IF HTML

<DOCTYPE html>

    <html lang="en">

    <head>

        <meta charset="UTF-8">

        <meta name="viewport" content="width=device width,

        initial-scale=1.0">

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

        <title>Interesting Facts</title>

        <link href="IF.css" rel="stylesheet">

        <script defer src="IF.js"></script>

    </head>

    <body>

        <!-- <button data-modal-target="#modal">Open Modal</button> -->

        <button type="button" class="button">

            <span class="button\_\_icon">

                <ion-icon name="boat-outline"></ion-icon></span></button>

        <div class ="modal" id="modal">

            <div class="modal-header">

                <div class="title">Modal Title</div>

                <button data-close-button class="close-button">&times;</button>

            </div>

            <div class="modal-body">

                TEXT

            </div>

        </button>

        <script src="https://unpkg.com/ionicons@5.4.0/dist/ionicons.js"></script>

    </body>

    </html>

        <!-- <button type="button" class="button">

            <span class="button\_\_text">Download</span>

            <span class="button\_\_icon">

                <ion-icon name="home-outline"></ion-icon>

            </span>

        </button>

        <button type="button" class="button">

            <span class="button\_\_icon">

                <ion-icon name="car-outline"></ion-icon>

            </span>

        </button>

        <button type="button" class="button">

            <span class="button\_\_icon">

                <ion-icon name="enter-outline"></ion-icon>

            </span>

        </button>

        <button type="button" class="button">

            <span class="button\_\_icon">

                <ion-icon name="exit-outline"></ion-icon>

            </span>

        </button>

        <script src="https://unpkg.com/ionicons@5.4.0/dist/ionicons.js"></script>

    </body>

    </html> -->

IF CSS

.button {

  display: inline-flex;

  height: 65px;

  padding: 0;

  background: #2C3860;

  border: none;

  outline: none;

  border-radius: 5px;

  overflow: hidden;

  font-family: "Quicksand", sans-serif;

  font-size: 40px;

  font-weight: 500;

  cursor: pointer;

}

.button:hover {

  background: #F49E92;

}

.button:active {

  background: #FFCD7F;

}

.button\_\_text,

.button\_\_icon {

  display: inline-flex;

  align-items: center;

  padding: 0 24px;

  color: #fff;

  height: 100%;

}

.button\_\_icon {

  font-size: 1.5em;

  background: rgba(0, 0, 0, 0.08);

}

.modal {

    position: fixed;

    top: 50%;

    left: 50%;

    transform: translate(-50%, -50%);

    /\* scale(0) \*/

    /\* transition: 200ms ease-in-out; \*/

    border: 1px solid black;

    border-radius: 10px;

    z-index: 10;

    background-color: white;

    width: 500px;

    max-width: 80%;

}

/\* .modal.active {

    transform: translate(-50%, -50%) scale(1);

} \*/

.modal-header {

    padding: 10px 15px;

    display: flex;

    justify-content: space-between;

    align-items: center;

    border-bottom: 1px solid Black;

}

.modal-header  .title {

    font-size: 1.25rem;

    font-weight: bold;

}

.modal-header .close-button {

    cursor: pointer;

    border: none;

    outline: none;

    background: none;

    font-size: 1.25rem;

    font-weight: bold;

}

.modal-body {

    padding: 10px 15px;

}

#overlay {

    position: fixed;

    opacity: 0;

    transition: 200ms ease-in-out;

    top: 0;

    left: 0;

    right: 0;

    bottom: 0;

    background-color: rgba(0, 0, 0, .5);

    pointer-events: none;

}

#overlay.active {

    opacity: 1;

    pointer-events: all;

}

IF JS

const openModalButtons = document.querySelectorAll('[data-modal-target]')

const closeModalButtons = document.querySelectorAll('[data-close-button]')

const overlay = document.getElementById('overlay')

openModalButtons.forEach(button => {

    button.addEventListener('click', () => {

        const modal = document.querySelector(button.dataset.modalTarget)

        openModal(modal)

    })

})

overlay.addEventListener('click', () => {

    const modals = document.querySelectorAll(',modal.active')

    modals.forEach(modal => {

        closeModal(modal)

    })

})

closeModalButtons.forEach(button => {

    button.addEventListener('click', () => {

        const modal = button.closest('modal')

        closeModal(modal)

    })

})

function openModal(modal) {

    if (modal == null) return

    modal.classlist.add('active')

    overlay.classList.add('active')

}

function closeModal(modal) {

    if (modal == null) return

    modal.classlist.remove('active')

    overlay.classList.remove('active')

}