

GOVT. ITI. KADMA



(CHARKHI DADRI)

Session :- 2024 - 25



Project File :- Zombie Game in Html

Submitted To :-

Sh. Vinod Kumar
Computar Instructor

Submitted By :-

Poonam , Manju Kumari
, Savita , Rajat , Surender
, Amandeep

DECLARATION

We (Poonam , Manju kumari , Savita , Rajat , Surender , Amandeep) are student of **COPA Trade** at **Government ITI Kadma**, hereby declare that the project titled "**Zombie Game** " is my original work and has been prepared as part of the curriculum requirements for the **COPA Trade** course. This project is submitted for educational purposes only.

I confirm that all the information, data, and content used in this project have been duly referenced and credited to their respective sources. No part of this project has been copied or reproduced from any other source without proper acknowledgment.

This project has been completed under the guidance of **Sh.Vinod Kumar**, and I am solely responsible for any errors or omissions in this work.

I also declare that this project has not been submitted elsewhere for any other purpose.

Student Name: Poonam , Manju kumari , Savita , Rajat , Surender , Amandeep
COPA Trade, Government ITI Kadma

Signature: Poonam , Manju kumari , Savita , Rajat , Surender , Amandeep

Date: _____

Supervisor/Instructor Declaration

This project has been reviewed and approved as part of the academic requirements for the **COPA Trade** course at **Government ITI Kadma**.

Instructor Name: SH. Vinod Kumar

Signature: _____

Date: _____

Acknowledgement

We would like to express our sincere gratitude to all those who have contributed to the successful completion of this project report for the COPA (Computer Operator and Programming Assistant) trade at Government ITI, Kadma. This project would not have been possible without the guidance, support, and encouragement of numerous individuals and organizations.

First and foremost, we extend our heartfelt thanks to the **Government ITI, Kadma**, for providing us with the opportunity to undertake this project and for offering the necessary resources and infrastructure. We are deeply indebted to our esteemed **Principal** and **Instructors** for their constant guidance, valuable insights, and encouragement throughout the project.

We would also like to thank our **Project Guide** for their unwavering support, patience, and expert advice, which helped us navigate through the challenges and complexities of this project. Their constructive feedback and suggestions were instrumental in shaping the final outcome.

Our sincere appreciation goes to our **classmates and peers** for their cooperation, motivation, and shared knowledge, which made this journey enjoyable and enriching. Special thanks to the **non-teaching staff** of the institute for their assistance and support in various aspects of the project.

Lastly, we are grateful to our **family and friends** for their constant encouragement and belief in our abilities, which kept us motivated throughout the project.

This project has been a significant learning experience, and we are thankful to everyone who played a role, directly or indirectly, in its successful completion.

Thank You.

Poonam , Manju kumari , Savita , Rajat , Surender , Amandeep

COPA (NcvT)/ 1st group
Government ITI, Kadma
Date

Program Code :-

Sr. NO.	Topics	Page No.
1	Declaration	2
2	Acknowledgment	3
3	Program Code	6-87
4	Output	88-91

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

  <meta http-equiv="X-UA-Compatible" content="IE=edge" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <title>Rajat Project Session=2024-2025</title>

<!-- browser information .css file -->

<link rel="stylesheet" href="style1.css" />

<style>

  /* font */

  @font-face {

    font-family: "Cinzel";

    font-style: normal;

    font-weight: 400;

    src: local("Cinzel-Regular"),

        url("https://s3-us-west-

2.amazonaws.com/s.cdn.io/250758/i2BwM1Eq2JyiNOY_VrkubOvvDin1pK8aKteLpeZ5c0A.woff2")

        format("woff2");

    unicode-range: U+0000-00FF, U+0131, U+0152-0153, U+02C6, U+02DA, U+02DC,

        U+2000-206F, U+2074, U+20AC, U+2212, U+2215, U+E0FF, U+EFFD, U+F000;

  }
```

```
/* removing unwanted padding */
```

```
.info-board,
```

```
#footer p,
```

```
html,
```

```
body {
```

```
    margin: 0;
```

```
    padding: 0;
```

```
}
```

```
/* to center all items */
```

```
body {
```

```
    display: flex;
```

```
    justify-content: center;
```

```
    align-items: center;
```

```
}
```

```
*,
```

```
*:before,
```

```
*:after {
```

```
    box-sizing: border-box;
```

```
}
```

```
/* perfectly centering */
```

```
html,
```

```
body {
```

```
    width: 100%;
```

```
height: 100%;  
}
```

```
/* Changing the color of selection */
```

```
::-moz-selection {  
    background: transparent;  
}
```

```
::selection {  
    background: transparent;  
}
```

```
/* to blacken area for stage */
```

```
body {  
    width: 100%;  
    height: 100%;  
    overflow: hidden;  
    background: #000000;  
    font-family: "Century Gothic", AppleGothic, "CenturyGothic", sans-serif;  
    display: flex;  
    justify-content: center;  
    align-items: center;  
    flex-direction: column;  
    /* color: white; */
```



```
}
```

```
/* pre-defining all the animations required */
```

```
/* walking animation */
```

```
@-webkit-keyframes walk {
```

```
  to {
```

```
    transform: translateX(-1024px);
```

```
  }
```

```
}
```

```
@keyframes walk {
```

```
  to {
```

```
    transform: translateX(-1024px);
```

```
  }
```

```
}
```

```
@-webkit-keyframes reload {
```

```
  to {
```

```
    transform: rotate(6turn);
```

```
  }
```

```
}
```

```
/* reload */
```

```
@keyframes reload {
```

```
to {  
  transform: rotate(6turn);  
}  
}
```

```
@-webkit-keyframes reload-trigger {
```

```
  0% {  
    transform: rotate(-360deg);  
  }
```

```
  20% {  
    transform: rotate(0deg);  
  }
```

```
  100% {  
    transform: rotate(0deg);  
  }
```

```
}
```

```
@keyframes reload-trigger {
```

```
  0% {  
    transform: rotate(-360deg);  
  }
```

```
  20% {
```

```
transform: rotate(0deg);  
}
```

```
100% {  
    transform: rotate(0deg);  
}  
}
```

```
@-webkit-keyframes flash {
```

```
0% {  
    opacity: 1;  
}
```

```
25% {  
    opacity: 0.5;  
}
```

```
50% {  
    opacity: 0.8;  
}
```

```
75% {  
    opacity: 0.4;  
}
```

```
100% {  
  opacity: 0.2;  
}  
}
```

```
@keyframes flash {
```

```
  0% {  
    opacity: 1;  
  }
```

```
  25% {  
    opacity: 0.5;  
  }
```

```
  50% {  
    opacity: 0.8;  
  }
```

```
  75% {  
    opacity: 0.4;  
  }
```

```
  100% {  
    opacity: 0.2;  
  }
```

```
}
```

```
/* text glitch animation */
```

```
@-webkit-keyframes dripping {
```

```
0% {
```

```
text-shadow: 6px 4px 0 red, -5px 5px 0 red, -3px 6px 0 red,
```

```
2px 11px 0 red;
```

```
}
```

```
30% {
```

```
text-shadow: 5px 3px 0 red, -4px 4px 0 red, -2px 5px 0 red,
```

```
1px 10px 0 red;
```

```
}
```

```
60% {
```

```
text-shadow: 6px 4px 0 red, -5px 5px 0 red, -3px 6px 0 red,
```

```
2px 11px 0 red;
```

```
}
```

```
90% {
```

```
text-shadow: 5px 3px 0 red, -4px 4px 0 red, -2px 5px 0 red,
```

```
1px 10px 0 red;
```

```
}
```

```
100% {
```

```
text-shadow: 6px 4px 0 red, -5px 5px 0 red, -3px 6px 0 red,  
2px 11px 0 red;  
}  
}
```

```
@keyframes dripping {
```

```
0% {  
text-shadow: 6px 4px 0 red, -5px 5px 0 red, -3px 6px 0 red,  
2px 11px 0 red;  
}
```

```
30% {  
text-shadow: 5px 3px 0 red, -4px 4px 0 red, -2px 5px 0 red,  
1px 10px 0 red;  
}
```

```
60% {  
text-shadow: 6px 4px 0 red, -5px 5px 0 red, -3px 6px 0 red,  
2px 11px 0 red;  
}
```

```
90% {  
text-shadow: 5px 3px 0 red, -4px 4px 0 red, -2px 5px 0 red,  
1px 10px 0 red;  
}
```

```
100% {  
    text-shadow: 6px 4px 0 red, -5px 5px 0 red, -3px 6px 0 red,  
        2px 11px 0 red;  
}  
}
```

```
/* setting up the canvas and cursor. */
```

```
#canves {  
    width: 1024px;  
    height: 550px;  
    min-width: 1024px;  
    min-height: 550px;  
    position: relative;  
    overflow: hidden;  
    background-color: #000000;  
    cursor: crosshair;  
    border-radius: 8px;  
    font-family: "Cinzel", serif;  
    transition: background-image 1.3s ease;  
    background-repeat: no-repeat;  
    background-size: 1024px 550px;  
}
```

```
#canves[data-wave="1"] {
```

```
background-image: url("wave1.png");  
}
```

```
#canves[data-wave="2"] {  
background-image: url("wave2.png");  
}
```

```
#canves[data-wave="3"] {  
background-image: url("wave3.png");  
}
```

```
#canves[data-wave="4"] {  
background-image: url("wave4.png");  
}
```

```
#canves .game-cover {  
position: absolute;  
top: 0;  
right: 0;  
bottom: 0;  
left: 0;  
background: url("hero--cover.png");  
background-size: 1024px 550px;  
display: block;  
z-index: 100000;
```



```
}
```

```
/* adding worn out corner look */
```

```
#canves:after {
```

```
    position: absolute;
```

```
    top: 0;
```

```
    right: 0;
```

```
    bottom: 0;
```

```
    left: 0;
```

```
    content: "";
```

```
    pointer-events: none;
```

```
    display: block;
```

```
    background: url("frame.png") no-repeat;
```

```
    background-size: cover;
```

```
    border-radius: 8px;
```

```
    z-index: 1000000;
```

```
}
```

```
/* github logo animation */
```

```
.github-corner:hover .octo-arm {
```

```
    -webkit-animation: octocat-wave 560ms ease-in-out;
```

```
    animation: octocat-wave 560ms ease-in-out;
```

```
}
```

```
@-webkit-keyframes octocat-wave {
```

```
  0%,
```

```
  100% {
```

```
    transform: rotate(0);
```

```
  }
```

```
  20%,
```

```
  60% {
```

```
    transform: rotate(-25deg);
```

```
  }
```

```
  40%,
```

```
  80% {
```

```
    transform: rotate(10deg);
```

```
  }
```

```
}
```

```
@keyframes octocat-wave {
```

```
  0%,
```

```
  100% {
```

```
    transform: rotate(0);
```

```
  }
```

```
20%,  
60% {  
  transform: rotate(-25deg);  
}
```

```
40%,  
80% {  
  transform: rotate(10deg);  
}  
}
```

```
@media (max-width: 500px) {  
  .github-corner:hover .octo-arm {  
    -webkit-animation: none;  
    animation: none;  
  }
```

```
.github-corner .octo-arm {  
  -webkit-animation: octocat-wave 560ms ease-in-out;  
  animation: octocat-wave 560ms ease-in-out;  
}  
}
```

```
.github-corner {  
  position: absolute;
```

```
top: 0;  
right: 0;  
z-index: 10000000;  
}
```

```
@-webkit-keyframes octocat-wave {
```

```
0%,  
100% {  
    transform: rotate(0);  
}
```

```
20%,  
60% {  
    transform: rotate(-25deg);  
}
```

```
40%,  
80% {  
    transform: rotate(10deg);  
}
```

```
}
```

```
@keyframes octocat-wave {
```

```
0%,  
100% {  
    transform: rotate(0);  
}
```

```
20%,  
60% {  
    transform: rotate(-25deg);  
}
```

```
40%,  
80% {  
    transform: rotate(10deg);  
}  
}
```

```
/* adding footer + heart animation */
```

```
#footer {  
    font-size: 11px;  
    color: #5a5a5a;  
    margin-top: 20px;  
    margin-bottom: 20px;  
    line-height: 21px;  
    text-align: center;  
}
```

```
#footer:hover svg {  
  fill: #ff0000;  
}
```

```
@media screen and (max-width: 625px) {  
  #footer {  
    text-align: center;  
  }  
}
```

```
#footer a {  
  color: currentColor;  
  text-decoration: none;  
}
```

```
#footer .credits {  
  color: #353535;  
}
```

```
#footer svg {  
  fill: #646464;  
  width: 15px;  
  transition: fill 0.3s ease;  
}
```

```
.get-code.generate-code #footer {  
  margin-bottom: 164px;  
}
```

```
@media screen and (max-width: 625px) {  
  .get-code.generate-code #footer {  
    margin-bottom: 208px;  
  }  
}
```

```
/* setting up the loader */
```

```
.loader {  
  position: absolute;  
  top: 0;  
  right: 0;  
  bottom: 0;  
  left: 0;  
  opacity: 0;  
  pointer-events: none;  
  z-index: 1000000;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  flex-direction: column;
```

```
transition: opacity 0.3s ease;

background: url("hero--cover.png");

background-size: 1024px 550px;

}
```

```
.loading .loader {

  opacity: 1;

}
```

```
.loader .zombie-loader {

  transform: scale(0.5);

  margin-bottom: 20px;

}
```

```
.loader .zombie-loader:before {

  content: "";

  display: block;

  -webkit-animation-duration: 0.7s !important;

  animation-duration: 0.7s !important;

}
```

```
.loader span {

  -webkit-animation: flash 1.8s ease infinite;

  animation: flash 1.8s ease infinite;

  color: #ffffff;
```



```
font-size: 9px;  
position: absolute;  
bottom: 62px;  
right: 72px;  
}
```

```
loader .loading-char {  
  background: radial-gradient(ellipse at center,  
    black 0%,  
    rgba(246, 246, 246, 0) 50%);  
  position: absolute;  
  bottom: -20px;  
  right: 30px;  
}
```

```
.info-board {  
  display: block;  
  position: absolute;  
  top: 35px;  
  left: 35px;  
  font-size: 25px;  
  color: #ffffff;  
  z-index: 10000;  
  padding: 0;  
}
```

```
.intro .info-board,  
.end-game .info-board,  
.game-over .info-board,  
.level-message .info-board {  
    display: none;  
}
```

```
.info-board li {  
    list-style: none;  
    margin-bottom: 16px;  
}
```

```
.info-board li.killed-status {  
    background: url("icon-pack.png") no-repeat;  
    background-position: 0 0;  
    background-size: 114px;  
    width: 114px;  
    height: 39.5px;  
    padding: 0 0 10px 51px;  
    font-size: 32px;  
    line-height: 41px;  
}
```

```
.info-board li.life {
```

```
margin-top: 20px;  
margin-bottom: 9px;  
}
```

```
.info-board li.life .heart-icon {  
    background: url("icon-pack.png") no-repeat;  
    background-position: 0 -47px;  
    background-size: 114px;  
    width: 24px;  
    height: 23px;  
    display: inline-block;  
    margin-right: 15px;  
    opacity: 1;  
    transition: opacity 0.3s ease;  
}
```

```
.info-board li.life .heart-icon.hide {  
    -webkit-animation: flash 0.6s ease;  
    animation: flash 0.6s ease;  
    opacity: 0.2;  
}
```

```
.info-board li.life .heart-icon:last-child {  
    margin-right: 0;  
}
```

```
/* buttons */

.info-board li#mute-music {

    background: url("icon-pack.png") no-repeat;

    background-position: 0 -75px;

    background-size: 114px;

    width: 25px;

    height: 25px;

    cursor: pointer;

    display: inline-block;

    margin-right: 6px;

}

.info-board li#mute-music.muted {

    background-position: 0 -105px;

}

.info-board li#mute-sounds {

    background: url("icon-pack.png") no-repeat;

    background-position: -30px -73px;

    background-size: 114px;

    width: 22px;

    height: 25px;

    cursor: pointer;

    display: inline-block;
```

```
margin-right: 6px;  
}
```

```
.info-board li#mute-sounds.muted {  
    background-position: -30px -104px;  
}
```

```
#pause-game {  
    background: url("icon-pack.png") no-repeat;  
    background-position: -60px -106px;  
    background-size: 114px;  
    width: 19px;  
    height: 22px;  
    cursor: pointer;  
    position: absolute;  
    top: 35px;  
    right: 35px;  
    z-index: 100000;  
}
```

```
#pause-game.paused {  
    background-position: -60px -75px;  
}
```

```
.intro #pause-game,
```

```
.end-game #pause-game,  
.game-over #pause-game,  
.level-message #pause-game {  
    display: none;  
}
```

```
.game-paused #pause-game {  
    z-index: 100000;  
}
```

```
.ammo {  
    background: url("icon-pack.png") no-repeat;  
    background-position: 0 -134px;  
    background-size: 114px;  
    width: 70px;  
    height: 74.5px;  
    transition: transform 0.3s ease;  
    position: absolute;  
    bottom: 35px;  
    right: 35px;  
    z-index: 10000;  
    display: block;  
}
```

```
.intro .ammo,
```

```
.end-game .ammo,  
.game-over .ammo,  
.level-message .ammo {  
    display: none;  
}
```

```
.ammo[data-ammo="0"] {  
    background-position: 0px -581px;  
    transform: rotate(360deg);  
}
```

```
.ammo[data-ammo="1"] {  
    background-position: 0px -507px;  
    transform: rotate(300deg);  
}
```

```
.ammo[data-ammo="2"] {  
    background-position: 0px -432px;  
    transform: rotate(240deg);  
}
```

```
.ammo[data-ammo="3"] {  
    background-position: 0px -358px;  
    transform: rotate(180deg);  
}
```

```
.ammo[data-ammo="4"] {  
    background-position: 0 -283px;  
    transform: rotate(120deg);  
}
```

```
.ammo[data-ammo="5"] {  
    background-position: 0 -208px;  
    transform: rotate(60deg);  
}
```

```
.ammo[data-ammo="6"] {  
    background-position: 0 -134px;  
}
```

```
.ammo.reload {  
    -webkit-animation: reload 0.6s ease;  
    animation: reload 0.6s ease;  
}
```

```
.reload-key {  
    position: absolute;  
    top: 0;  
    right: 0;  
    bottom: 0;
```



```
left: 0;

pointer-events: none;

display: flex;

justify-content: center;

align-items: center;

font-size: 9px;

color: #878787;

transform: translateX(0);

transition: transform 0.3s cubic-bezier(0.6, 1.14, 0.04, 1.02),
    opacity 0.3s cubic-bezier(0.6, 1.14, 0.04, 1.02);

z-index: 100000;
}
```

```
[data-ammo="6"] .reload-key,
.intro .reload-key,
.end-game .reload-key,
.game-over .reload-key,
.level-message .reload-key {
    display: none;
}
```

```
.reload-hint {
    position: absolute;
    bottom: 55px;
    right: 25px;
```

```
transform: translateX(100%);  
transition: transform 0.3s cubic-bezier(0.6, 1.14, 0.04, 1.02),  
    opacity 0.3s cubic-bezier(0.6, 1.14, 0.04, 1.02);  
padding: 0 28px;  
color: #ffffff;  
opacity: 0;  
z-index: 1000000;  
display: block;  
}
```

```
.reload-hint.visible {  
    transform: translateX(0);  
    opacity: 1;  
}
```

```
.reload-hint.visible .reload-trigger {  
    -webkit-animation: reload-trigger 2s ease infinite;  
    animation: reload-trigger 2s ease infinite;  
}
```

```
.intro .reload-hint,  
.end-game .reload-hint,  
.game-over .reload-hint,  
.level-message .reload-hint {  
    display: none;
```

```
}
```

```
.reload-trigger {  
  cursor: pointer;  
  background: url("icon-pack.png") no-repeat;  
  background-size: 114px;  
  background-position: bottom right;  
  width: 35px;  
  height: 35px;  
  transition: transform 0.3s ease;  
}
```

```
.overlay-screen-levels,  
.overlay-screen {  
  position: absolute;  
  top: 0;  
  right: 0;  
  bottom: 0;  
  left: 0;  
  background: rgba(0, 0, 0, 0.8);  
  visibility: hidden;  
  transition: visibility 0.3s ease, opacity 0.3s ease;  
  z-index: 100000;  
  display: flex;  
  justify-content: center;
```

```
align-items: center;

opacity: 0;

}
```

```
.end-game .overlay-screen-levels,
.game-over .overlay-screen-levels,
.level-message .overlay-screen-levels,
.game-paused .overlay-screen-levels,
.end-game .overlay-screen,
.game-over .overlay-screen,
.level-message .overlay-screen,
.game-paused .overlay-screen {
  visibility: visible;
  opacity: 1;
}
```

```
.big-title {
  top: 115px;
  color: #ffffff;
  text-align: center;
  font-size: 128px;
  margin: 0;
  text-shadow: 5px 3px 0 red, -4px 4px 0 red, -2px 5px 0 red,
    1px 10px 0 red;
  transition: opacity 0.3s ease, visibility 0.3s ease;
```

```
-webkit-animation: dripping 0.2s ease infinite;  
animation: dripping 0.2s ease infinite;  
cursor: default;  
display: none;  
}
```

```
.big-title:hover {  
  -webkit-animation: flash 0.3s ease infinite, dripping 0.2s ease infinite;  
  animation: flash 0.3s ease infinite, dripping 0.2s ease infinite;  
}
```

```
.big-title div {  
  text-transform: uppercase;  
  font-size: 99px;  
  letter-spacing: 20px;  
  display: block;  
  margin-top: -73px;  
  margin-left: 31px;  
}
```

```
.game-over .big-title.game-over-title {  
  display: block;  
}
```

```
.big-title.game-over-title span {
```

```
margin-left: 18px;  
}
```

```
.end-game .big-title.end-game-title {  
    display: block;  
}
```

```
.big-title.end-game-title span {  
    margin-left: 19px;  
}
```

```
.game-paused .big-title.game-pause-title {  
    display: block;  
}
```

```
.level-message .big-title.level-title {  
    display: block;  
}
```

```
.restart-hint {  
    -webkit-animation: flash 1.8s ease infinite;  
    animation: flash 1.8s ease infinite;  
    color: #838383;  
    font-size: 23px;  
    position: absolute;
```

```
bottom: 109px;  
display: none;  
cursor: pointer;
```

```
}
```

```
.end-game .restart-hint,  
.game-over .restart-hint {  
  display: block;  
}
```

```
.zombie {  
  position: absolute;  
  bottom: 74px;  
  right: 0;  
  transform: translateX(400px);  
  -webkit-animation: walk linear infinite;  
  animation: walk linear infinite;  
}
```

```
.zombie:before {  
  content: "";  
  display: block;  
}
```

```
.zombie.killed {  
  pointer-events: none;  
  transition: none;  
  -webkit-animation-play-state: paused;  
  animation-play-state: paused;  
}
```

```
.zombie.killed:before {  
  pointer-events: none;  
  -webkit-animation-fill-mode: forwards;  
  animation-fill-mode: forwards;  
}
```

```
.game-paused .zombie {  
  pointer-events: none;  
  transition: none;  
  -webkit-animation-play-state: paused;  
  animation-play-state: paused;  
}
```

```
.game-paused .zombie:before {  
  pointer-events: none;  
  -webkit-animation-play-state: paused;  
  animation-play-state: paused;  
}
```



```
.zombie-1:before {  
    background-image: url("zombie-1.png");  
    background-repeat: no-repeat;  
    -webkit-animation: zombie-1 steps(16);  
    animation: zombie-1 steps(16);  
    -webkit-animation-duration: 0.6s;  
    animation-duration: 0.6s;  
    -webkit-animation-iteration-count: infinite;  
    animation-iteration-count: infinite;  
    -webkit-animation-direction: normal;  
    animation-direction: normal;  
    width: 194px;  
    height: 255px;  
    background-position: 0 top;  
}
```

```
@-webkit-keyframes zombie-1 {  
    to {  
        background-position: 0 -4080px;  
    }  
}
```

```
@keyframes zombie-1 {  
    to {
```

```
background-position: 0 -4080px;

}

}
```

```
.zombie-1.killed:before {

background-image: url("zombie-1-death.png");

background-repeat: no-repeat;

-webkit-animation: zombie-1-death steps(20);

animation: zombie-1-death steps(20);

-webkit-animation-duration: 0.6s;

animation-duration: 0.6s;

-webkit-animation-iteration-count: infinite;

animation-iteration-count: infinite;

-webkit-animation-direction: normal;

animation-direction: normal;

width: 340px;

height: 261px;

background-position: 0 top;

transform: translate(158px, 22px);

}
```

```
@-webkit-keyframes zombie-1-death {

to {

background-position: 0 -5220px;

}
```

```
}
```

```
@keyframes zombie-1-death {  
  to {  
    background-position: 0 -5220px;  
  }  
}
```

```
.zombie-2:before {  
  background-image: url("zombie-2.png");  
  background-repeat: no-repeat;  
  -webkit-animation: zombie-2 steps(16);  
  animation: zombie-2 steps(16);  
  -webkit-animation-duration: 0.6s;  
  animation-duration: 0.6s;  
  -webkit-animation-iteration-count: infinite;  
  animation-iteration-count: infinite;  
  -webkit-animation-direction: normal;  
  animation-direction: normal;  
  width: 211px;  
  height: 256px;  
  background-position: 0 top;  
}
```

```
@-webkit-keyframes zombie-2 {
```

```
to {  
    background-position: 0 -4096px;  
}  
}
```

```
@keyframes zombie-2 {  
    to {  
        background-position: 0 -4096px;  
    }  
}
```

```
.zombie-2.killed:before {  
    background-image: url("zombie-2-death.png");  
    background-repeat: no-repeat;  
    -webkit-animation: zombie-2-death steps(20);  
    animation: zombie-2-death steps(20);  
    -webkit-animation-duration: 0.6s;  
    animation-duration: 0.6s;  
    -webkit-animation-iteration-count: infinite;  
    animation-iteration-count: infinite;  
    -webkit-animation-direction: normal;  
    animation-direction: normal;  
    width: 347px;  
    height: 270px;  
    background-position: 0 top;
```

```
transform: translate(159px, 31px);  
}
```

```
@-webkit-keyframes zombie-2-death {  
  to {  
    background-position: 0 -5400px;  
  }  
}
```

```
@keyframes zombie-2-death {  
  to {  
    background-position: 0 -5400px;  
  }  
}
```

```
.zombie-3:before {  
  background-image: url("zombie-3.png");  
  background-repeat: no-repeat;  
  -webkit-animation: zombie-3 steps(16);  
  animation: zombie-3 steps(16);  
  -webkit-animation-duration: 0.6s;  
  animation-duration: 0.6s;  
  -webkit-animation-iteration-count: infinite;  
  animation-iteration-count: infinite;  
  -webkit-animation-direction: normal;
```

```
animation-direction: normal;

width: 189px;

height: 234px;

background-position: 0 top;

}
```

```
@-webkit-keyframes zombie-3 {

  to {

    background-position: 0 -3744px;

  }

}
```

```
@keyframes zombie-3 {

  to {

    background-position: 0 -3744px;

  }

}
```

```
.zombie-3.killed:before {

  background-image: url("zombie-3-death.png");

  background-repeat: no-repeat;

  -webkit-animation: zombie-3-death steps(20);

  animation: zombie-3-death steps(20);

  -webkit-animation-duration: 0.6s;

  animation-duration: 0.6s;

}
```

```
-webkit-animation-iteration-count: infinite;

animation-iteration-count: infinite;

-webkit-animation-direction: normal;

animation-direction: normal;

width: 334px;

height: 256px;

background-position: 0 top;

transform: translate(156px, 25px);

}
```

```
@-webkit-keyframes zombie-3-death {

  to {

    background-position: 0 -5120px;

  }

}
```

```
@keyframes zombie-3-death {

  to {

    background-position: 0 -5120px;

  }

}
```

```
.zombie-4:before {

  background-image: url("zombie-4.png");

  background-repeat: no-repeat;
```

```
-webkit-animation: zombie-4 steps(16);  
animation: zombie-4 steps(16);  
-webkit-animation-duration: 0.6s;  
animation-duration: 0.6s;  
-webkit-animation-iteration-count: infinite;  
animation-iteration-count: infinite;  
-webkit-animation-direction: normal;  
animation-direction: normal;  
width: 386px;  
height: 375px;  
background-position: 0 top;  
}
```

```
@-webkit-keyframes zombie-4 {  
  to {  
    background-position: 0 -6000px;  
  }  
}
```

```
@keyframes zombie-4 {  
  to {  
    background-position: 0 -6000px;  
  }  
}
```



```
.zombie-4.killed:before {  
    background-image: url("zombie-4-death.png");  
    background-repeat: no-repeat;  
    -webkit-animation: zombie-4-death steps(16);  
    animation: zombie-4-death steps(16);  
    -webkit-animation-duration: 0.6s;  
    animation-duration: 0.6s;  
    -webkit-animation-iteration-count: infinite;  
    animation-iteration-count: infinite;  
    -webkit-animation-direction: normal;  
    animation-direction: normal;  
    width: 497px;  
    height: 401px;  
    background-position: 0 top;  
    transform: translate(65px, 45px);  
}
```

```
@-webkit-keyframes zombie-4-death {  
    to {  
        background-position: 0 -6416px;  
    }  
}
```

```
@keyframes zombie-4-death {  
    to {
```

```
background-position: 0 -6416px;

}

}
```

```
.zombie-5:before {

background-image: url("zombie-5.png");

background-repeat: no-repeat;

-webkit-animation: zombie-5 steps(16);

animation: zombie-5 steps(16);

-webkit-animation-duration: 0.6s;

animation-duration: 0.6s;

-webkit-animation-iteration-count: infinite;

animation-iteration-count: infinite;

-webkit-animation-direction: normal;

animation-direction: normal;

width: 386px;

height: 376px;

background-position: 0 top;

}
```

```
@-webkit-keyframes zombie-5 {

to {

background-position: 0 -6016px;

}

}
```

```
@keyframes zombie-5 {  
  to {  
    background-position: 0 -6016px;  
  }  
}
```

```
.zombie-5.killed:before {  
  background-image: url("zombie-5-death.png");  
  background-repeat: no-repeat;  
  -webkit-animation: zombie-5-death steps(16);  
  animation: zombie-5-death steps(16);  
  -webkit-animation-duration: 0.6s;  
  animation-duration: 0.6s;  
  -webkit-animation-iteration-count: infinite;  
  animation-iteration-count: infinite;  
  -webkit-animation-direction: normal;  
  animation-direction: normal;  
  width: 488px;  
  height: 390px;  
  background-position: 0 top;  
  transform: translate(70px, 37px);  
}
```

```
@-webkit-keyframes zombie-5-death {
```

```
to {  
    background-position: 0 -6240px;  
}  
}
```

```
@keyframes zombie-5-death {  
    to {  
        background-position: 0 -6240px;  
    }  
}
```

```
.zombie-6:before {  
    background-image: url("zombie-6.png");  
    background-repeat: no-repeat;  
    -webkit-animation: zombie-6 steps(16);  
    animation: zombie-6 steps(16);  
    -webkit-animation-duration: 0.6s;  
    animation-duration: 0.6s;  
    -webkit-animation-iteration-count: infinite;  
    animation-iteration-count: infinite;  
    -webkit-animation-direction: normal;  
    animation-direction: normal;  
    width: 386px;  
    height: 404px;  
    background-position: 0 top;
```

```
}
```

```
@-webkit-keyframes zombie-6 {  
  to {  
    background-position: 0 -6464px;  
  }  
}
```

```
@keyframes zombie-6 {  
  to {  
    background-position: 0 -6464px;  
  }  
}
```

```
.zombie-6.killed:before {  
  background-image: url("zombie-6-death.png");  
  background-repeat: no-repeat;  
  -webkit-animation: zombie-6-death steps(16);  
  animation: zombie-6-death steps(16);  
  -webkit-animation-duration: 0.6s;  
  animation-duration: 0.6s;  
  -webkit-animation-iteration-count: infinite;  
  animation-iteration-count: infinite;  
  -webkit-animation-direction: normal;  
  animation-direction: normal;
```

```
width: 510px;

height: 419px;

background-position: 0 top;

transform: translate(75px, 44px);

}
```

```
@-webkit-keyframes zombie-6-death {

  to {

    background-position: 0 -6704px;

  }

}
```

```
@keyframes zombie-6-death {

  to {

    background-position: 0 -6704px;

  }

}
```

```
/* health bar's */

.strength-bar {

  position: absolute;

  top: -15px;

  left: 50%;

  transform: translateX(-50%);

  border: 1px solid #ffffff;
```

```
height: 5px;  
border-radius: 14px;  
opacity: 1;  
transition: opacity 0.3s ease;  
}
```

```
.strength-bar.hide {  
  opacity: 0;  
}
```

```
.zombie-1 .strength-bar {  
  width: 15px;  
}
```

```
.zombie-2 .strength-bar {  
  width: 30px;  
}
```

```
.zombie-3 .strength-bar {  
  width: 45px;  
}
```

```
.zombie-4 .strength-bar {  
  width: 60px;  
}
```

```
.zombie-5 .strength-bar {  
  width: 75px;  
}
```

```
.zombie-6 .strength-bar {  
  width: 90px;  
}
```

```
.strength-bar:after {  
  position: absolute;  
  top: 0;  
  right: 0;  
  bottom: 0;  
  left: 0;  
  content: "";  
  background: red;  
  transition: width 0.3s ease;  
  display: block;  
  width: 0;  
}
```

```
.zombie-1[data-strength="0"] .strength-bar:after {  
  width: 0;  
}
```



```
.zombie-1[data-strength="1"] .strength-bar:after {  
  width: 100%;  
}
```

```
.zombie-2[data-strength="0"] .strength-bar:after {  
  width: 0;  
}
```

```
.zombie-2[data-strength="1"] .strength-bar:after {  
  width: 50%;  
}
```

```
.zombie-2[data-strength="2"] .strength-bar:after {  
  width: 100%;  
}
```

```
.zombie-3[data-strength="0"] .strength-bar:after {  
  width: 0;  
}
```

```
.zombie-3[data-strength="1"] .strength-bar:after {  
  width: 33%;  
}
```

```
.zombie-3[data-strength="2"] .strength-bar:after {  
  width: 66%;  
}
```

```
.zombie-3[data-strength="3"] .strength-bar:after {  
  width: 100%;  
}
```

```
.zombie-4[data-strength="0"] .strength-bar:after {  
  width: 0%;  
}
```

```
.zombie-4[data-strength="1"] .strength-bar:after {  
  width: 25%;  
}
```

```
.zombie-4[data-strength="2"] .strength-bar:after {  
  width: 50%;  
}
```

```
.zombie-4[data-strength="3"] .strength-bar:after {  
  width: 75%;  
}
```

```
.zombie-4[data-strength="4"] .strength-bar:after {
```

```
width: 100%;  
}
```

```
.zombie-5[data-strength="0"] .strength-bar:after {  
width: 0;  
}
```

```
.zombie-5[data-strength="1"] .strength-bar:after {  
width: 20%;  
}
```

```
.zombie-5[data-strength="2"] .strength-bar:after {  
width: 40%;  
}
```

```
.zombie-5[data-strength="3"] .strength-bar:after {  
width: 60%;  
}
```

```
.zombie-5[data-strength="4"] .strength-bar:after {  
width: 80%;  
}
```

```
.zombie-5[data-strength="5"] .strength-bar:after {  
width: 100%;
```

```
}
```

```
.zombie-6[data-strength="0"] .strength-bar:after {  
  width: 0;  
}
```

```
.zombie-6[data-strength="1"] .strength-bar:after {  
  width: 25%;  
}
```

```
.zombie-6[data-strength="2"] .strength-bar:after {  
  width: 40%;  
}
```

```
.zombie-6[data-strength="3"] .strength-bar:after {  
  width: 55%;  
}
```

```
.zombie-6[data-strength="4"] .strength-bar:after {  
  width: 70%;  
}
```

```
.zombie-6[data-strength="5"] .strength-bar:after {  
  width: 85%;  
}
```

```
.zombie-6[data-strength="6"] .strength-bar:after {  
    width: 100%;  
}
```

```
/* waling speeds */
```

```
.walk-speed-1 {  
    -webkit-animation-duration: 11s;  
    animation-duration: 11s;  
}
```

```
.walk-speed-1:before {  
    -webkit-animation-duration: 0.5s;  
    animation-duration: 0.5s;  
}
```

```
.walk-speed-2 {  
    -webkit-animation-duration: 12s;  
    animation-duration: 12s;  
}
```

```
.walk-speed-2:before {  
    -webkit-animation-duration: 1s;  
    animation-duration: 1s;  
}
```

```
.walk-speed-3 {  
  -webkit-animation-duration: 13s;  
  animation-duration: 13s;  
}
```

```
.walk-speed-3:before {  
  -webkit-animation-duration: 1.5s;  
  animation-duration: 1.5s;  
}
```

```
.walk-speed-4 {  
  -webkit-animation-duration: 14s;  
  animation-duration: 14s;  
}
```

```
.walk-speed-4:before {  
  -webkit-animation-duration: 2s;  
  animation-duration: 2s;  
}
```

```
.walk-speed-5 {  
  -webkit-animation-duration: 15s;  
  animation-duration: 15s;  
}
```

```
.walk-speed-5:before {  
  -webkit-animation-duration: 2.5s;  
  animation-duration: 2.5s;  
}
```

```
.walk-speed-6 {  
  -webkit-animation-duration: 16s;  
  animation-duration: 16s;  
}
```

```
.walk-speed-6:before {  
  -webkit-animation-duration: 3s;  
  animation-duration: 3s;  
}
```

```
.walk-delay-1 {  
  -webkit-animation-delay: 0.6666666667s;  
  animation-delay: 0.6666666667s;  
}
```

```
.walk-delay-2 {  
  -webkit-animation-delay: 1.3333333333s;  
  animation-delay: 1.3333333333s;  
}
```

```
.walk-delay-3 {
```

```
  -webkit-animation-delay: 2s;
```

```
  animation-delay: 2s;
```

```
}
```

```
.walk-delay-4 {
```

```
  -webkit-animation-delay: 2.6666666667s;
```

```
  animation-delay: 2.6666666667s;
```

```
}
```

```
.walk-delay-5 {
```

```
  -webkit-animation-delay: 3.3333333333s;
```

```
  animation-delay: 3.3333333333s;
```

```
}
```

```
.walk-delay-6 {
```

```
  -webkit-animation-delay: 4s;
```

```
  animation-delay: 4s;
```

```
}
```

```
</style>
```

```
<script>
```

```
  window.console = window.console || function (t) {};
```

```
</script>
```



```

<script>

  if (document.location.search.match(/type=embed/gi)) {

    window.parent.postMessage("resize", "*");

  }

</script>

</head>

<body translate="no">

  <a href="https://github.com/KodeMeister-YT" class="github-corner" target="_blank"><svg width="80"
height="80"

  viewBox="0 0 250 250" style="

    fill: #171717;

    color: #fff;

    position: absolute;

    top: 0;

    border: 0;

    right: 0;

  ">

  <!-- github cat corner button shape -->

  <path d="M0,0 L115,115 L130,115 L142,142 L250,250 L250,0 Z"></path>

  <path

    d="M128.3,109.0 C113.8,99.7 119.0,89.6 119.0,89.6 C122.0,82.7 120.5,78.6 120.5,78.6 C119.2,72.0
123.4,76.3 123.4,76.3 C127.3,80.9 125.5,87.3 125.5,87.3 C122.9,97.6 130.6,101.9 134.4,103.2"

    fill="currentColor" style="transform-origin: 130px 106px" class="octo-arm"></path>

  <path

```

```
d="M115.0,115.0 C114.9,115.1 118.7,116.5 119.8,115.4 L133.7,101.6 C136.9,99.2 139.9,98.4
142.2,98.6 C133.8,88.0 127.5,74.4 143.8,58.0 C148.5,53.4 154.0,51.2 159.7,51.0 C160.3,49.4 163.2,43.6
171.4,40.1 C171.4,40.1 176.1,42.5 178.8,56.2 C183.1,58.6 187.2,61.8 190.9,65.4 C194.5,69.0 197.7,73.2
200.1,77.6 C213.8,80.2 216.3,84.9 216.3,84.9 C212.7,93.1 206.9,96.0 205.4,96.6 C205.1,102.4
203.0,107.8 198.3,112.5 C181.9,128.9 168.3,122.5 157.7,114.1 C157.9,116.9 156.7,120.9 152.7,124.9
L141.0,136.5 C139.8,137.7 141.6,141.9 141.8,141.8 Z"
```

```
fill="currentColor" class="octo-body"></path>
```

```
</svg></a>
```

```
<div id="canves" data-wave="" class="loading intro">
```

```
<div class="loader">
```

```
<div class="loading-char">
```

```
<div class="zombie-loader"></div>
```

```
<span>Loading</span>
```

```
</div>
```

```
</div>
```

```
<div class="game-cover"></div>
```

```
<div class="overlay-screen">
```

```
<h2 class="big-title game-over-title">
```

```
Game
```

```
<div>Over</div>
```

```
</h2>
```

```
<h2 class="big-title end-game-title">
```

```
You
```

```
<div>Won</div>
```

```
</h2>
```

```
<span class="restart-hint">Click Here to Restart</span>
```

```
<h2 class="big-title game-pause-title">Paused</h2>
```

```

<h2 class="big-title level-title">Level <span></span></h2>

</div>

<ul class="info-board">

  <li class="killed-status"><span>0</span></li>

  <li class="life">

    <i class="heart-icon"></i><i class="heart-icon"></i><i class="heart-icon"></i>

  </li>

  <li id="mute-music" data-tootik="Mute Music" data-tootik-conf="right invert"></li>

  <li id="mute-sounds" data-tootik="Mute Sounds" data-tootik-conf="right invert"></li>

</ul>

<div id="pause-game"></div>

<div class="ammo" data-ammo="6"><span class="reload-key">R</span></div>

<div class="reload-hint">

  <div class="reload-trigger"></div>

</div>

</div>

<div id="footer">

  <p>

    Handcrafted by RAJAT ROHILLA

    <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 492.7 492.7" width="8" height="8">

      <!-- heart -->

      <path

        d="M492.7 166c0-73.5-59.6-133.1-133.1-133.1-48 0-89.9 25.5-113.3 63.6 -23.4-38.1-65.3-63.6-
        113.3-63.6C59.6 33 0 92.5 0 166c0 40 17.7 75.8 45.7 100.2l188.5 188.6c3.2 3.2 7.6 5 12.1 5 4.6 0 8.9-1.8
        12.1-5l188.5-188.6C475 241.8 492.7 206 492.7 166z" />

```

```
</svg>
```

```
</p>
```

```
<p></p>
```

```
<div class="credits">
```

```
<p>Some other Contributions:</p>
```

```
TARUN KUMAR
```

```
</div>
```

```
</div>
```

```
<script src="2.js"></script>
```

```
<script src="3.js"></script>
```

```
<script id="rendered-js">
```

```
"use strict";
```

```
$(document).ready(function () {
```

```
  const ZOMBIEMAYEM = (function () {
```

```
    // Cache te DOM
```

```
    let $canves = $("#canves"),
```

```
    $overlayScreen = $canves.find(".overlay-screen"),
```

```
    $gameCover = $canves.find(".game-cover"),
```

```
    $killedTitle = $canves.find(".killed-status span"),
```

```
    $lifeIcons = $canves.find(".life"),
```

```
    $muteMusic = $canves.find("#mute-music"),
```

```
    $muteSounds = $canves.find("#mute-sounds"),
```

```
    $ammoTitle = $canves.find(".ammo"),
```

```
$reloadHint = $canves.find(".reload-hint"),
$reloadHintSpinner = $reloadHint.find(".reload-trigger"),
$pauseGameTrigger = $canves.find("#pause-game");

// Sounds
const SHOOT_SOUND = "SHOOT_SOUND",
NO_AMMO_SOUND = "NO_AMMO_SOUND",
RELOAD_SOUND = "RELOAD_SOUND",
ROAR_3 = "ROAR_3",
ROAR_4 = "ROAR_4",
ROAR_5 = "ROAR_5",
ROAR_6 = "ROAR_6",
LAUGHTER = "LAUGHTER",
SOUNDTRACK = "SOUNDTRACK",
PUNCH_1 = "PUNCH_1",
PUNCH_2 = "PUNCH_2",
PUNCH_3 = "PUNCH_3",
PUNCH_4 = "PUNCH_4";

// General
let pauseZombieTracking;

// Game Info
let mutedMusic = false,
mutedSounds = false,
gamePaused = false,
```

```
life = 3,  
  
zombieKilled = 0,  
  
wave = 0,  
  
ammo = 6;  
  
// Random number  
  
const getRandom = function (min, max) {  
    return Math.floor(Math.random() * (max - min + 1)) + min;  
};  
  
// Zombie Frequency per Level  
  
const WAVE_1_ZOMBIE_FRQ = getRandom(1300, 1700),  
    WAVE_2_ZOMBIE_FRQ = getRandom(1500, 1900),  
    WAVE_3_ZOMBIE_FRQ = getRandom(1700, 2100),  
    WAVE_4_ZOMBIE_FRQ = getRandom(1800, 2400);  
  
// Zombie Quantity per Level  
  
const WAVE_1_ZOMBIE_QTY = getRandom(5, 7),  
    WAVE_2_ZOMBIE_QTY = getRandom(7, 10),  
    WAVE_3_ZOMBIE_QTY = getRandom(10, 14),  
    WAVE_4_ZOMBIE_QTY = getRandom(14, 17),  
    ALL_ZOMBIES =  
    WAVE_1_ZOMBIE_QTY +  
    WAVE_2_ZOMBIE_QTY +  
    WAVE_3_ZOMBIE_QTY +  
    WAVE_4_ZOMBIE_QTY;
```

```
// Load sounds

(function loadSound() {

    createjs.Sound.registerSound("shoot_1.mp3", SHOOT_SOUND);

    createjs.Sound.registerSound("noammo.mp3", NO_AMMO_SOUND);

    createjs.Sound.registerSound("reload.mp3", RELOAD_SOUND);

    createjs.Sound.registerSound("3.mp3", ROAR_3);

    createjs.Sound.registerSound("4.mp3", ROAR_4);

    createjs.Sound.registerSound("5.mp3", ROAR_5);

    createjs.Sound.registerSound("6.mp3", ROAR_6);

    createjs.Sound.registerSound("laughter.mp3", LAUGHTER);

    createjs.Sound.registerSound("soundtrack.mp3", SOUNDTRACK);

    createjs.Sound.registerSound("punch_2.mp3", PUNCH_2);

    createjs.Sound.registerSound("punch_3.mp3", PUNCH_3);

    createjs.Sound.registerSound("punch_4.mp3", PUNCH_4);

})();

const playSound = function (sound) {

    createjs.Sound.play(sound);

};

// Create Zombie

const createZombies = function () {

    let zombieType =

        wave === 1 ?

        getRandom(1, 3) :

        wave === 2 ?
```

```
getRandom(1, 4) :
getRandom(1, 6);
$canves.append(
$(
'<div class="zombie zombie-' +
zombieType +
" walk-speed-" +
getRandom(1, 6) +
" walk-delay-" +
getRandom(1, 6) +
"" data-strength="" +
zombieType +
""><div class="strength-bar"></div></div>'
)
);
};
```

```
const setHandlers = function () {
// Shoot Handler
$canves.on("click", function () {
if (ammo > 0) {
ammo--;
if (!mutedSounds) {
playSound(SHOOT_SOUND);
}
}
```



```
$ammoTitle.attr("data-ammo", ammo);  
  
}  
  
if (ammo === 0) {  
  
    $reloadHint.addClass("visible");  
  
    if (!mutedSounds) {  
  
        playSound(NO_AMMO_SOUND);  
  
    }  
  
}  
  
});  
  
// Reload Handler  
$("body").on("keydown", function (e) {  
  
    if (e.which === 82) {  
  
        if (ammo !== 6) {  
  
            reload();  
  
        }  
  
    }  
  
});  
  
$reloadHintSpinner.on("click", function () {  
  
    if (ammo === 0) {  
  
        reload();  
  
        return false;  
  
    }  
  
});  
  
// Zombie Kill Handler  
  
$canves.delegate(".zombie", "click", function (e) {
```

```
e.stopPropagation;  
  
if (zombieKilled >= ALL_ZOMBIES) {  
    return false;  
}
```

```
if (ammo === 0) {  
    if (!mutedSounds) {  
        playSound(NO_AMMO_SOUND);  
    }  
    return false;  
}
```

```
let $this = $(this),  
    strength = e.target.dataset.strength.toString(),  
    $strengthBar = $this.find(".strength-bar");
```

```
if (strength === "1" && strength !== 0) {  
    zombieKilled++;  
    $killedTitle.html(zombieKilled);  
    $this.css("pointer-events", "none");  
    $strengthBar.addClass("hide");
```

```
setTimeout(function () {  
    $this.addClass("killed");  
    if (!mutedSounds) {
```

```
        playSound("ROAR_" + getRandom(3, 6));
    }
}, 220);

setTimeout(function () {
    $this.fadeOut(function () {
        $this.remove();
    });
}, 370);

calcWave();
}

strength--;
$this.attr("data-strength", strength);
});
};

const interval = function (func, wait, times) {
    const interv = (function (w, t) {
        return function () {
            if (typeof t === "undefined" || t-- > 0) {
                setTimeout(interv, w);

                try {
                    func.call(null);
                } catch (e) {
                    t = 0;
                }
            }
        };
    })(wait, times);
    return interv;
};
```

```
        throw e.toString();
    }
}
};
})(wait, times);
```

```
setTimeout(interv, wait);
};
// Start Waves
const startWave = function (frequency, quantity) {
    $("body").off("keydown");
    $canves.off("click");
    $reloadHint.removeClass("visible");
    pauseZombieTracking = true;
    wave++;

    $overlayScreen.find(".level-title span").html(wave);
    $canves.addClass("level-message");
    $canves.attr("data-wave", wave);

    setTimeout(function () {
        interval(
            function () {
                createZombies();
            },
```

```
frequency,  
quantity  
);  
  
}, 1000);  
  
setTimeout(function () {  
    $canves.removeClass("level-message");  
    if (wave === 1) {  
        $canves.removeClass("intro");  
    }  
  
    // Start Track Zombies  
    pauseZombieTracking = false;  
    trackZombies();  
    setHandlers();  
}, 2200);  
};  
  
// End Game  
const endGame = function (endType) {  
    $("body").off("keydown");  
    $canves.off("click");  
    pauseZombieTracking = true;  
    let screenType = endType === "lose" ? "game-over" : "end-game";
```

```
$canves.addClass(screenType);

if (!mutedMusic) {
    playSound(LAUGHTER);
}

$canves.find(".restart-hint").on("click", function () {

    $canves.removeClass(screenType);

    // reset game
    resetGame();

    if (endType === "win") {
        startGame();
    } else {
        if (!mutedMusic) {
            playSound(SOUNDTRACK);
        }
        startGame();
    }
});

// Calc Wave
const calcWave = function () {

    if (zombieKilled === WAVE_1_ZOMBIE_QTY) {

        // Start Wave 2

        if (!mutedMusic) {
            playSound(SOUNDTRACK);
        }
    }
}
```

```
}

startWave(WAVE_2_ZOMBIE_FRQ, WAVE_2_ZOMBIE_QTY);

} else if (zombieKilled === WAVE_1_ZOMBIE_QTY + WAVE_2_ZOMBIE_QTY) {

    // Start Wave 3

    if (!mutedMusic) {

        playSound(SOUNDTRACK);

    }

    startWave(WAVE_3_ZOMBIE_FRQ, WAVE_3_ZOMBIE_QTY);

} else if (

    zombieKilled ===

    WAVE_1_ZOMBIE_QTY + WAVE_2_ZOMBIE_QTY + WAVE_3_ZOMBIE_QTY

){

    // Start Wave 4

    if (!mutedMusic) {

        playSound(SOUNDTRACK);

    }

    startWave(WAVE_4_ZOMBIE_FRQ, WAVE_4_ZOMBIE_QTY);

} else if (zombieKilled >= ALL_ZOMBIES) {

    // End Game

    endGame("win");

}

};

// Track Zombies on screen

let trackZombies = function repeatOften() {

    let $zombie = $(".zombie");
```

```
if ($zombie.length !== 0) {  
  for (let i = 0, z = $zombie.length; i < z; i++) {  
    let zombieWidth = $zombie.eq(i).width() - 20;  
  
    if ($zombie.eq(i).hasClass("tracking")) {  
      if ($zombie.eq(i).position().left.toFixed() <= -zombieWidth) {  
        $zombie.eq(i).remove();  
        createZombies();  
        life--;  
        if (!mutedSounds) {  
          playSound("PUNCH_" + getRandom(2, 4));  
        }  
        $lifelcons  
          .find(".heart-icon")  
          .not(".hide")  
          .eq(-1)  
          .addClass("hide");  
      }  
    } else {  
      $zombie.eq(i).addClass("tracking");  
    }  
  }  
}
```



```
if (life !== 0) {  
  if (!pauseZombieTracking) {  
    requestAnimationFrame(trackZombies);  
  }  
} else {  
  endGame("lose");  
}  
};
```

```
// Reload Ammo
```

```
const reload = function () {  
  if (!mutedSounds) {  
    playSound(RELOAD_SOUND);  
  }  
  ammo = 6;  
  
  setTimeout(function () {  
    $ammoTitle.addClass("reload");  
  }, 120);  
  setTimeout(function () {  
    $ammoTitle.attr("data-ammo", ammo);  
  }, 150);  
  setTimeout(function () {  
    $ammoTitle.removeClass("reload");
```

```
    }, 250);

    $reloadHint.removeClass("visible");
};

// Mute Music
$muteMusic.on("click", function () {
    let $this = $(this);
    $this.toggleClass("muted");
    if (!mutedMusic) {
        createjs.Sound.stop();
        mutedMusic = true;
    } else {
        mutedMusic = false;
    }
    if (ammo !== 0) {
        ammo++;
    }
});

// Mute Sounds
$muteSounds.on("click", function () {
    let $this = $(this);
    $this.toggleClass("muted");

    !mutedSounds ? (mutedSounds = true) : (mutedSounds = false);
```

```
if (ammo !== 0) {  
    ammo++;  
}  
});  
  
// Pause Game  
$pasueGameTrigger.on("click", function () {  
    let $this = $(this);  
  
    if (!gamePaused) {  
        $this.addClass("paused");  
        $canves.addClass("game-paused");  
        createjs.Sound.stop();  
        $("body").off("keydown");  
        $canves.off("click");  
        gamePaused = true;  
    } else {  
        $this.removeClass("paused");  
        $canves.removeClass("game-paused");  
        setHandlers();  
        gamePaused = false;  
        ammo++;  
    }  
});
```

```
const resetGame = function () {  
  
    zombieKilled = 0;  
  
    wave = 0;  
  
    ammo = 6;  
  
    life = 3;  
  
    $lifeIcons.find(".heart-icon").removeClass("hide");  
  
    $skilledTitle.html(zombieKilled);  
  
    $ammoTitle.attr("data-ammo", ammo);  
  
    createjs.Sound.stop();  
  
    // Clear all zombies on screen  
    $(".zombie").remove();  
  
};  
  
  
// Start Game  
const startGame = function () {  
  
    // reset game  
    resetGame();  
}
```

```
// Game Cover

$gameCover.fadeIn("slow", function () {

  $canves.attr("data-wave", "1");

});

setTimeout(function () {

  $gameCover.fadeOut("slow", function () {

    // Start Wave 1

    if (!mutedMusic) {

      playSound(SOUNDTRACK);

    }

    startWave(WAVE_1_ZOMBIE_FRQ, WAVE_1_ZOMBIE_QTY);

  });

}, 2500);

}; const preload = function (files, cb) {

var len = files.length;

$(

  files

  .map(function (f) {

    return '';

  })

  .join("")

).load(function () {

  if (--len === 0) {

    cb();

  }

}
```

```
});  
  
};  
  
return {  
  initInto: function () {  
    $canves  
      .find(".zombie-loader")  
      .addClass("zombie-" + getRandom(1, 3));  
  
    // Preload all games graphics  
    preload(  
      [  
        "zombie-1.png",  
        "zombie-2.png",  
        "zombie-3.png",  
        "wave1.png",  
        "wave2.png",  
        "wave3.png",  
        "wave4.png",  
        "frame.png",  
        "icon-pack.png",  
        "zombie-1-death.png",  
        "zombie-2-death.png",  
        "zombie-3-death.png",  
        "zombie-4.png",  
        "zombie-4-death.png",
```

```
"zombie-5.png",  
"zombie-5-death.png",  
"zombie-6.png",  
"zombie-6-death.png",  
],  
function () {  
    $canves.find(".loader").remove();  
    startGame()  
}  
);  
},  
killed: zombieKilled,  
ammoLeft: ammo,  
};  
})();  
// Init Game  
ZOMBIEMAYEM.initInto();  
});  
</script>  
</body>  
  
</html>
```

OUTPUT





