github : <https://github.com/ChhornKakada/IntenetProgramming/tree/tp5>

# ex1. Activity Suggestion

* [Index.html](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex1.%20Activity%20Suggestion/index.html) (ex1. Activity Suggestion)

<!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>Activity Suggestion</title>

  <script *src*="./tailwindcss.js"></script>

  <link *rel*="stylesheet" *href*="styles.css">

</head>

*<!-- bg-blue-500 sm:bg-black md:bg-green-500 lg:bg-red-500*

*xl:bg-[#D3662D] 2xl:bg-[#FFFF00] 3xl:bg-[#A020F0] -->*

<body *class*="text-[1.2rem] mt-12 ">

  <div *style*="font-family: 'Julee', cursive;" *class*="font-bold">

    <center>

      <div *class*="text-[144px]">😒</div>

      <h1 *class*="text-[5rem] mt-[-40px]">Are you boring ?</h1>

      <button *onclick*="getMeAnActivity()" *id*="generateAct"

*class*="bg-blue-500 px-[50px] py-[10px] text-[1.2rem] text-white rounded-lg my-[26px]">

        Tell me what to do</button>

    </center>

  </div>

*<!-- Activity -->*

  <div *class*="activityContainer w-[460px] h-[320px] border-2 rounded-lg mx-auto p-[25px]">

    <div *class*="flex justify-center items-center" *id*="loadingContainer">

      <div *class*="lds-hourglass" *id*="loadingText" *hidden*></div>

    </div>

    <div *id*="activity" *hidden*>

      <div *id*="activityName" *class*="text-[1.5rem] text-center mb-6 font-bold"></div>

*<!-- type -->*

      <div *class*="iconContainer">

        <div>

          <svg *xmlns*="http://www.w3.org/2000/svg" *viewBox*="0 0 24 24" *fill*="currentColor" *class*="w-6 h-6">

            <path *fill-rule*="evenodd"

*d*="M5.25 2.25a3 3 0 …0 000 2.25z"

*clip-rule*="evenodd" />

          </svg>

        </div>

        <div>

          <p> Type : <span *id*="type"></span></p>

        </div>

      </div>

*<!-- participants -->*

      <div *class*="iconContainer">

        <div>

          <svg *xmlns*="http://www.w3.org/2000/svg" *viewBox*="0 0 24 24" *fill*="currentColor" *class*="w-6 h-6 ">

            <path

*d*="M4.5 6.375a4.125 …011.71 5.157v.003z" />

          </svg>

        </div>

        <div>Participants : <span *id*="participants"></span></div>

      </div>

*<!-- Price -->*

      <div *class*="iconContainer">

        <div>

          <svg *xmlns*="http://www.w3.org/2000/svg" *viewBox*="0 0 24 24" *fill*="currentColor" *class*="w-6 h-6 ">

            <path

*d*="M10.464 8.746c.227-.18.497-…0 01-.921.42z" />

            <path *fill-rule*="evenodd"

*d*="M12 2.25c-5.385 0-9.75 …0 00-1.719-.755V6z"

*clip-rule*="evenodd" />

          </svg>

        </div>

        <div *class*="">Price : $<span *id*="price"></span></div>

      </div>

*<!-- Link -->*

      <div *class*="iconContainer w-full">

        <div *class*="self-start">

          <svg *xmlns*="http://www.w3.org/2000/svg" *viewBox*="0 0 24 24" *fill*="currentColor" *class*="w-6 h-6">

            <path *fill-rule*="evenodd"

*d*="M19.902 4.098a3.75 3.75 …0 01-.354-1z"

*clip-rule*="evenodd" />

          </svg>

        </div>

        <div *class*="">

          <p *class*="">Link : <span *id*="link"></span></p>

        </div>

      </div>

    </div>

  </div>

  <script *src*="./app.js"></script>

</body>

</html>

* [app.js](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex1.%20Activity%20Suggestion/app.js) (ex1. Activity Suggestion)

const activity = document.getElementById("activity");

const activityContainer = document.getElementById("activityContainer");

const activityName = document.getElementById("activityName");

const type = document.getElementById("type");

const participants = document.getElementById("participants");

const price = document.getElementById("price");

const link = document.getElementById("link");

const loadingContainer = document.getElementById("loadingContainer");

const loadingStop = document.getElementById("loadingStop");

function getMeAnActivity() {

  loadingContainer.style.height = "100%";

  loadingText.hidden = false;

  activity.hidden = true;

  fetch("https://www.boredapi.com/api/activity")

  .then((res) => {

    res.json().then((data) => {

      activityName.innerHTML = `${data["activity"]}`;

      type.innerHTML = data["type"];

      participants.innerHTML = data["participants"];

      price.innerHTML = data["price"];

      link.innerText = data["link"];

      console.log(data);

      activity.hidden = false;

      loadingContainer.style.height = "0px";

      loadingText.hidden = true;

    })

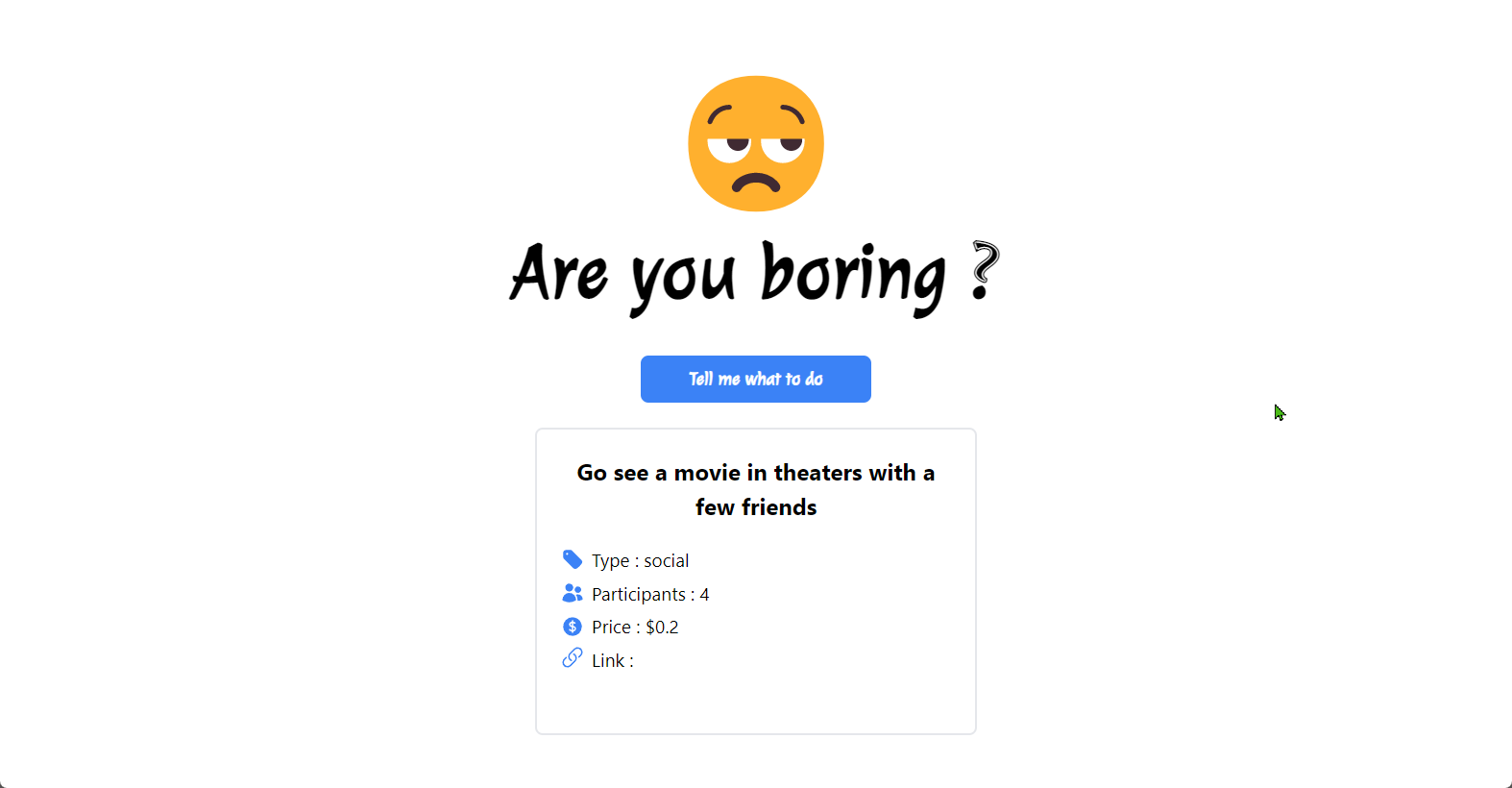
  })

  .catch((err) => {

    console.log(err);

  })

}



# ex2. Gender Guessing

* [index.html](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex2.%20Gender%20Guessing/index.html) (ex2. Gender Guessing)

<!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>Gender Guessing</title>

  <script *src*="./tailwindcss.js"></script>

  <link *rel*="stylesheet" *href*="./styles.css">

</head>

<body>

  <div *class*="pt-[60px]">

    <center>

      <div>

        <div *class*="text-[9rem]">🤔</div>

        <h1 *class*="text-[3rem] sm:text-[5rem] font-bold" *style*="font-family: 'Julee', cursive;">I know your <span

*class*="text-blue-500">Gender</span></h1>

        <div *class*="mb-4 text-[1.5rem] py-4">

          <input *type*="text" *id*="username" *name*="" *class*="border-2 rounded-lg h-10 pl-2" *placeholder*="Tell me your name"

*required*>

          <input *type*="submit" *id*="submitBtn" *class*="bg-blue-500 h-10 text-white px-4 rounded-lg" *onclick*="startGuessing()">

        </div>

      </div>

*<!-- data -->*

      <div *class*="py-4">

        <div *id*="loading" *class*="mt-[70px]" *hidden*>

          <div *class*="lds-ripple">

            <div></div>

            <div></div>

          </div>

        </div>

        <div *id*="data" *hidden* *class*=" bg-[#F4F4F4] border-2 w-[460px] h-[220px] rounded-2xl py-4">

          <p *id*="name" *class*="font-bold text-[2rem] mb-4"></p>

          <p *id*="gender" *class*="font-bold text-[#F06E24] text-[3.2rem]"></p>

          <p *id*="percentage" *class*="font-bold text-[#72AAFF] text-[1.8rem]"></p>

        </div>

      </div>

    </center>

  </div>

  <script *src*="./app.js"></script>

</body>

</html>

* [app.js](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex2.%20Gender%20Guessing/app.js) (ex2. Gender Guessing)

const username = document.getElementById("username");

const loading = document.getElementById("loading");

const Data = document.getElementById("data");

var guessing = {

  name: document.getElementById("name"),

  gender: document.getElementById("gender"),

  percentage: document.getElementById("percentage")

}

function startGuessing() {

  var url = "https://api.genderize.io?name=" + username.value;

  loading.hidden = false;

  Data.hidden = true;

  fetch(url)

    .then((res) => {

      res.json().then((data) => {

        guessing.name.innerHTML = data["name"];

*if* (data["gender"] == null && data["probability"] == 0) {

          guessing.gender.innerHTML = "Sorry, I have no idea for this name.";

          guessing.gender.style.fontSize = "1.5rem";

          guessing.percentage.innerHTML = "😥";

          guessing.percentage.style.fontSize = "3.8rem";

        }

*else* {

          guessing.gender.innerHTML = data["gender"].charAt(0).toUpperCase() + data["gender"].slice(1);

          guessing.percentage.innerHTML = (data["probability"] \* 100) + "%";

          guessing.gender.style.fontSize = "";

          guessing.percentage.style.fontSize = "";

        }

        loading.hidden = true;

        Data.hidden = false;

        username.value = "";

      })

    })

    .catch((err) => {

      alert("error");

      console.log(err);

    })

}

*// when the user press enter to submit*

username.addEventListener("keyup", (event) => {

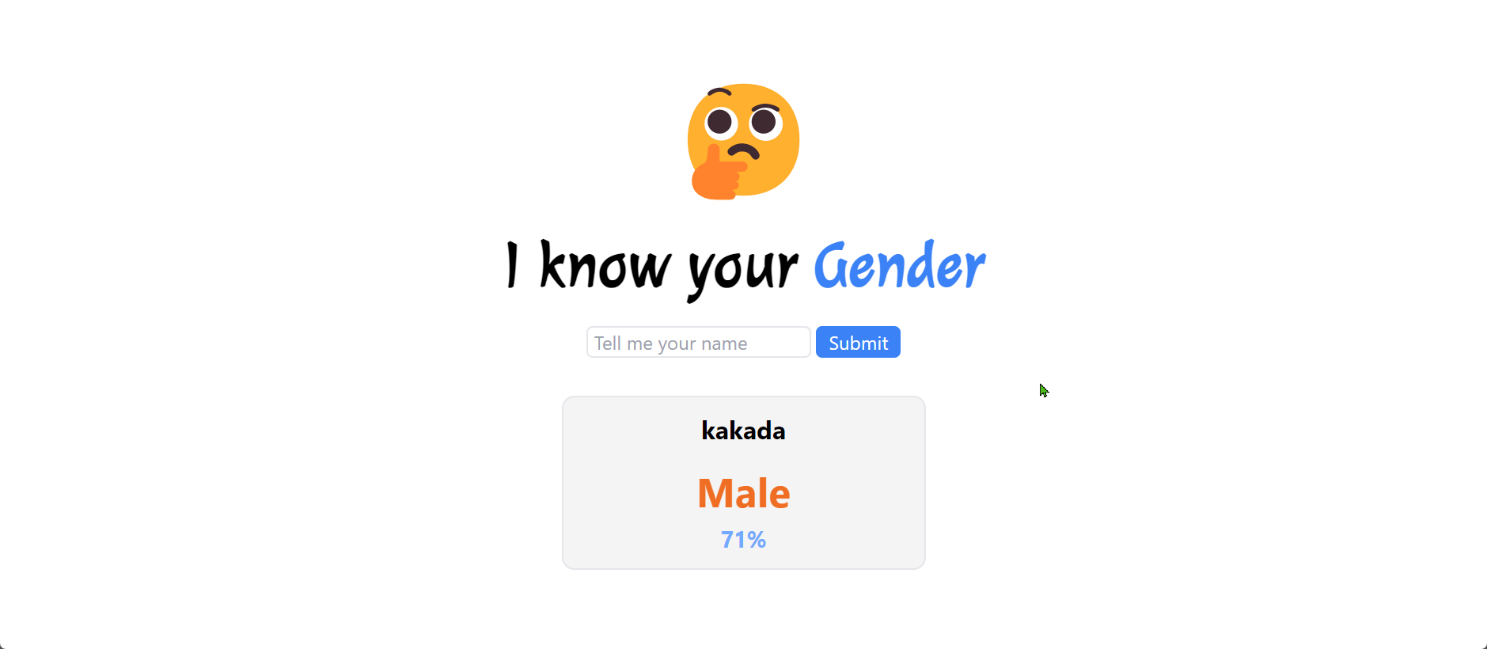
*if* (event.key === "Enter") {

    event.preventDefault();

    document.getElementById("submitBtn").click();

  }

})



# ex3. Nationality Guessing

* [index.html](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex3.%20Nationality%20Guessing/index.html) (ex3. Nationality Guessing)

<!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>Nationality Guessing</title>

  <script *src*="./tailwindcss.js"></script>

  <link *rel*="stylesheet" *href*="./styles.css">

</head>

<body>

  <div *class*="mt-10">

    <center>

      <div *class*="mb-6">

        <p *class*="text-[8rem]">🌎</p>

        <p *class*="text-[3rem] sm:text-[4rem] font-bold juleeFont">I know your

          <span *class*="text-blue-500">Home</span>

        </p>

        <div>

          <input *type*="text" *id*="username" *name*="" *class*="border-2 rounded-lg h-10 pl-2 text-[1.2rem] mr-2"

*placeholder*="Tell me your name" *required*>

          <button *id*="submitBtn" *class*="bg-blue-500 h-10 text-white px-4 rounded-lg text-[1.3rem]"

*onclick*="startGuessing()">Guess</button>

        </div>

      </div>

*<!-- loading -->*

      <div *id*="loading" *class*="mt-24" *hidden*>

        <div *class*="lds-hourglass"></div>

      </div>

*<!-- output -->*

      <div *class*="text-[1.3rem] font-bold" *id*="output" *hidden*>

        <div *class*="text-[2rem] juleeFont" *id*="nameInput"></div>

*<!-- datas -->*

        <div *id*="results"></div>

      </div>

    </center>

  </div>

  <script *src*="./app.js"></script>

</body>

</html>

* [app.js](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex3.%20Nationality%20Guessing/app.js) (ex3. Nationality Guessing)

const results = document.getElementById("results");

const username = document.getElementById("username");

const nameInput = document.getElementById("nameInput");

const submitBtn = document.getElementById("submitBtn");

const loading = document.getElementById("loading");

const output = document.getElementById("output");

function toPercentage(num) {

*return* (num \* 100).toFixed(2);

}

*// when there are data.country.length > 0*

function guessed(data) {

*for* (let i = 0; i < data.country.length; i++) {

    results.innerHTML += `

    <div class="flex w-[330px] justify-between h-10 border-2 rounded-xl items-center px-4 bg-[#F4F4F4] mt-1">

      <p>${data.country[i].country\_id}</p>

      <p class="text-[#FF5C00]">${toPercentage(data.country[i].probability)}%</p>

    </div>

    `

  }

}

*// when there are data.country.length = 0*

function cannotGuess() {

  results.innerHTML = `

  <div class="w-[330px] border-2 rounded-xl py-4 bg-[#F4F4F4] mt-4">

    <p class="text-[#FF5C00] text-[1.5rem]">Sorry, I have no idea.</p>

    <p class="text-[4rem]">😥</p>

  </div>

  `

}

function startGuessing() {

  results.innerHTML = "";

  loading.hidden = false;

  output.hidden = true;

  fetch(`https://api.nationalize.io?name=${username.value}`)

    .then((res) => {

      res.json().then((data) => {

        nameInput.innerHTML = username.value;

        data.country.length == 0 ? cannotGuess() : guessed(data);

        username.value = "";

        loading.hidden = true;

        output.hidden = false;

      })

    })

    .catch((err) => {

      console.log(err);

    })

}

*// when the user enter to input the name instead of click the button*

username.addEventListener("keyup", (event) => {

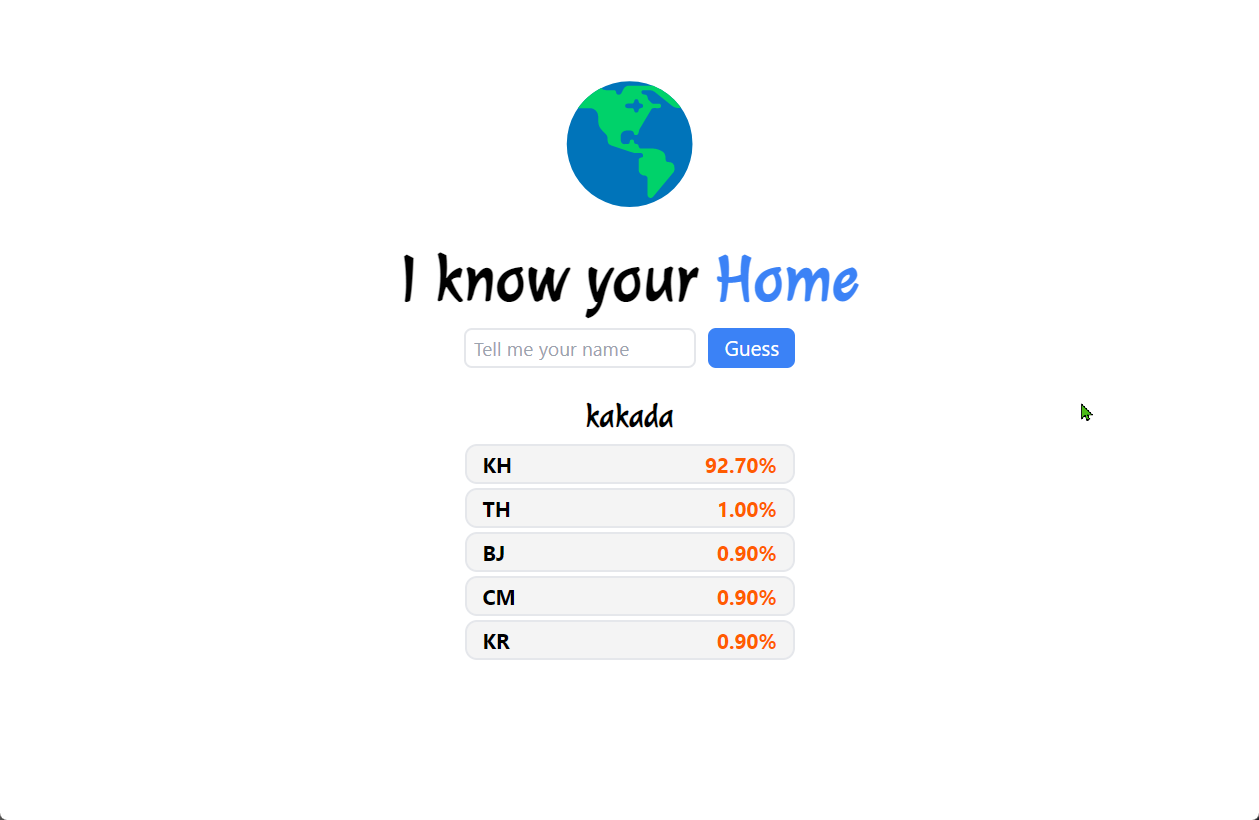
*if* (event.key === "Enter") {

    event.preventDefault();

    submitBtn.click();

  }

})



# ex4. University List

* [index.html](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex4.%20University%20List/index.html) (ex4. University List)

<!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>University List</title>

  <script *src*="./tailwindcss.js"></script>

  <link *rel*="stylesheet" *href*="./styles.css">

</head>

*<!-- bg-blue-500 sm:bg-black md:bg-green-500 lg:bg-red-500*

*xl:bg-[#D3662D] 2xl:bg-[#FFFF00] 3xl:bg-[#A020F0] -->*

<body *class*="">

  <div>

*<!-- header -->*

    <div *class*="header shadow-xl shadow-white">

      <center>

        <p *class*="text-[8rem]">🏫</p>

        <p *class*="text-[3rem] sm:text-[4rem] font-bold juleeFont px-8 xl:px-0">

          Which Univ. Do you want to study?

        </p>

        <div>

          <select *name*="unis" *id*="unis" *onchange*="displayUnivs()"

*class*="border-2 rounded-lg h-12 pl-2 text-[1.5rem] mr-2 sm:w-[500px] w-[70%]">

            <option *selected*="selected" *value*="Cambodia">Cambodia</option>

            <option *value*="Thailand">Thailand</option>

*<!-- <option value="France">France</option> -->*

            <option *value*="Canada">Canada</option>

            <option *value*="Singapore">Singapore</option>

            <option *value*="Ukraine">Ukraine</option>

            <option *value*="Australia">Australia</option>

            <option *value*="Switzerland">Switzerland</option>

*<!-- <option value="Japan">Japan</option> -->*

          </select>

        </div>

      </center>

    </div>

*<!-- loading -->*

    <div *id*="loading" *class*="pt-10" *hidden*>

      <center>

        <div *class*="lds-ellipsis">

          <div></div>

          <div></div>

          <div></div>

          <div></div>

        </div>

      </center>

    </div>

*<!-- output -->*

    <div *class*=" w-[90%] flex justify-center mx-auto overflow-auto flex-wrap gap-10 my-5" *id*="uniList" *hidden*>

    </div>

  </div>

  <script *src*="./app.js"></script>

</body>

</html>

* [app.js](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex4.%20University%20List/app.js) (ex4. University List)

const unis = document.getElementById("unis");

const uniList = document.getElementById("uniList");

const loading = document.getElementById("loading");

function displayAUni(uni) {

  uniList.innerHTML += `

  <div class="items-center border-2 w-[280px] text-[1.1rem] rounded-xl bg-[#F4F4F4] py-2 text-center px-3">

    <p class="font-bold text-[1.2rem]">${uni.name}</p>

    <p class="mb-3">(${uni.country} - ${uni.alpha\_two\_code})</p>

    <a href="${uni.web\_pages}" class="" target="\_blank">

      <span class="linkUni flex gap-2 text-blue-500 items-center justify-center">

        <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 24 24" fill="currentColor" class="w-6 h-6">

          <path

            d="M15.75 8.25a.75…..75.75 0 01.75-.75z" />

          <path fill-rule="evenodd"

            d="M12 2.25c-5.385 0-9.75 ...1.125 0 00-1.906.172L4.575 15.6z"

            clip-rule="evenodd" />

        </svg>

        <p>${uni.web\_pages}</p>

      </span>

    </a>

  </div>

  `

}

function displayUnivs() {

  uniList.innerHTML = "";

  loading.hidden = false;

  uniList.hidden = true;

  fetch(`http://universities.hipolabs.com/search?country=${unis.value}`)

    .then(async (res) => {

      const data = *await* res.json();

*for* (let uni of data) {

        displayAUni(uni);

        loading.hidden = true;

        uniList.hidden = false;

      }

    }).catch((err) => {

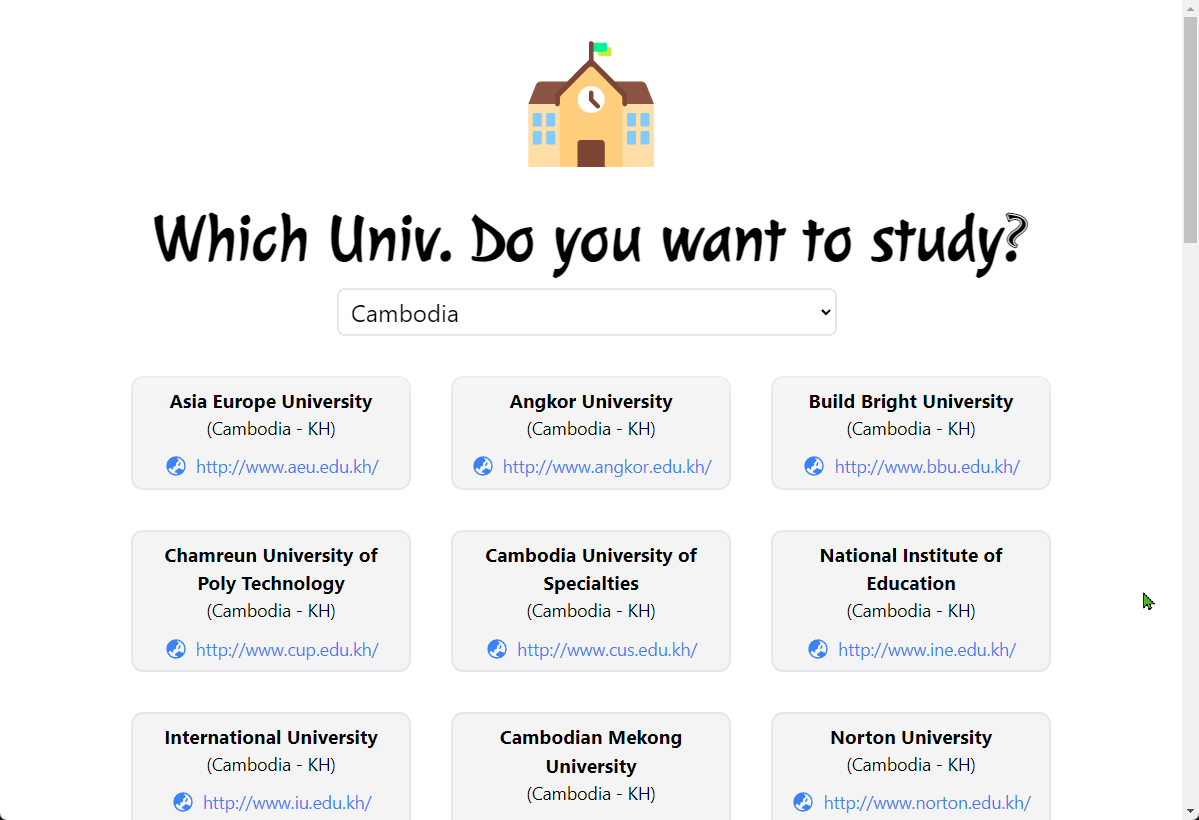
      alert("error");

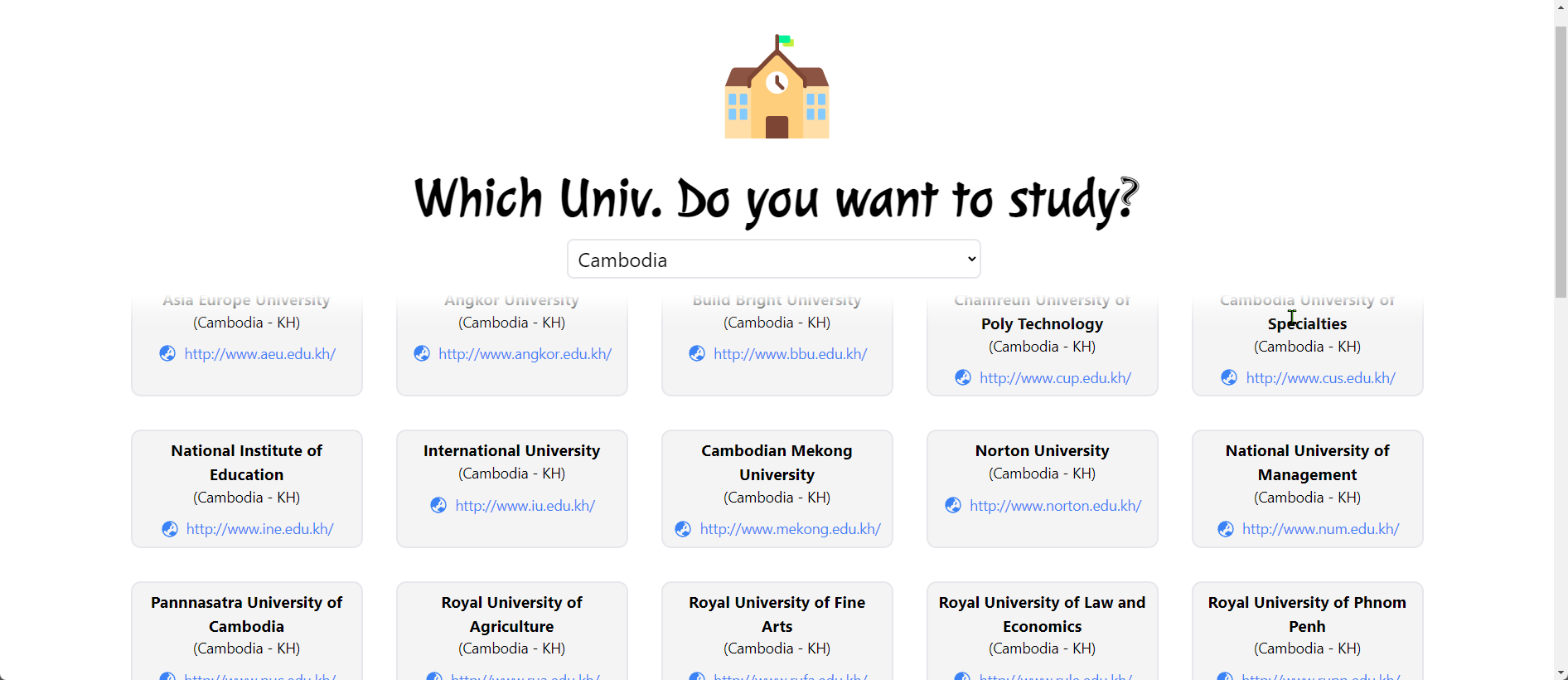
      console.log(err);

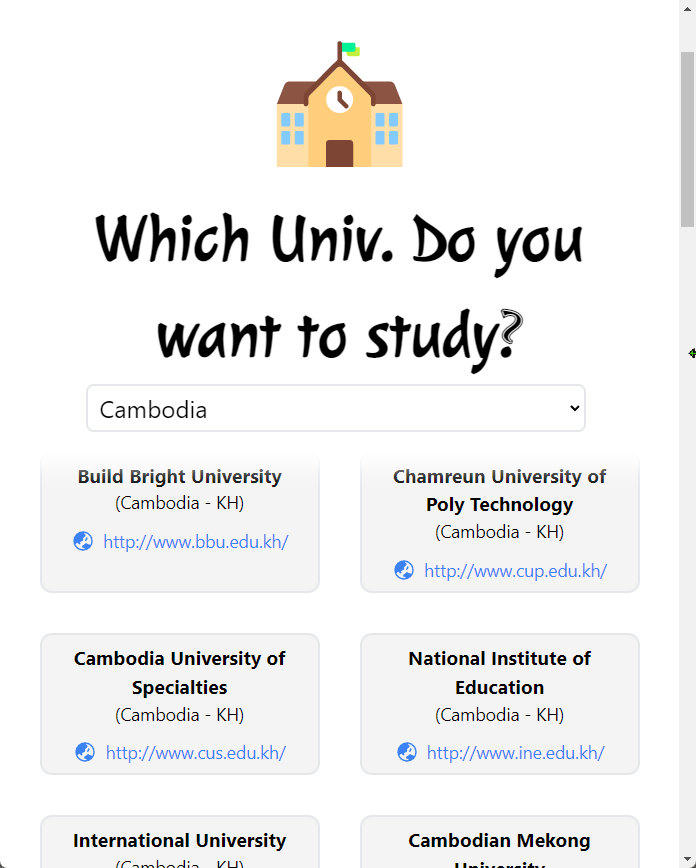
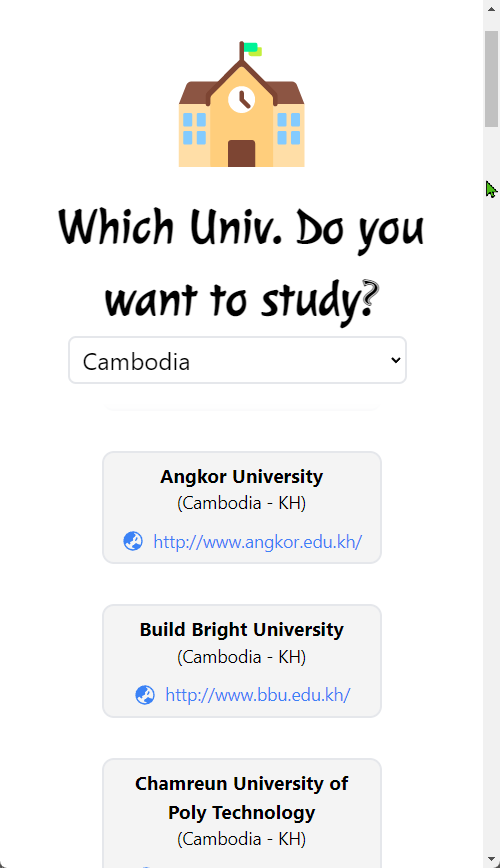
    })

}

displayUnivs();







# ex5. Book Store

* [index.html](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex5.%20Book%20Store/index.html) (ex5. Book Store)

<!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>Book Store</title>

  <script *src*="./tailwindcss.js"></script>

  <link *rel*="stylesheet" *href*="./styles.css">

</head>

<body *class*="">

  <div *class*="flex">

*<!-- input -->*

    <div *class*="min-w-[280px]">

      <div *class*="w-full bg-slate-100 border-2 rounded-lg p-8 min-h-[270px] h-[100vh]">

        <div>

          <label *for*="name">Name : </label> <br>

          <input *type*="text" *id*="name" *name*="name" *value*="" *placeholder*="C programming" *class*="inputForm">

        </div>

        <div>

          <label *for*="category">Category : </label> <br>

          <input *type*="text" *id*="category" *name*="category" *value*="" *placeholder*="Coding" *class*="inputForm">

        </div>

        <div>

          <label *for*="price">Price : </label> <br>

          <input *type*="number" *id*="price" *name*="price" *value*="" *placeholder*="1,000 riel" *class*="inputForm">

        </div>

        <div>

          <button *type*="submit" *class*="px-5 rounded-lg bg-blue-500 text-white"

*onclick*="addANewBook()" *id*="submitBtn">Add</button>

        </div>

      </div>

    </div>

*<!-- books -->*

    <div *id*="bookStore" *class*="flex h-[100vh] min-h-[270px] flex-wrap gap-1 overflow-auto p-1">

*<!-- a book -->*

    </div>

  </div>

  <script *src*="./app.js"></script>

</body>

</html>

* [app.js](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex5.%20Book%20Store/app.js) (ex5. Book Store)

const bookStore = document.getElementById("bookStore");

const newBook = {

  title: document.getElementById("name"),

  category: document.getElementById("category"),

  price: document.getElementById("price")

}

var localBooks = localStorage.getItem("books") != null ? JSON.parse(localStorage.getItem("books")) : [];

*// random color*

function randomColor() {

  const hexValue = [1, 2, 3, 4, 5, 6, 7, 8, 9, 'a', 'b', 'c', 'd', 'e', 'f'];

  var color = '';

*for* (let i = 0; i < 6; i++) {

    var index = Math.floor(Math.random() \* hexValue.length);

    color += hexValue[index];

  }

*return* color;

}

*// for a book that create and from localStorage*

function showABookLocal(book) {

  const params = `?title=${book.title}&category=${book.category}&price=${book.price}&img=${book.urlImage}`;

  bookStore.innerHTML += `

  <div class="w-[200px] border-2 rounded-lg p-[25px]">

    <img src="${book.urlImage}" alt="">

    <h3 class="font-bold mt-6">${book.title}</h3>

    <p>Category : ${book.category}</p>

    <p>Price : ${book.price}</p>

    <a href="./view.html${params}" target="\_blank" class="text-blue-500">see</a>

  </div>

  `

}

function displayLocalBook() {

*for* (let book of localBooks) {

    showABookLocal(book);

  }

}

fetch("https://jsonplaceholder.typicode.com/photos")

  .then(async (res) => {

    const books = *await* res.json();

*for* (let i = 0; i < 30; i++) {

      showABookApi(books[i]);

    }

    displayLocalBook();

  }).catch((err) => {

    console.log(err);

  })

*// for a book that get from API*

function showABookApi(book) {

  bookStore.innerHTML += `

  <div class="w-[200px] border-2 rounded-lg p-[25px]">

    <img src="${book.thumbnailUrl}" alt="">

    <h3 class="font-bold mt-6">${book.title}</h3>

    <p>Album id : ${book.albumId}</p>

    <p>Category : 1</p>

    <a href="./view.html?id=${book.id}" target="\_blank" class="text-blue-500">

See

</a>

  </div>

  `

}

function resetForm() {

  newBook.title.value = "";

  newBook.category.value = "";

  newBook.price.value = "";

}

function addANewBook() {

  var book = {};

  book["title"] = document.getElementById("name").value;

  book["category"] = document.getElementById("category").value;

  book["price"] = document.getElementById("price").value;

  book["urlImage"] = `https://via.placeholder.com/150/${randomColor()}`;

  localBooks.push(book);

  saveToLocalstorage(localBooks);

  showABookLocal(book);

  resetForm();

}

const saveToLocalstorage = (books) => {

  localStorage.setItem("books", JSON.stringify(books));

}

* [view.html](https://github.com/ChhornKakada/IntenetProgramming/blob/tp5/ex5.%20Book%20Store/view.html) (ex5. Book Store)

<!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>View</title>

  <script *src*="./tailwindcss.js"></script>

</head>

<body>

  <center>

  <div *id*="view" *class*="text-[1.3rem]">

  </div>

</center>

  <script>

    const keyValue = window.location.search;

    const urlParams = new URLSearchParams(keyValue);

    const bookId = urlParams.get("id");

    const localBook = {

      title : urlParams.get("title"),

      category : urlParams.get("category"),

      price : urlParams.get("price"),

      imgUrl : urlParams.get("img"),

    }

    const view = document.getElementById("view");

    localBook.title != null ? viewDetailABookLocal(localBook) : viewInHtmlApi();

    function viewInHtmlApi() {

      fetch(`https://jsonplaceholder.typicode.com/photos/${bookId}`)

      .then((res) => {

        res.json().then((data) => {

          viewDetailABookApi(data);

        })

      })

      .catch((err) => {

        console.log(err);

      })

    }

    function viewDetailABookApi(book) {

      view.innerHTML = `

      <div class="bg-[#474E5D] text-white p-12">

      <h2 class="text-[2rem]">${book.title}</h2>

      <p >${book.id}</p>

      <p>Resize the browser windoow to see that this page is responsive by the way.</p>

      </div>

      <div class="">

        <h3 class="my-4">${book.id}</h3>

        <img src="${book.thumbnailUrl}" alt="">

      </div>

      `

    }

    function viewDetailABookLocal(book) {

      view.innerHTML = `

      <div class="bg-[#474E5D] text-white p-12">

      <h2 class="text-[2rem]">${book.title}</h2>

      <p>Category : ${book.category}</p>

      <p>Resize the browser windoow to see that this page is responsive by the way.</p>

      </div>

      <div class="">

        <h3 class="my-4">Price : ${book.price}</h3>

        <img src="${book.imgUrl}" alt="">

      </div>

      `

    }

  </script>

</body>

</html>

