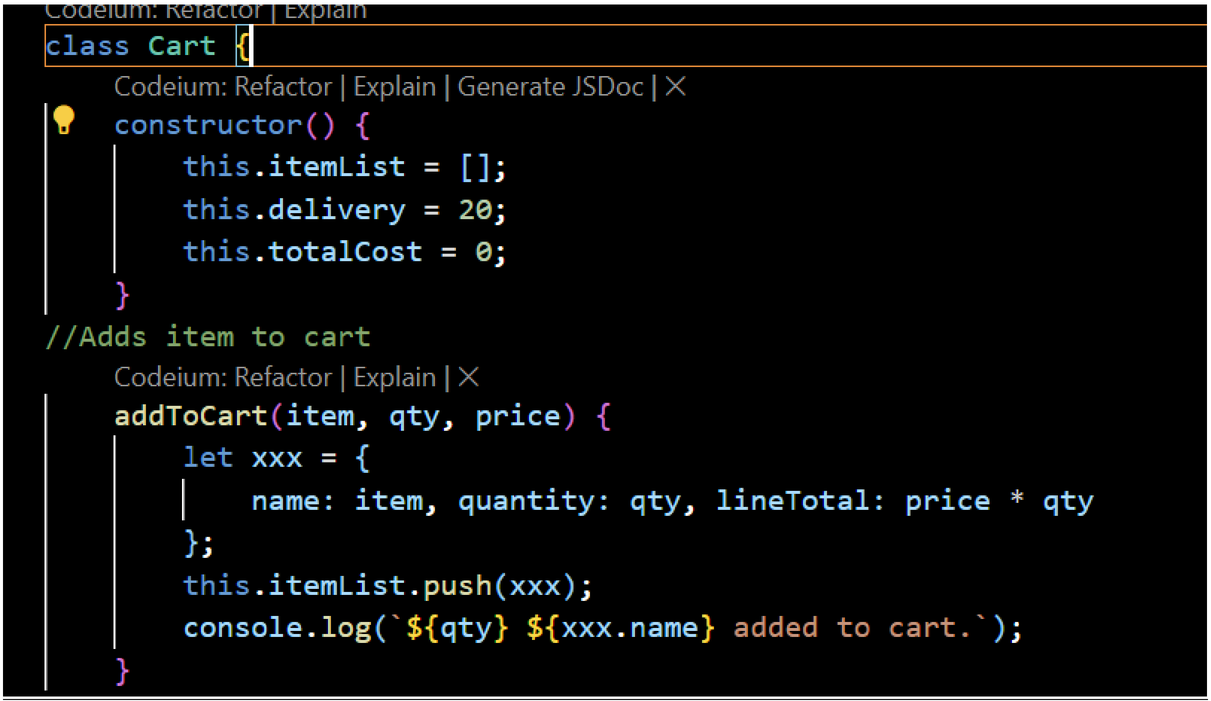
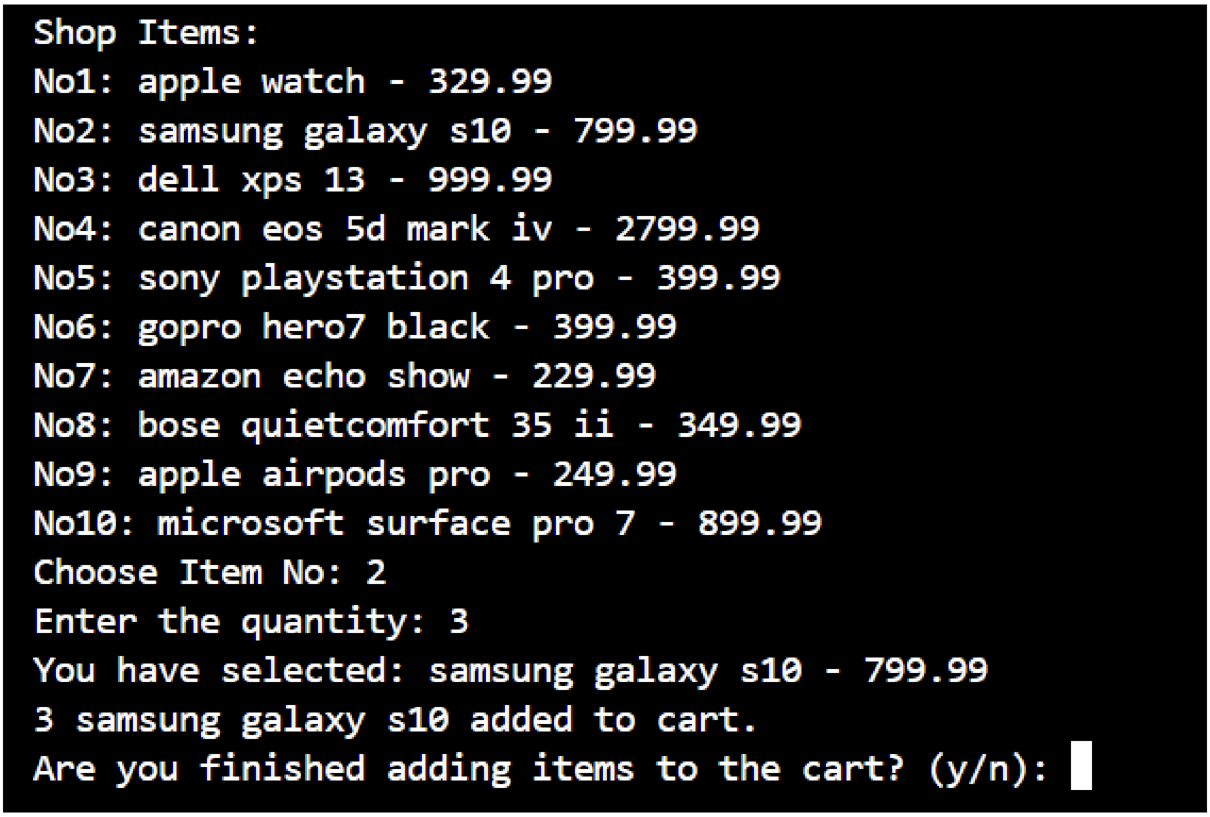
JavaScript Shop

Add To Cart:


Codeium: Refactor | Explain
Codeium: Refactor | Explain | Generate JSDoc | ✕
Codeium: Refactor | Explain | ✕

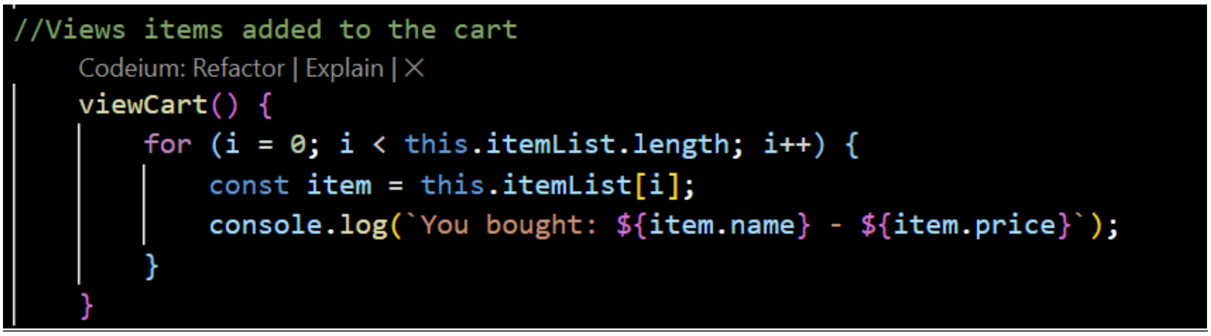
```
class Cart {
    constructor() {
        this.itemList = [];
        this.delivery = 20;
        this.totalCost = 0;
    }
//Adds item to cart
    addToCart(item, qty, price) {
        let xxx = {
            name: item, quantity: qty, lineTotal: price * qty
        };
        this.itemList.push(xxx);
        console.log(`${qty} ${xxx.name} added to cart.`);
    }
```

This method adds an Item to cart based on what the user selects as shown below:



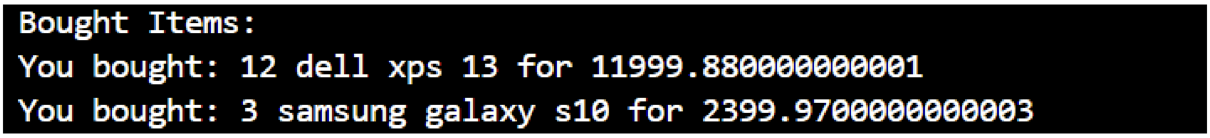
```
Shop Items:
No1: apple watch - 329.99
No2: samsung galaxy s10 - 799.99
No3: dell xps 13 - 999.99
No4: canon eos 5d mark iv - 2799.99
No5: sony playstation 4 pro - 399.99
No6: gopro hero7 black - 399.99
No7: amazon echo show - 229.99
No8: bose quietcomfort 35 ii - 349.99
No9: apple airpods pro - 249.99
No10: microsoft surface pro 7 - 899.99
Choose Item No: 2
Enter the quantity: 3
You have selected: samsung galaxy s10 - 799.99
3 samsung galaxy s10 added to cart.
Are you finished adding items to the cart? (y/n):
```

View Cart



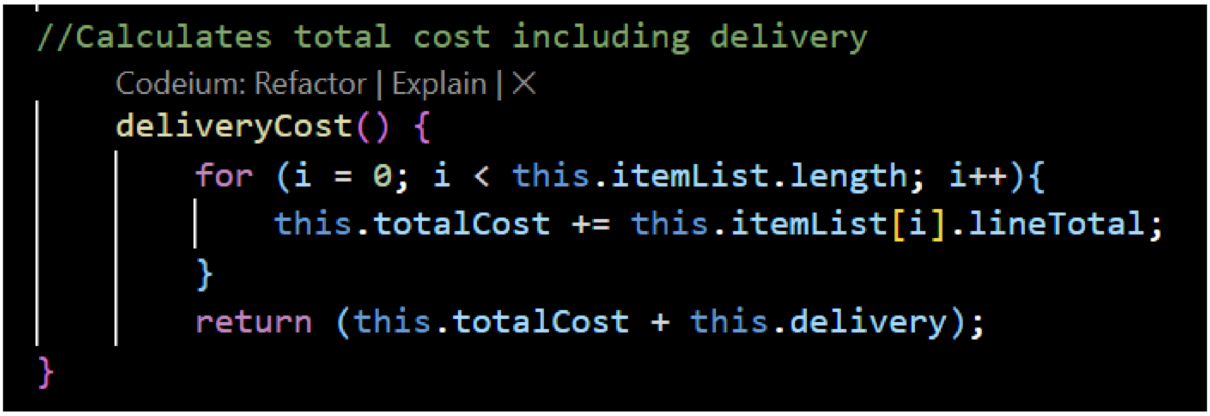
```
//Views items added to the cart
    viewCart() {
        for (i = 0; i < this.itemList.length; i++) {
            const item = this.itemList[i];
            console.log(`You bought: ${item.name} - ${item.price}`);
        }
    }
```

This method allows shows the items that a user has already added into their cart as shown below:



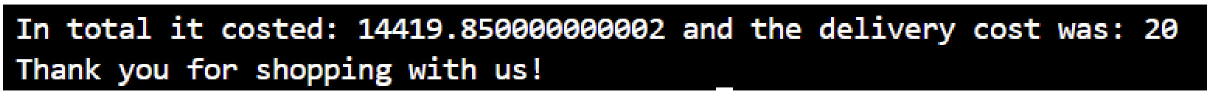
```
Bought Items:
You bought: 12 dell xps 13 for 11999.880000000001
You bought: 3 samsung galaxy s10 for 2399.9700000000003
```

Delivery Cost



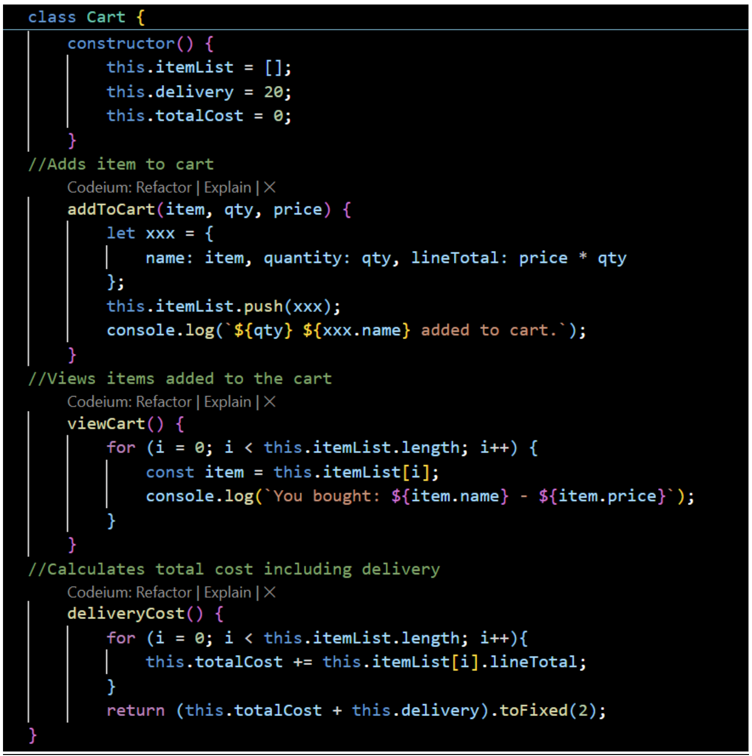
```
//Calculates total cost including delivery
    deliveryCost() {
        for (i = 0; i < this.itemList.length; i++){
            this.totalCost += this.itemList[i].lineTotal;
        }
        return (this.totalCost + this.delivery);
}
```

This method calculates the cost of the items will be, including the delivery charge as shown below:

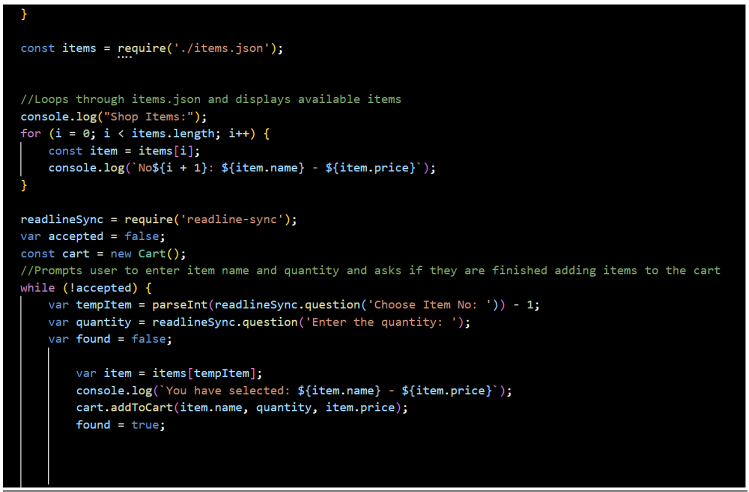


```
In total it costed: 14419.850000000002 and the delivery cost was: 20
Thank you for shopping with us!
```

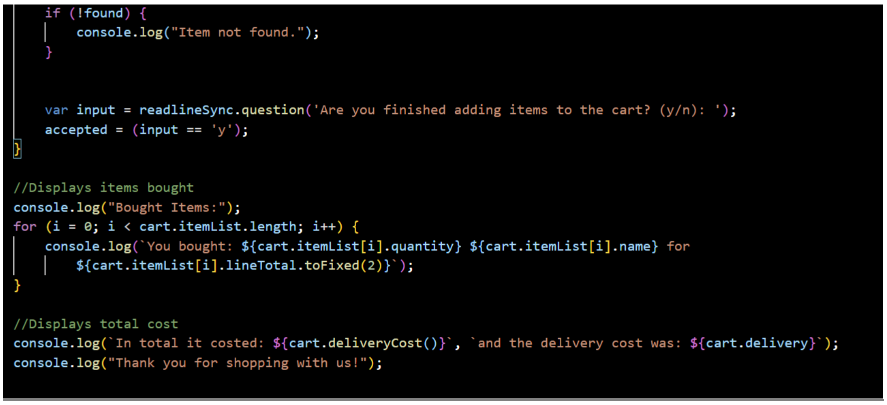
Full Code:



```
class Cart {
    constructor() {
        this.itemList = [];
        this.delivery = 20;
        this.totalCost = 0;
    }
//Adds item to cart
    Codeium: Refactor | Explain | ×
    addToCart(item, qty, price) {
        let xxx = {
            name: item, quantity: qty, lineTotal: price * qty
        };
        this.itemList.push(xxx);
        console.log(`${qty} ${xxx.name} added to cart.`);
    }
//Views items added to the cart
    Codeium: Refactor | Explain | ×
    viewCart() {
        for (i = 0; i < this.itemList.length; i++) {
            const item = this.itemList[i];
            console.log(`You bought: ${item.name} - ${item.price}`);
        }
    }
//Calculates total cost including delivery
    Codeium: Refactor | Explain | ×
    deliveryCost() {
        for (i = 0; i < this.itemList.length; i++){
            this.totalCost += this.itemList[i].lineTotal;
        }
        return (this.totalCost + this.delivery).toFixed(2);
}
```

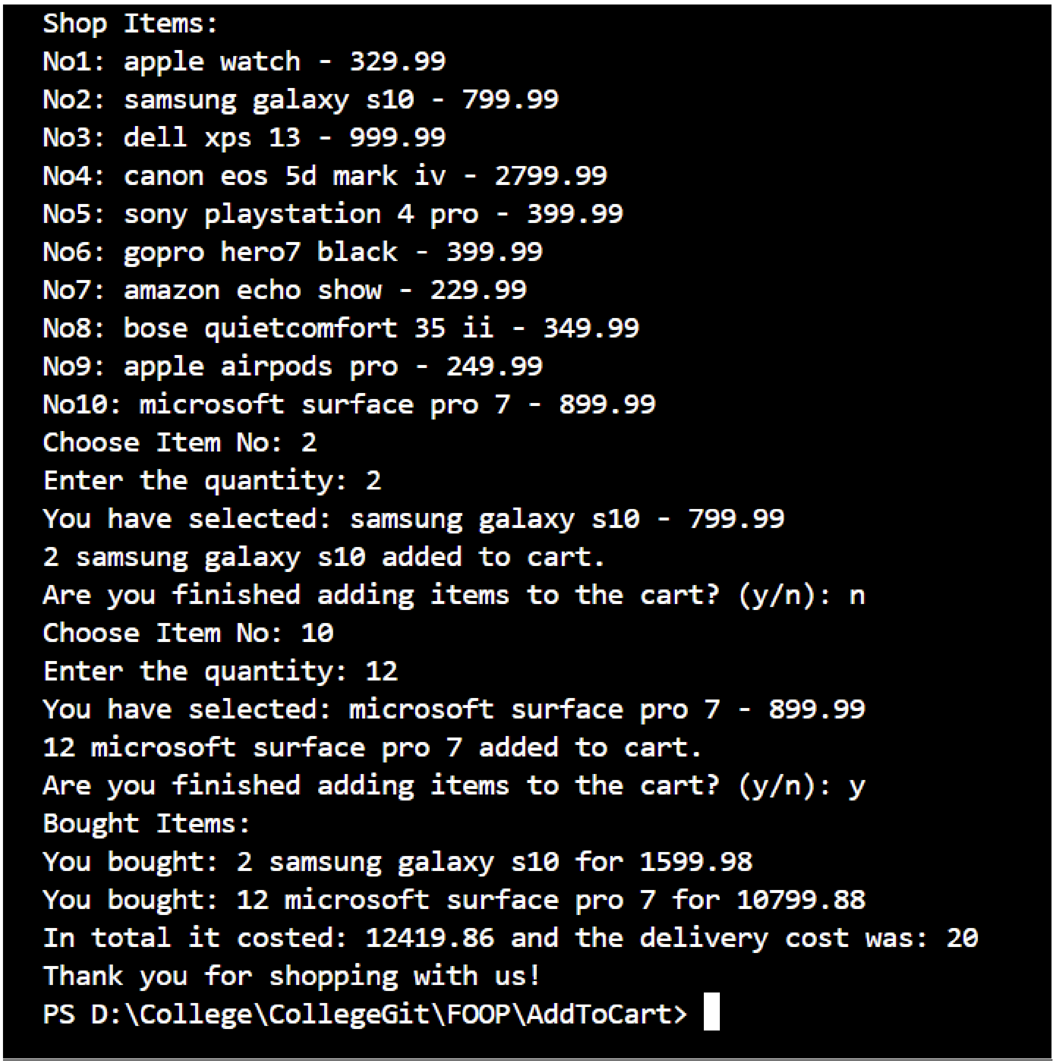


```
}

const items = require('./items.json');


//Loops through items.json and displays available items
console.log("Shop Items:");
for (i = 0; i < items.length; i++) {
    const item = items[i];
    console.log(`No${i + 1}: ${item.name} - ${item.price}`);
}

readlineSync = require('readline-sync');
var accepted = false;
const cart = new Cart();
//Prompts user to enter item name and quantity and asks if they are finished adding items to the cart
while (!accepted) {
    var tempItem = parseInt(readlineSync.question('Choose Item No: ')) - 1;
    var quantity = readlineSync.question('Enter the quantity: ');
    var found = false;

        var item = items[tempItem];
        console.log(`You have selected: ${item.name} - ${item.price}`);
        cart.addToCart(item.name, quantity, item.price);
        found = true;
```



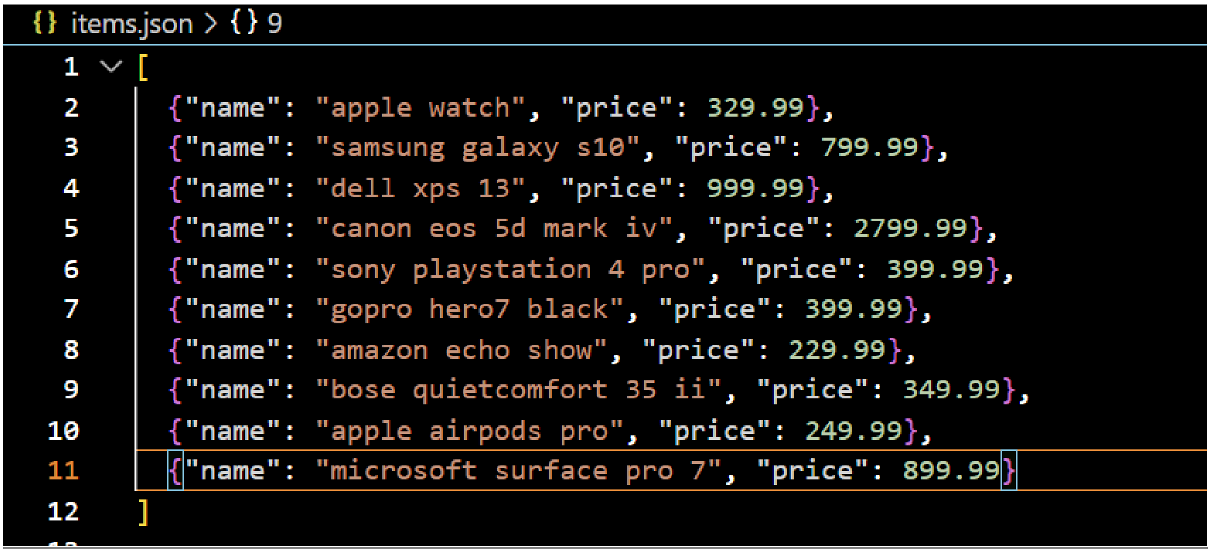
```
    if (!found) {
        console.log("Item not found.");
    }


    var input = readlineSync.question('Are you finished adding items to the cart? (y/n): ');
    accepted = (input == 'y');
}

//Displays items bought
console.log("Bought Items:");
for (i = 0; i < cart.itemList.length; i++) {
    console.log(`You bought: ${cart.itemList[i].quantity} ${cart.itemList[i].name} for
        ${cart.itemList[i].lineTotal.toFixed(2)}`);
}

//Displays total cost
console.log(`In total it costed: ${cart.deliveryCost()}`, `and the delivery cost was: ${cart.delivery}`);
console.log("Thank you for shopping with us!");
```

Full Output:



```
Shop Items:
No1: apple watch - 329.99
No2: samsung galaxy s10 - 799.99
No3: dell xps 13 - 999.99
No4: canon eos 5d mark iv - 2799.99
No5: sony playstation 4 pro - 399.99
No6: gopro hero7 black - 399.99
No7: amazon echo show - 229.99
No8: bose quietcomfort 35 ii - 349.99
No9: apple airpods pro - 249.99
No10: microsoft surface pro 7 - 899.99
Choose Item No: 2
Enter the quantity: 2
You have selected: samsung galaxy s10 - 799.99
2 samsung galaxy s10 added to cart.
Are you finished adding items to the cart? (y/n): n
Choose Item No: 10
Enter the quantity: 12
You have selected: microsoft surface pro 7 - 899.99
12 microsoft surface pro 7 added to cart.
Are you finished adding items to the cart? (y/n): y
Bought Items:
You bought: 2 samsung galaxy s10 for 1599.98
You bought: 12 microsoft surface pro 7 for 10799.88
In total it costed: 12419.86 and the delivery cost was: 20
Thank you for shopping with us!
PS D:\College\CollegeGit\FOOP\AddToCart>
```

JSON Data

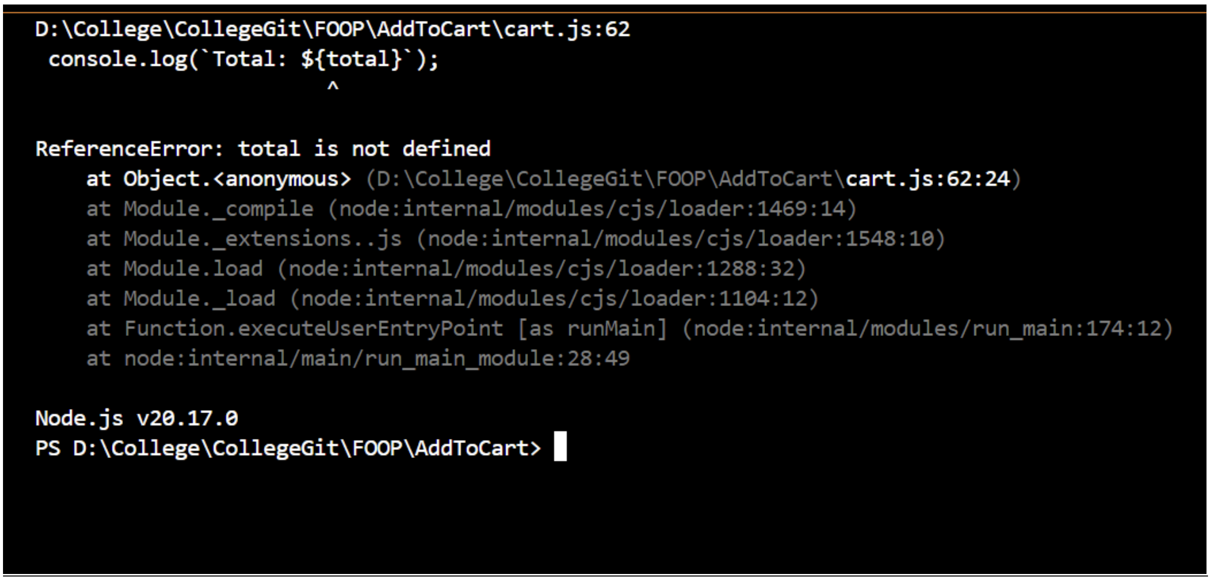


```json
[
  {"name": "apple watch", "price": 329.99},
  {"name": "samsung galaxy s10", "price": 799.99},
  {"name": "dell xps 13", "price": 999.99},
  {"name": "canon eos 5d mark iv", "price": 2799.99},
  {"name": "sony playstation 4 pro", "price": 399.99},
  {"name": "gopro hero7 black", "price": 399.99},
  {"name": "amazon echo show", "price": 229.99},
  {"name": "bose quietcomfort 35 ii", "price": 349.99},
  {"name": "apple airpods pro", "price": 249.99},
  {"name": "microsoft surface pro 7", "price": 899.99}
]
```

<u>Error Documentation</u>

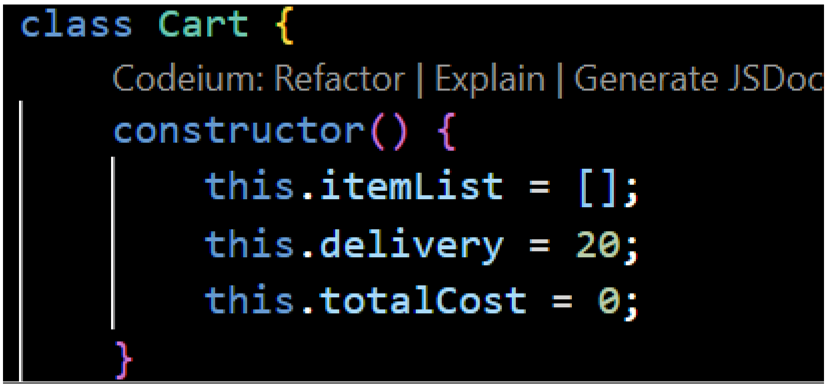
<u>Problem:</u>

The Total Cost as not being calculated, resulting in failure when being searched for.



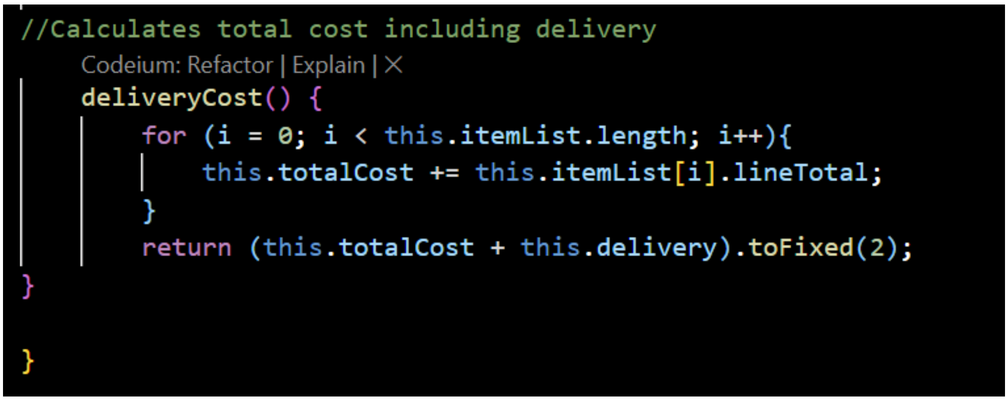
```
D:\College\CollegeGit\FOOP\AddToCart\cart.js:62
 console.log(`Total: ${total}`);
                     ^

ReferenceError: total is not defined
    at Object.<anonymous> (D:\College\CollegeGit\FOOP\AddToCart\cart.js:62:24)
    at Module._compile (node:internal/modules/cjs/loader:1469:14)
    at Module._extensions..js (node:internal/modules/cjs/loader:1548:10)
    at Module.load (node:internal/modules/cjs/loader:1288:32)
    at Module._load (node:internal/modules/cjs/loader:1104:12)
    at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:174:12)
    at node:internal/main/run_main_module:28:49

Node.js v20.17.0
PS D:\College\CollegeGit\FOOP\AddToCart>
```

<u>Solution:</u>



```
class Cart {
    constructor() {
        this.itemList = [];
        this.delivery = 20;
        this.totalCost = 0;
    }
```

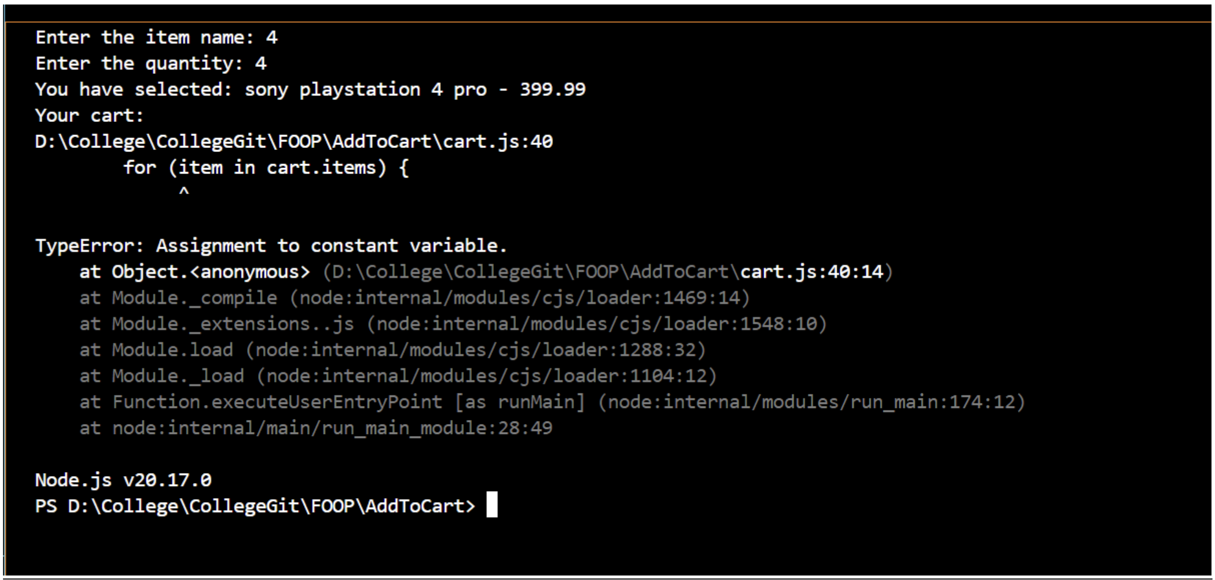
Adding the totalCost variable to the constructor to retrieve from later fixed this issue as I can now return this cost after calculating:



```
//Calculates total cost including delivery
    deliveryCost() {
        for (i = 0; i < this.itemList.length; i++){
            this.totalCost += this.itemList[i].lineTotal;
        }
        return (this.totalCost + this.delivery).toFixed(2);
}

}
```

Problem:

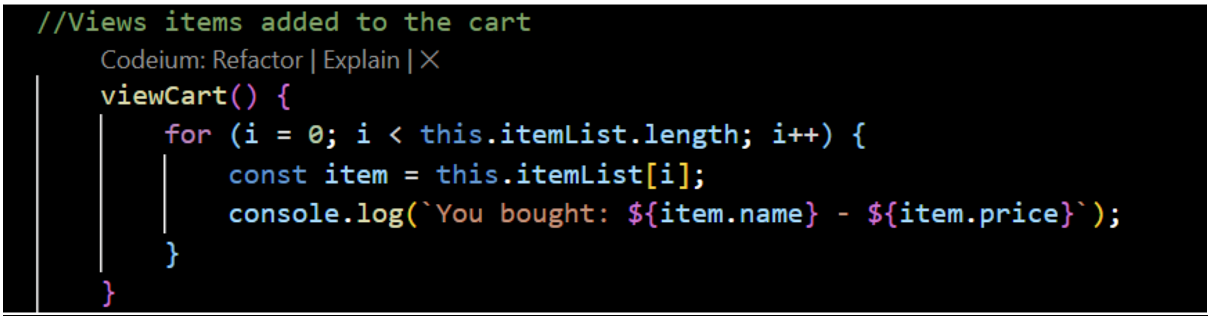
The for loop was not searching through the cart.items array.



```
Enter the item name: 4
Enter the quantity: 4
You have selected: sony playstation 4 pro - 399.99
Your cart:
D:\College\CollegeGit\FOOP\AddToCart\cart.js:40
        for (item in cart.items) {
             ^

TypeError: Assignment to constant variable.
    at Object.<anonymous> (D:\College\CollegeGit\FOOP\AddToCart\cart.js:40:14)
    at Module._compile (node:internal/modules/cjs/loader:1469:14)
    at Module._extensions..js (node:internal/modules/cjs/loader:1548:10)
    at Module.load (node:internal/modules/cjs/loader:1288:32)
    at Module._load (node:internal/modules/cjs/loader:1104:12)
    at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:174:12)
    at node:internal/main/run_main_module:28:49

Node.js v20.17.0
PS D:\College\CollegeGit\FOOP\AddToCart>
```

Solution:

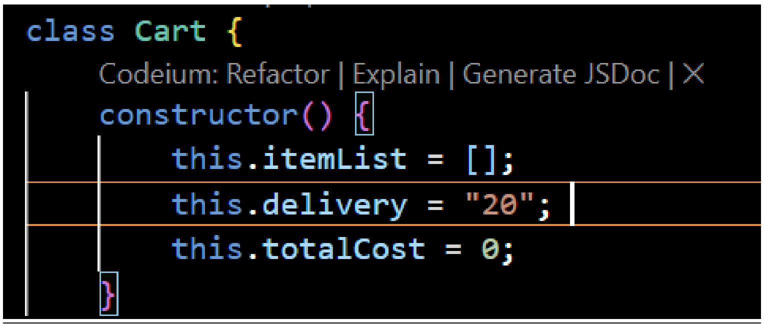
Creating a Constructor ItemList array & searching the length while logging the name & price of each item within the array on the loop. Using a “for each” instead of a “for in”.



```
//Views items added to the cart
    Codeium: Refactor | Explain | ×
    viewCart() {
        for (i = 0; i < this.itemList.length; i++) {
            const item = this.itemList[i];
            console.log(`You bought: ${item.name} - ${item.price}`);
        }
    }
```

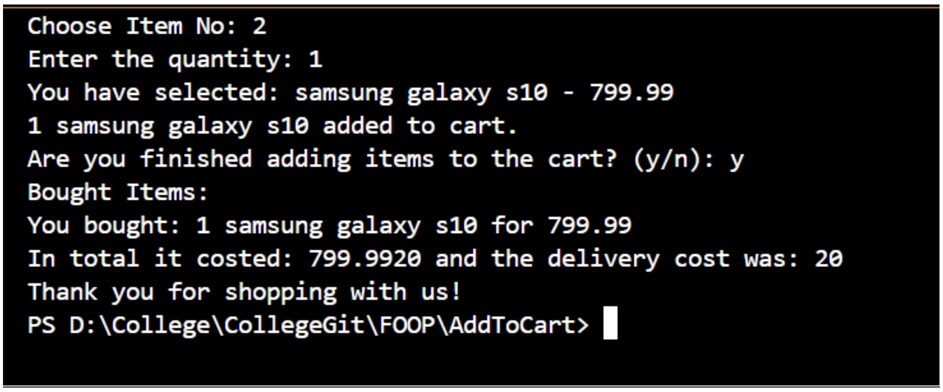
Semantic Error

Integer was set incorrectly as a String.



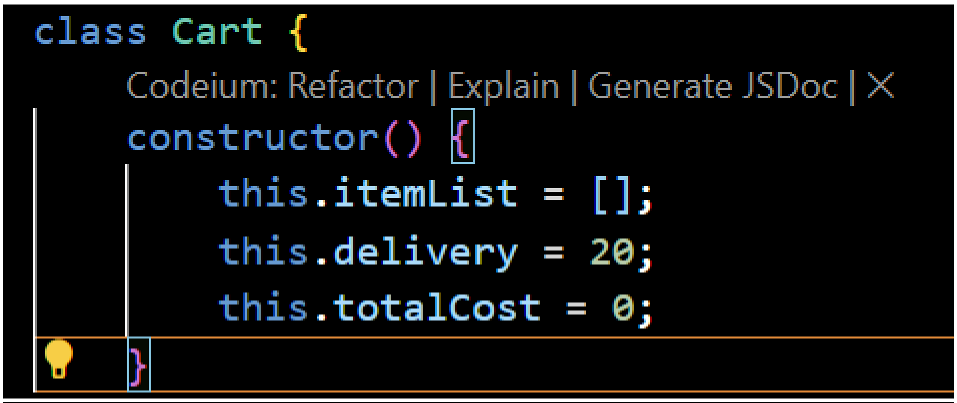
```
class Cart {
    constructor() {
        this.itemList = [];
        this.delivery = "20";
        this.totalCost = 0;
    }
```

This caused the total to be 799.9920 instead of 799.99+20 = 819.99



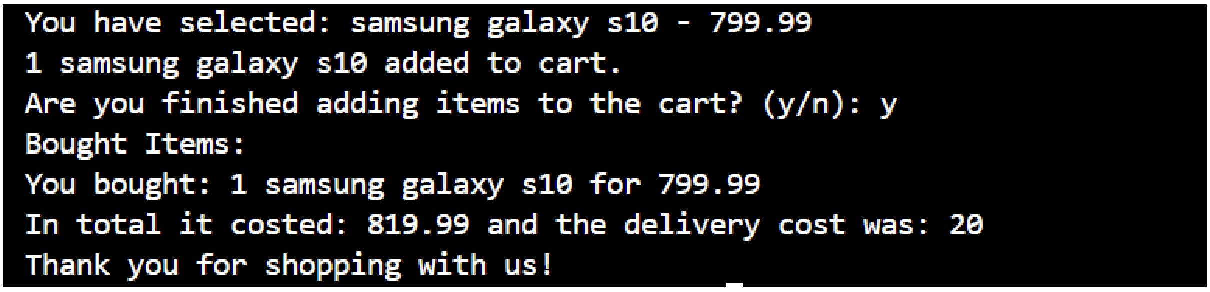
```
Choose Item No: 2
Enter the quantity: 1
You have selected: samsung galaxy s10 - 799.99
1 samsung galaxy s10 added to cart.
Are you finished adding items to the cart? (y/n): y
Bought Items:
You bought: 1 samsung galaxy s10 for 799.99
In total it costed: 799.9920 and the delivery cost was: 20
Thank you for shopping with us!
PS D:\College\CollegeGit\FOOP\AddToCart>
```

Solution:



```
class Cart {
    constructor() {
        this.itemList = [];
        this.delivery = 20;
        this.totalCost = 0;
    }
```

Changing the String to an Integer allowed correct calculations.



```
You have selected: samsung galaxy s10 - 799.99
1 samsung galaxy s10 added to cart.
Are you finished adding items to the cart? (y/n): y
Bought Items:
You bought: 1 samsung galaxy s10 for 799.99
In total it costed: 819.99 and the delivery cost was: 20
Thank you for shopping with us!
```