; position: absolute; left: 0px; top: 0px; display: block;">

            </div>

        </div>

        <div id="intro">

            <img src="./img/遊戲說明.jpg" style="width: 100%;">

        </div>

        <!-- 遊戲版權&關於作者 -->

        <div>

            <footer>

                copyright &copy; Jim space\_war

                <h1>(2020前端程式設計)</h1>

                <a href="https://b06608062.github.io/" id="about\_me" target="\_blank">About\_me</a>

            </footer>

        </div>

    </div>

</body>

</html>