```
<html>
    <head>
        <title>Tank Simulation</title>

        <script language="javascript">
            //

            //makes variables
            var invaderSpeedX = 5;
            var invaderSpeedY = 40;

            var invaderHealth = 10;

            var colorOne = "#0000FF";

            //defines the boundaries of the canvas as varaibles to make it easier to hot swap
            var canvasBoundX = 700;
            var canvasBoundY = 500;

            var speed = 6;

            var xDef = 350;
            var yDef = 450;

            var xInvaders = 350;
            var yInvaders = 50;

            var targetX =  Math.floor(Math.random() * (canvasBoundX - 100)) + 50;
            var targetY =  Math.floor(Math.random() * (canvasBoundY - 100)) + 50;
            var targetActive = true;

            var currentMovement = "none";
            var previousMovement = "none";

            var bulletShot = false;
            var bulletX = 0;
            var bulletY = 0;
            var bulletDirection = "none";

            var timestamp = 0;
            var totalTime = 60000;
            var totalShots = 0;
            var targetsHit = 0;

            var isOn = false;
            //used for timer to determine if it is on or not

            //adds keyboard listener
            window.addEventListener("keydown", function(event){
                //changes keyboard input into diretions
                switch(event.key){
                    case "a":
                        currentMovement = "left";
                        break;
                    case "d":
                        currentMovement = "right";
                        break;
                    case "w":
                        currentMovement = "up";
                        break;
                    case "s":
                        currentMovement = "down";
                        break;
                    default:
                        currentMovement = "none";
                        break;
```

```
        }

        if(currentMovement != "none")
            previousMovement = currentMovement;

    }, true);

    window.addEventListener("keyup", function(event){
        currentMovement = "none";

    }, true);


    //fill background and turn  on timer when the body gets initialized
    function initialize(){
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");

        context.fillStyle="#ADD8E6";
        context.fillRect(0, 0, canvasBoundX, canvasBoundY);

        turnOn();
    }

    //periodically called ever 20ms to update the game
    function update(){
        resetBackground();
        writeText(timestamp / 1000, 10, 50);
        writeText(totalTime / 1000, 10, 100);
        writeText("Total Shots: " + totalShots, 500, 50);
        writeText("Points: " + targetsHit, 500, 90);
        moveDef();
        drawTarget();
        moveBullet();
        timestamp += 20;
        totalTime -= 20;

        if(totalTime <= 0){
            toggleTimer();
            resetBackground();
            writeText("Game Over!", canvasBoundX / 2, canvasBoundY / 2);
        }
    }

    //writes text on the screen given what to write and the location
    function writeText(text, x, y){
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");

        context.font = "30px Arial";
        context.fillStyle = "#FF0000";
        context.fillText(text, x, y);
    }

    //moves the bullet across the screen based on the direction
    function moveBullet(){
        if(bulletShot == true){
            var canvas = document.getElementById("myCanvas");
            var context = canvas.getContext("2d");

            if(bulletDirection == "left")
                bulletX -= 10;
            else if(bulletDirection == "right")
                bulletX += 10;
            else if(bulletDirection == "down")
                bulletY += 10;
            else if(bulletDirection == "up")
```

```
                    bulletY -= 10;
                else
                    bulletX -= 10;

                context.fillStyle = "#FF0000";
                context.beginPath();
                context.arc(bulletX, bulletY, 8, 0, 2 * Math.PI, true);
                context.closePath();
                context.fill();

                //if the bullet hits anything reset it
                if(bulletY <= 10 || bulletY >= canvasBoundY - 10 || bulletX <= 10 ||
                bulletX >= canvasBoundX - 10 || hitTarget()){
                    context.strokeStyle = "#ADD8E6";
                    context.stroke();
                    bulletShot = false;
                }

                //if it hits the target specifically then reset the target
                if(hitTarget()){
                    targetActive = false;
                    targetsHit++;
                    placeTarget();
                }
            }
        }

        //places the target to the given coordinate
        function drawTarget(){
            var canvas = document.getElementById("myCanvas");
            var context = canvas.getContext("2d");

            if(targetActive){
                context.fillStyle = "#FF0000";
                context.beginPath();
                context.arc(targetX, targetY, 16, 0, 2 * Math.PI, true);
                context.closePath();
                context.fill();

                context.fillStyle = "#FFFFFF";
                context.beginPath();
                context.arc(targetX, targetY, 12, 0, 2 * Math.PI, true);
                context.closePath();
                context.fill();

                context.fillStyle = "#FF0000";
                context.beginPath();
                context.arc(targetX, targetY, 8, 0, 2 * Math.PI, true);
                context.closePath();
                context.fill();

                context.fillStyle = "#FFFFFF";
                context.beginPath();
                context.arc(targetX, targetY, 4, 0, 2 * Math.PI, true);
                context.closePath();
                context.fill();

                if(timestamp > 1500){
                    targetActive = false;
                    placeTarget();
                }
            }
        }

        //redraws the target at a different location
        function placeTarget(){
            if(!targetActive){
```

```
                targetX = Math.floor(Math.random() * (canvasBoundX - 100)) + 50;
                targetY = Math.floor(Math.random() * (canvasBoundY - 100)) + 50;

                timestamp = 0;
                targetActive = true;
            }
        }

        //toggles firing the bullet
        function fireBullet(){
            if(!bulletShot){
                bulletX = xDef;
                bulletY = yDef;
                bulletDirection = previousMovement;
                totalShots++;
            }

            bulletShot = true;

        }

        //moves the tank based on the given direction
        function moveDef(){
            switch(currentMovement){
                case "left":
                    if(xDef > 20)
                        xDef -= speed;
                    moveDefender(xDef, yDef, previousMovement);
                    break;
                case "right":
                    if(xDef < canvasBoundX - 20)
                        xDef += speed;
                    moveDefender(xDef, yDef, previousMovement);
                    break;
                case "up":
                    if(yDef >= 20)
                        yDef -= speed;
                    moveDefender(xDef, yDef, previousMovement);
                    break;
                case "down":
                    if(yDef <= canvasBoundY - 20)
                        yDef += speed;
                    moveDefender(xDef, yDef, previousMovement);
                    break;
                default:
                    moveDefender(xDef, yDef, previousMovement);
                    break;

            }
        }

        //used to reset the background to avoid smearing
        function resetBackground(){
            var canvas = document.getElementById("myCanvas");
            var context = canvas.getContext("2d");
            //paint the background of the canvas
            context.fillStyle="#ADD8E6";
            context.fillRect(0, 0, canvasBoundX, canvasBoundY);
        }

        //places the defender to a given location and moves the turret based on the
        direction
        function moveDefender(x, y, direction){
            var canvas = document.getElementById("myCanvas");
            var context = canvas.getContext("2d");

            var directX = 0;
```

```
                var directY = 0;

                context.fillStyle = colorOne;
                context.beginPath();
                context.arc(x, y, 15, 0, 2 * Math.PI, true);
                context.closePath();
                context.fill();

                if(direction == "up")
                    directY = -1;
                else if(direction == "down")
                    directY = 1;
                else if(direction == "left")
                    directX = -1;
                else if(direction == "right")
                    directX = 1;

                context.fillStyle = colorOne;
                context.beginPath();
                context.arc(x + (15 * directX), y + (15 * directY), 7.5, 0, 2 * Math.PI,
                true);
                context.closePath();
                context.fill();
            }

            //returns true if the bullet is within the hitbox of the target
            function hitTarget(){
                return Math.abs(bulletX - targetX) < 16 && Math.abs(bulletY - targetY) <
                16;
            }

            //toggles the timer
            function toggleTimer(){
                //if the timer is on the turn it off, if its not then turn it on
                isOn ? turnOff() : turnOn();
            }

            //manually turns on the timer
            function turnOn(){
                //turns the timer on
                timer = setInterval("update()", 20);
                isOn = true;
            }

            //manually turns off the timer
            function turnOff(){
                //turns the timer off
                clearInterval(timer);
                isOn = false;
            }
        </script>
    </head>

    <body onload="initialize()" onclick="fireBullet()">
        <center>
            <h1>Tank Simulation</h1>

    </br>
    </br>

        <canvas id="myCanvas" width="700" height="500"
            style="border:2px solid rgb(195, 195, 195);">
            Your browser does not suppport the canvas element
        </canvas>

    </br>
    </br>
```

```
331             </center>
332         </body>
333     </html>
```