

CP1295 Advanced JavaScript Assignment 01 v1



CP1295 Assignment 01

Contents

A. Brief Description	3
B. Initial Page.....	4
C. Box Car Form	5
D. Cargo Processing Form	6
E. Process_Cargo JavaScript	7
F. Submission Requirements	8
Page 1	8
Page 2	8
Page 3	8
Page 4	8
Page 5	8
Page 6	9
Page 7	9
G. Code Requirements	10
H. Grading Rubric	11

A. Brief Description

Assignment for the course are threaded.

First Assignment begins the thread.

Each subsequent component adds onto the previous assignment.

Since this is a first run for this assignment thread, there is no solution yet established.

I will be creating each of the 10 assignments and solutions as the course progresses.

I have created the specifications of the first assignment that should provide a solid thread path through all 10 assignments. As the assignments are assigned, it is hopeful that most of the code from the previous assignment will form much of the code for the next assignment.

The goal is to spend less time explaining the model and more time working on the solution.

The model for the assignments involves managing cargo for a rail system.

In Assignment 01 we will require a single web page form that will contain

- (1) data regarding a Box Car that will eventually be attached to a train.
- (2) data regarding new Cargo Items that will be shipped.
- (3) Output Display showing all of the cargo that is in the Box Car.

B. Initial Page

The screenshot shows a web form titled "CNA RAIL SYSTEM" in blue text. It contains two distinct form sections. The first section, for Box Car information, includes three text input fields: "Box Car ID:" (containing "BX500"), "Empty Weight:" (containing "15000"), and "Total Weight:" (containing "0"). Below these fields are two buttons: "Process Box Car" and "Reset Form". The second section, for cargo information, includes three text input fields: "Transport ID:", "Description:", and "Cargo Weight:". Below these fields are two buttons: "Process Cargo" and "Reset Form".

Initial Page will contain only two forms.

(1) Box Car form with three text fields and labels as shown here.

Box Car form has non-functional buttons

(3) Process Cargo Form with three text fields and labels as shown here.

Process Cargo Form's has two functional buttons

The index.html should only contain code for what is shown in the initial form.

C. Box Car Form

CNA Rail System has only ONE box car.

These are the details for the box car form of the page.

There are three fields on this form

- all fields are set to read only
- set initial values for each text field as indicated in the form below

CNA RAIL SYSTEM

Box Car ID:
Empty Weight:
Total Weight:

The two buttons are non functional for this assignment.

The empty weight signifies the empty weight of the Box Car.

The total Weight will be updated as the cargo is added to be box car by the cargo_processing JavaScript.

D. Cargo Processing Form

Cargo processing component of the page has three active fields.

Transport ID:

Description:

Cargo Weight:

The fields are filled out then 'Process Cargo' is clicked to activate the processing of the data by calling a JavaScript 'process_cargo.js'.

Reset Form will set all three fields to blank.

No Form Data Checking is required for this assignment.

Enter the 5 sets of data as shown in the spreadsheet extract below for processing.

After the entering of each set, click on 'Process Cargo'. This will call 'process_cargo.js'.

Transport ID	Description	Weight
TXL2031S01	50,000 Shirts	25000
TXL2031S02	30,000 Coats	30000
MED2033S01	Medical MX45000	16000
MED2033S02	Medical MX34111	16000
GSX2037S01	Lamp Oil K1 Drum	10000

E. Process_Cargo JavaScript

The script will perform the following tasks

CNA RAIL SYSTEM

Box Car ID:

Empty Weight:

Total Weight:

Transport ID:

Description:

Cargo Weight:

Cargo Box Car Manifest for Box Car XXXXX

Transport ID	Description	Cargo Weight
XXXXX	XXXXX	XXXXX
XXXXX	XXXXX	XXXXX
XXXXX	XXXXX	XXXXX
XXXXX	XXXXX	XXXXX
XXXXX	XXXXX	XXXXX
XXXXX	XXXXX	XXXXX
Total Cargo Weight:		XXXXX

(1) Create an output Label for Cargo Box Car Manifest for Box CarXXXXX,

(2) A Label is created “Car Box Car Manifest for Box Car. XXXXX will be replaced with BX500 by using a DOM selection technique.

(2) Table is created with

(a) Header as shown “Transport ID “

(b) Each of the data entries will be displayed where indicated by the “XXXXX” Place holders.

(c) Total Cargo Weight as shown. It is updated as the cargo data is entered.

(d) The ‘Total Weight’ in the Box Car form is updated as the cargo data is entered.

Technique to adding in the table data to the page is unrestricted for this assignment. You may use Document.write for this assignment .

F. Submission Requirements

Generate a word document called Assignment 02 – Your name and student number.

Page 1

- a. Add Course Number
- b. Your Name
- c. Your Student Number

Page 2

Screen Shot of initial FORM prior to any data entry

Page 3

Screen Shot of FIRST data entry after 'Process Cargo' button pressed.

Expect to observe 1 row of output with correct Total Cargo Weight calculations posted to output and to Box Car form for Total Weight.

Page 4

Screen shot of LAST data entry after 'Process Cargo' button pressed.

Expect to observe 5 rows of output with correct Total Cargo Weight calculations posted to output and to Box Car form for Total Weight.

Page 5

Copy and paste the code for your CSS documents. (Not screen shots)

Page 6 (or next blank page thereafter)

Copy and paste the code for your index.html (Not screen shots)

Page 7 (or next blank page thereafter)

Copy and paste the code for 'process_cargo.js'

G. Code Requirements

The assignments are based course outline requirements.

Code used for the completion of this assignment must follow the following guidelines.

Inclusions

(1) Code must be based on code demonstrated in this course or its pre-requisite course(s)

- Course Text Book

- Course Notes

- Course Handouts

Exclusions

(2) Code must follow the following exclusion rule(s)

- a. Note: `getElementById`, `getElementsByTagName`, `innerHTML`, `outerHTML` are not permitted in this course. Where DOM selection is required use only `document.querySelector(sel)` , `document.querySelectorAll(sel)` OR a jQuery selector.

H. Grading Rubric

To maximize your grade be sure to consider the Grading Rubric as part of the of list of requirements.

Items missed are marks that you will not receive.

A	Name, Student #, Assignment # on Page 1	5
B1	HTML file contained Box Car Form	2
B2	HTML file contained Process Cargo Form	2
B3	HTML FORM was LIMITED only to B1 and B2 contents	4
	BOX CAR FORM (BCM)	
C1	Box Car ID / Text Field RO default set BX500	4
C2	Empty Weight / Text Field RO default set 15000	4
C3	Total Weight / Text Field RO default set 0	4
C4	Process Box Car Non-Functional Button	2
C5	Reset Form Non-Functional Button	2
	Cargo Processing Form (CPF)	
D1	Transport ID / Text Field	2
D2	Description / Text Field	2
D3	Cargo Weight / Text Field	2
D4	Process Cargo Button (exists)	2
D5	Reset Form Button (Exists)	2
D6	Process Cargo Button - calls Java Script	5
D7	Reset Form Button - resets form to blanks	5
	Process Cargo JavaScript	
E1	All data fields read	5
E2	Output Header "Cargo Box ... created	5
E3	Table Header Generated "Transport ID" ...	5
E4	Data Row Created for First data entry	5
E5	Subsequent Data Rows created	5
E6	Total Cargo Weight - After First data row entered	5
E7	Total Cargo Weight - After All data rows entered	5
E8	Box Car Field - Total Weight - After First data row entered	5
E9	Box Car Field - Total Weight - After all data rows entered	5
	Submission	
FP2	Screen shot of initial form prior to data entry	1
FP3	Screen shot first data entry after 'Process Cargo" btn	1

CP1295 Advanced JavaScript – Assignment 01 – 3%

FP4	Screen shot first data entry after last 'Process Cargo" btn	1
FP5	CSS Code "not screen shots" in document	1
FP6	Index.html "not screen shots" in document	1
FP7	All java Script code "not screen shots" in document	1
		100

End of Assignment