

Assignments

- Assignments
 - PE01
 - PE02
 - PE03
 - Task #1
 - Task #2
 - Task #3
 - PE04
 - PE05
 - PE06
 - PE07
 - PE08
 - PE09
 - PE10

PE01

► PE01

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Trevors' Webpage</title>
    <link rel="stylesheet" href="style.css">
    <meta name="author" content="Trevor Sparks"/>
  </head>
  <body>
    <p>Hello City University of Seattle!</p>
    <a href="https://www.cityu.edu" class="cityULink" target="_blank">City
Univeristy of Seattle homepage</a>

    <h2>What Can JavaScript Do?</h2>

    <p>JavaScript can change HTML attribute values.</p>

    <p>In this case JavaScript changes the value of the src (source) attribute
of an image.</p>

    <button
onclick="document.getElementById('myImage').src='pic_bulbon.gif'">Turn on the
light</button>

    

    <button
onclick="document.getElementById('myImage').src='pic_bulboff.gif'">Turn off the
```

```
light</button>

</body>
</html>
```

CSS

```
body{
  background-color: lightgrey;
}

p{
  font-family: "monaco", monospace;
}

.cityULink{
  font-family: "monaco", monospace;
}
.cityULink:link{color: black;}
.cityULink:visited{color: blue;}
```

PE02

► PE03

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Trevors' Webpage</title>
    <link rel="stylesheet" href="style.css">
    <meta name="author" content="Trevor Sparks"/>
  </head>
  <body>
    <h1>PE01</h1>
    <p>Hello City University of Seattle!</p>
    <a href="https://www.cityu.edu" class="cityULink" target="_blank">City
Univeristy of Seattle homepage</a><br>

    <h1>PE02</h1>
    <img id="lightbulb"><br>
    <button id="switch"></button><br>
    <a href="https://www.w3schools.com/js/js_intro.asp"
target="_blank">Lightbulb photos from W3 school</a><br>

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

```
</body>
</html>
```

```
body{
  background-color: lightgrey;
}

p{
  font-family: "monaco", monospace;
}

.cityULink{
  font-family: "monaco", monospace;
}
.cityULink:link{color: black;}
.cityULink:visited{color: blue;}
```

```
// Gets litbulb photo info and sets the lighbulb to
// the off lightbulb photo
var lightbulb = document.getElementById('lightbulb')
lightbulb.src = "pic_bulboff.gif"

// Gets litbulb button info and sets the lighbulb to
// the off lightbulb button text to "off"
var lightswitch = document.getElementById('switch')
var stateOfLightswitch = document.getElementById('switch').innerText;
lightswitch.innerText = "Off"

// The function below turns on an off the lightbulb with an if statement.
lightswitch.onclick = function() {
  if(lightswitch.innerText == "On"){
    lightswitch.innerText = "Off"
    lightbulb.src = "pic_bulboff.gif"
  }else{
    lightswitch.innerText = "On"
    lightbulb.src = "pic_bulbon.gif"
  }
}
```

PE03

► PE03

Task #1

```
// Declaring Variables
var firstName:string;
var lastName:string;
var fullName:string;
var age:number;
var ukCitizen: boolean;

// Initilizing variables
firstName = 'Rebecca';
lastName = 'Smith';
age = 42;
ukCitizen = false;
fullName = firstName + ' ' + lastName;

// If else statement that checks if the bool value of 'ukCitizen' is true
if(ukCitizen){
    console.log("My name is " + fullName + ", I'm " + age + ", and I'm a citizen
of the United Kingdom.");
} else {
    console.log("My name is " + fullName + ", I'm " + age + ", and I'm not a
citizen of the United Kingdom.");
}

// Declaring Variables
var firstName:string;
var lastName:string;
var fullName:string;
var age:number;
var ukCitizen: boolean;

// Initilizing variables
firstName = 'Rebecca';
lastName = 'Smith';
age = 42;
ukCitizen = false;
fullName = firstName + ' ' + lastName;

// If else statement that checks if the bool value of 'ukCitizen' is true
if(ukCitizen){
    console.log("My name is " + fullName + ", I'm " + age + ", and I'm a citizen
of the United Kingdom.");
} else {
    console.log("My name is " + fullName + ", I'm " + age + ", and I'm not a
citizen of the United Kingdom.");
}
}
```

Task #2

```
// Declaring variables
var x: number;
var y: number;
var a: number;
```

```
// Initializing variables
x = 5;
y = 7;
a = x+y;

// Outputting the value of a
console.log(a);
```

Task #3

```
// Declaring variables
// The value of randomNumbers is initialized to an empty array
var randomNumbers: number[] = [];
var nextNumber: number;

// Pushes random numbers to the randomNumber array
for (let i = 0; i < 10; i++) {
    nextNumber = Math.floor(Math.random() * (100 - 1)) + 1;
    randomNumbers.push(nextNumber);
}

// Outputs the randomNumbers array
console.log(randomNumbers);
```

PE04

► PE04

```
//let pizzaSlices:string[] = ["Cheese", "Peperoni", "Sausage", "BBQ", "Mac & Cheese", "Chicken Alfredo"];

let pizzaSlices =[
    {
        name: "Cheese",
        type: "Vegetarian",
        price: 40
    },{
        name: "Pepperoni",
        type: "Meat",
        price: 35
    },{
        name: "Sausage",
        type: "Meat",
        price: 35
    },{
        name: "BBQ Chicken",
```

```

    type: "Meat",
    price: 35

  }, {
    name: "Mac & Cheese",
    type: "Vegetarian",
    price: 40

  }, {
    name: "Chicken Alfredo",
    type: "Meat",
    price: 35

  }

]

const transactions = [
  {slices: ["Mac & Cheese", "Mac & Cheese", "Pepperoni", "Pepperoni", "BBQ Chicken"], cost: 220},
  {slices: ["Cheese", "Pepperoni", "Pepperoni", "Chicken Alfredo", "Pepperoni", "Cheese"], cost: 200},
  {slices: ["Chicken Alfredo", "Mac & Cheese", "BBQ Chicken", "Chicken Alfredo"], cost: 145},
  {slices: ["Sausage", "BBQ Chicken", "BBQ Chicken", "BBQ Chicken", "Sausage", "Sausage", "Chicken Alfredo"], cost: 245},
  {slices: ["Chicken Alfredo", "Mac & Cheese", "BBQ Chicken"], cost: 110}
]

var totalSales: number = 0 ;

totalSales = transactions.reduce((totalSum, currentSum) => totalSum + currentSum.cost, 0);

console.log(`You've made ${totalSales} today!`);

for(let i = 0; i < transactions.length; i++){
  const currentTransactionSlice = transactions[i].slices;
  console.log(currentTransactionSlice);

  let pizzaSum = currentTransactionSlice.reduce(...);
}

```

PE05

► PE05

```

// Upper and lower limit set
const upperLimit: number = 21;
const lowerLimit: number = 0;
// Initialization of userInput var
let userInput: number;

```

```
//Function to get user input
function getUserInput(): number{
    // Use prompt to prompt the user for input, and the number function to
    // convert the received string to a number
    userInput = Number(prompt('Please input a number between 1 and 20'));
    // call the validate user function to make sure the user input a valid number
    validateUserInput(upperLimit, lowerLimit, userInput);
    return userInput;
}

// function to validate user input. It checks to see if the users input is in
// between two
// pre-specified values
function validateUserInput(upperLimit: number, lowerLimit: number, userInput:
number){
    if(userInput > lowerLimit && userInput < upperLimit){
        return userInput;
    }else{
        console.log("The number is too high or too low");
        getUserInput();
    }
}

// Task 2 function
function someFunction(p1: number, p2: number):number{
    if(p1 <= 0){
        return p2;
    }else{
        p2 = p1 * p2;
        p1--;
        return someFunction(p1,p2);
    }
}

// recursive function to calculate the Fibonacci sequence at the user specified
// index
function calcFibb(input:number){
    if(input === 1){
        return 0;
    }else if(input === 2){
        return 1;
    }else{
        return calcFibb(input-2)+calcFibb(input-1);
    }
}

// function to call the get user function, calculate the other
// necessary numbers, and output HTML to the webpage
function whenTheButtonIsClicked(){
    getUserInput();
    let task2FunctionAnswer = someFunction(userInput, userInput);
    let calcFib = calcFibb(userInput);
    let outputHTML = document.getElementById('output');
    outputHTML.innerHTML = <p>The user input: `${userInput}`</p>
```

```

        <p>Task 2 function answer: `${task2FunctionAnswer}`</p>
        <p>The value of the Fibonacci sequence at the
`${userInput}` position is `${calcedFib}`</p>;
    }

```

```

<!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>PE05 - Trevor Sparks</title>
    <script type="text/javascript" src="PE05.js"></script>
</head>
<body>
    <h1>Click the button to perform calculations and get user input</h1>
    <button onclick="whenTheButtonIsClicked();">Click me!</button>
    <div id="output"></div>
</body>
</html>

```

PE06

► PE06

```

interface Pizza{
    type: string,
    slices: number,
    crust?: string
}

interface Toppings extends Pizza{
    sauce: "Tomato" | "Alfredo" | "BBQ Sauce",
    pineapple?: boolean,
    parmesan?: boolean,
    crust?: string
}

var myPizza: Toppings = ({
    type: "Meat lovers",
    slices: 6,
    sauce: "Tomato",
    pineapple: false,
    parmesan: true,
    crust: "Stuffed"
});

console.log("Number of slices in myPizza: ", myPizza.slices);

function checkSlices(pizza: Toppings): string{

```



```
    if(pizza.slices > 8){
      return "The number of slices is too high";
    }else if(pizza.slices == 8){
      return `The whole pizza is remaining`;
    }else{
      return `There is ${pizza.slices}/8 remaining of the pizza`;
    }
  };

  console.log(checkSlices(myPizza));
```

PE07

PE08

► PE08

```
function primitiveMultiply(num1, num2) {
  let chance = Math.random();
  if (chance > 0.2) {
    return num1 * num2;
  }

  throw new Error("MultiplicatorUnitFaiiure");
}

function Calculate(number1, number2) {
  let result = 0;
  for (;;) {
    try {
      result = primitiveMultiply(number1, number2);
      return result;
    } catch (e) {
      console.log("this try failed, attempting again");
    }
  }
}

console.log("hello from retry", Calculate(3, 6));s
```

```
const box = {
  locked: true,
  unlock() {
    this.locked = false;
  },
  lock() {
```

```
    this.locked = true;
  },
  _content: [],
  get content() {
    if (this.locked) throw new Error("Locked!");
    return this._content;
  },
};

function withBoxUnlocked(cb) {
  try {
    box.unlock();
    cb();
  } catch (error) {
  } finally {
    box.lock();
  }
}
```

```
// Upper and lower limit set
var upperLimit = 21;
var lowerLimit = 0;
// Initialization of userInput var
var userInput;
// function to validate user input. It checks to see if the users input is in
// between two
// pre-specified values
function myFunction() {
  while (true) {
    try {
      //getUserInput();
      userInput = Number(prompt('Please input a number between 1 and 20'));
      // call the validate user function to make sure the user input a valid
number
      if (userInput > lowerLimit && userInput < upperLimit) {
        return userInput;
      }
      else {
        alert("Number not within range, please try again.");
        throw "Number not within range.";
      }
    }
    catch (e) {
      console.log(e);
      continue;
    }
    finally {
      console.log("The user input: " + userInput);
    }
  }
}
```

PE09

► PE09

```
async function locateScalpel(nest){
  current_name = nest.name;
  for(let i=0; i<100; i++){
    console.log("We are currently looking at: " + current_name);

  }
}

async function locateScalpel2(nest){
  console.log(nest);
  return new Promise((resolve, reject) =>{
    // trying to find the scalpel
    let found = false;
    while(!found){
      anyStorage(nest, nest.name, 'scalpel')
        .then(next_nest_name =>{
          if(next_nest_name == nest.name){
            found = true;
            resolve("found it!");
          }
        })
        .catch(err => {
          reject("did not find it");
        })
    }
  })
}
```

```
let p1 = new Promise((resolve, reject) => {
  let v = Math.random();
  if(v >= .25){
    resolve("Success")
  }else{
    reject("Failed")
  }
})

let p2 = new Promise((resolve, reject) => {
  let v = Math.random();
  if(v >= .25){
    resolve("Success")
  }else{
    reject("Failed")
  }
})

let p3 = new Promise((resolve, reject) => {
```

```
    let v = Math.random();
    if(v >= .25){
      resolve("Success")
    }else{
      reject("Failed")
    }
  })
})

let arrayOfPromises = [p1, p2, p3];

function promise_all(arrayOfPromises){
  let returnValues = [];
  for (let index = 0; index < arrayOfPromises.length; ++index) {
    const element = arrayOfPromises[index];
    elementValue = null;
    element.then((message) =>{
      returnValues.push(message);
    }).catch(() =>{
      return null;
    })
  }
}

promise_all(arrayOfPromises);
```

```
<!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>Static Template</title>
  </head>
  <body>
    <form id="contact">
      <label>name</label><br />
      <input type="text" id="name" /><br />
      <label>email</label><br />
      <input type="email" id="email" /> <br />
      <label>Superhero Alias</label><br />
      <input type="text" id="superheroAlias" /><br />
      <button type="submit">Submit</button>
    </form>

    <script src="form.js">
  </script>
</body>
</html>
```

```
console.log("The page was loaded");
const url =
  "https://classdemo.amerabyte.net/Sample_Code/BIT_285-286-
  Summer2021/AJAX/process.php";
const nameEL = document.getElementById("name");
const emailEL = document.getElementById("email");
const shaEL = document.getElementById("superheroAlias");
const formEL = document.getElementById("form");

async function onSubmit(event) {
  event.preventDefault();
  const formData = new FormData();
  formData.append("name", name.value);
  formData.append("email", email.value);
  formData.append("superheroAlias", superheroAlias.vale);
  const response = await fetch(url, {
    method: "POST",
    body: formData,
  });
  const jsonData = await response.json();
  console.log(jsonData);
  if (jsonData.success === false) {
    alert(JSON.stringify(jsonData.errors.email));
  } else if (jsonData.success === true) {
    alert("success");
  } else {
    console.log("something else happend");
  }
}
formEL.addEventListener("submit", onSubmit);
```

PE10

► PE10

```
const loginSectionEL = document.getElementById("login") as HTMLDivElement;
const loginFormEL = document.getElementById("loginForm") as HTMLFormElement;
const loginFormName = document.getElementById("name") as HTMLInputElement;
const nameGoesHereEL = document.getElementById("nameGoesHere") as HTMLHeadElement;
const foodPriceEL = document.getElementById("foodPriceSpan") as HTMLSpanElement;
const carPriceEL = document.getElementById("carPriceSpan") as HTMLSpanElement;
const productsEL = document.getElementById("products") as HTMLDivElement;
const bodyEL = document.body;
const totalEl = document.getElementById("total") as HTMLHeadingElement;
const shopEL = document.getElementById("shop") as HTMLDivElement;
const navigateProductButton = document.getElementById("navigateProduct") as
HTMLButtonElement;
const navigateShopButton = document.getElementById("navigateShop") as
HTMLButtonElement;
const updateTaxButton = document.getElementById("updateTaxRate") as
HTMLButtonElement;
```

```
const taxRateEL = document.getElementById("taxRate") as HTMLInputElement;
const foodQuantityInputEL = document.getElementById("foodStock") as
HTMLInputElement;
const foodPriceInputEL = document.getElementById("foodPrice") as HTMLInputElement;
const carQuantityInputEL = document.getElementById("carStock") as
HTMLInputElement;
const carPriceInputEL = document.getElementById("carPrice") as HTMLInputElement;

const div1 = document.getElementById("div1") as HTMLDivElement;

let userName:string = "";
let taxRate = 0.01;

let cart: any = {
  "food":{
    price: 5,
    quantity: 0,
    stock: 10
  },
  "car":{
    price: 5000,
    quantity: 0,
    stock: 15
  }
};

function navigateProduct(){
  loginSectionEL.style.display = "none";
  shopEL.style.display = "none";
  nameGoesHereEL.textContent = "Users Name: " + userName;
  foodPriceEL.textContent = cart.food.price;
  carPriceEL.textContent = cart.car.price;
  totalEl.style.display = "block";
  calculate();
  productsEL.style.display = "block";
}

function navigateShop(){
  shopEL.style.display = "block";
  productsEL.style.display = "none";
  taxRateEL.value = taxRate.toString();

  foodPriceInputEL.value = cart.food.price.toString();
  carPriceInputEL.value = cart.car.price.toString();
  foodQuantityInputEL.value = cart.food.stock.toString();
  carQuantityInputEL.value = cart.car.stock.toString();
}

function calculate(){
  let total = 0;
  for(let product in cart){
    total += cart[product].price * cart[product].quantity
  }
  total = (1+taxRate)*total;
  totalEl.textContent = "Total: " + total.toFixed(2)
```

```
}

function addToCart(e: PointerEvent): any{
    console.log(e.target);
    const clickedElement = e.target as HTMLElement;
    if(clickedElement.id === "navigateShop"){
        navigateShop();
        return;
    }
    if(clickedElement.id === "navigateProducts"){
        navigateProduct();
        return;
    }
    if(clickedElement.id === "updateTaxRate"){
        taxRate = +taxRateEL.value;
        if(carPriceInputEL.valueAsNumber < 5000){
            alert("The price of the car is too low");
            return;
        }
        cart.food.price = +foodPriceInputEL.value;
        cart.food.stock = +foodQuantityInputEL.value;
        cart.car.price = +carPriceInputEL.value;
        cart.car.stock = +carQuantityInputEL.value;
        return;
    }

    if(clickedElement.classList.contains("addToCart")){
        const dataType = clickedElement.getAttribute("data-type") as string;
        if(cart[dataType].stock == 0){
            alert("Sorry, we do not have that much");
            return;
        }else{
            cart[dataType].stock -=1;
            cart[dataType].quantity+=1;
            console.log(dataType+"Quantity");
            (document.getElementById(dataType+"Quantity") as any).textContent =
cart[dataType].quantity;
            calculate();
            console.log(cart);
            console.log(cart[dataType].stock, cart[dataType].quantity);
        }
    }

    if(clickedElement.classList.contains("removeFromCart")){
        const dataType = clickedElement.getAttribute("data-type") as string;
        if(cart[dataType].quantity <= 0){
            alert("It is impossible to remove nothing from a cart");
            return;
        }else{
            cart[dataType].stock +=1;
            cart[dataType].quantity-=1;
            (document.getElementById(dataType+"Quantity") as any).textContent =
cart[dataType].quantity;
            calculate();
        }
    }
}
```

```

        console.log(cart[dataType].stock, cart[dataType].quantity);
    }
}

function hideLoginFormSection(e: Event){
    e.preventDefault();
    userName = loginFormName.value;
    navigateProduct();
}

loginFormEL.addEventListener("submit", hideLoginFormSection);

bodyEL.addEventListener("click", addToCart as any);

navigateProductButton.addEventListener("click", navigateProduct);
navigateShopButton.addEventListener("click", navigateShop)

```

```

<!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" />
    <style>
      #products, #total, #shop{
        display: none;
      }
    </style>
    <title>Document</title>
  </head>
  <body>
    <section id="login">
      <form id="loginForm">
        <label for="name">Name:</label>
        <input type="text" id="name" />

        <label for="password">Password:</label>
        <input type="password" id="password" />
        <button type="submit">Login</button>
      </form>
    </section>

    <section id="products">
      <h1 id="nameGoesHere"></h1>
      <button id="navigateShop">Go to shop page</button>
      <p>
        Food
        <button class="addToCart" data-type="food">Add to cart</button>
        <button class="removeFromCart" data-type="food">Remove from

```



```

    cart</button>
      Price: <span id="foodPriceSpan"></span>
      Quantity:
        <span id="foodQuantity">
          0
        </span>
    </p>
    <p>
      Car
      <button class="addToCart" data-type="car">Add to cart</button>
      <button class="removeFromCart" data-type="car">Remove from
    cart</button>
      Price: <span id="carPriceSpan">0</span>
      Quantity:
        <span id="carQuantity">
          0
        </span>
    </p>

    <h2 id="total">Total: </h2>
  </section>
<section id="shop">
  <h1>Shop</h1>
  <label for="taxRate">Tax Rate</label>
  <input type="number" id="taxRate" step="0.01"> <br>

  <label for="foodQuantity">Food Quantity</label>
  <input type="number" id="foodStock" step="1"> <br>
  <label for="foodPrice">Food Price</label>
  <input type="number" id="foodPrice" step="0.01"> <br>

  <label for="carQuantity">Car Quantity</label>
  <input type="number" id="carStock" step="1"> <br>
  <label for="carPrice">car Price</label>
  <input type="number" id="carPrice" step="0.01"> <br>

  <button id="updateTaxRate">Update Values</button><br>

  <button id="navigateProducts">Go to product page</button>
</section>

  <script src="index.js"></script>
</body>
</html>

```