



## Classification

(English from page 3)

Ennen kuin alat tehdä näitä harjoituksia, sinun tulee katsoa kaikki videot ja lukea web-linkit otsikon 'Before you start the exercises 2' alta.

Lataa Moodlesta *Choices2*-zip-tiedosto ja pura omalle työasemallesi. Avaa projekti NetBeansissa. Projekti sisältää yhden www-sivun, pari tyylitiedostoa ja javascript-tiedoston. Mukana on myös kirjoitetun ohjelmakoodin testaamiseen tarvittava osa. Tehtävänäsi on kirjoittaa vaadittu toiminta JavaScript-tiedoston funktioihin.

Avaa projekti NetBeansissa, ota esille tiedostot *index.html* ja *classification.js*. Kunkin tehtävän ratkaisun kirjoitat omaan funktioonsa. Avaa *index.html*-tiedosto myös selaimessa. Kun olet kirjoittanut ratkaisusi tehtävään, virkistä sivu selaimessa ja kokeile ratkaisiasi kirjoittamalla datan selaimen syöttökenttään ja klikkaamalla painiketta.

Älä muuta html- ja css-koodia. Esimerkkitulokset ovat tehtävien jälkeen.

Näissä tehtävissä sinun tulee tyhjentää tulostuselementit funktion alussa, jotta funktio toimii oikein. *Jos syöttötieto on väärin, niin sopiva virheilmoitus tulee tulostaa virhe-elementtiin.*

### V10: Grades - function v10()

Tentin arvosana perustuu pisteisiin (**points**), pisterajat oikealla taulukossa. Jos pistemäärä on taulukon ulkopuolella, niin sopiva virheilmoitus tulostetaan (**grade\_error**). Käytä funktioon määriteltyä vakiota virheilmoituksen tulostuksessa. Virhetilanteessa funktion toiminta lopetetaan. Pisteitä vastaava arvosana tulostetaan elementtiin **grade**.

points	grade
0-29	fail
30-34	satisfactory
35-39	fair
40-49	good
50-60	excellent

### V11: Clock- function v11()

Tervehdys perustuu kellotunteihin (**clock**) oikealla olevan taulukon mukaisesti. Jos kellotunnit ovat taulukon ulkopuolella, niin sopiva virheilmoitus tulostetaan (**clock\_error**). Käytä funktioon määriteltyä vakiota virheilmoituksen tulostuksessa. Virhetilanteessa funktion toiminta lopetetaan. Kellotuntia vastaava arvosana tulostetaan elementtiin **greeting**.

Clock hour	Greeting
21-24	Good night
18-20	Good evening
12-17	Good afternoon
4-11	Good morning
0-3	Good night

### V12: Shopping list - function v12()

Tarvitaan juhlien ostoslista. Ilmoittautuneiden osallistujien määrä vaikuttaa ostettavien ruokien ja juomien määrään sekä juhlatilan paikkaan. Perehdy funktioon määriteltyihin vakioihin.

Virhetilanteet:

- ~ vieraiden määrä on negatiivinen
- ~ vieraiden määrä on desimaaliluku
- ~ ilmoittautuneita vieraita on 3 tai vähemmän



Näin tutkit onko muuttujan sisältö kokonaisluku vai ei

```
// not an integer
if (Number.isInteger(guests) === false) {
```

Juhlapaikan osoite ja ruuan määrä perustuu vieraiden määrään

vieraiden määrä	osoite	ruuan ja juoman määrä / osallistuja	
4 - 15	Naapuritie 1	sausages: 3	drinks: 5
		tomatoes: 2	eggs: 1
16 - 50	Naapuritie 12	sausages: 2	drinks: 3
51 -	Juhlasalintie 34	tomatoes: 1	eggs: 0.5

Tulosta osoite (elementtiin **address**) ja ostoslista (elementtiin **shopping\_list**). Rivinvaihto tehdään sijoittamalla <br> tulostuslauseeseen. Esimerkkitulostuksia:

Guests:

Shopping list

Shopping list:  
- Sausages 36 pcs  
- Drinks 60 bottles  
- Tomatoes 24 pcs  
- Eggs 12 pcs

Naapuritie 1

Guests:

Shopping list

Shopping list:  
- Sausages 50 pcs  
- Drinks 75 bottles  
- Tomatoes 25 pcs  
- Eggs 12.5 pcs

Naapuritie 12

Guests:

Shopping list

Shopping list:  
- Sausages 120 pcs  
- Drinks 180 bottles  
- Tomatoes 60 pcs  
- Eggs 30 pcs

Juhlasalintie 34

## Classification

Before you start these exercises you should watch all the videos and read the web links under a title 'Before you start the exercises 2'.

Download from Moodle the zip file *Choices2*. Unzip it in your own computer. The file includes a NetBeans project. The project includes an html, a css and a JavaScript files. There is also a testing part for the project. Your task is to write program code in the JavaScript file.

Open the project in NetBeans, open the files *index.html* and *classification.js*. Each exercise solution is to be written in a different function. Open the *index.html* file also in a browser. After you have written your solution to an exercise you refresh the page in a browser and test your solution by inserting the data and clicking the button.

Please, do not change any html and css code.

These exercises require you to clear the output elements at the beginning of the function code to run properly. *If the input data is wrong there should be only a suitable error message printed.*

### V10: Grades - function v10()

Grades of an exam are based on the points (**points**) in the list on the right. If the inserted points are outside of the list an error message should be printed (**grade\_error**). Use predefined constant to print the error message. In the case of an error situation the operation of the function should stop. The corresponding grade is to be printed in the element **grade**.

points	grade
0-29	fail
30-34	satisfactory
35-39	fair
40-49	good
50-60	excellent

### V11: Clock- function v11()

A greeting is based on the clock hours (**clock**) in the list on the right. If the inserted clock hours are outside of the list an error message should be printed (**clock\_error**). Use predefined constant to print the error message. In the case of an error situation the operation of the function should stop. The corresponding greeting is to be printed in the element **greeting**.

Clock hour	Greeting
21-24	Good night
18-20	Good evening
12-17	Good afternoon
4-11	Good morning
0-3	Good night

### V12: Shopping list - function v12()

There is going to be a party. The number of enrolled guests affect the amount of the food and drinks to buy also to the room to hire for a party. Study the predefined constants of the exercise.

Error situations:

- ~ negative number of guests
- ~ decimal number inserted as number of guests
- ~ enrolled guests 3 or less

You can test whether a variable content is an integer or not

```
// not an integer
if (Number.isInteger(guests) === false) {
```

Party address and amount of food is based on the number of guests

number of guests	party address	amount of food and drink / guest	
4 - 15	Naapuritie 1	sausages: 3	drinks: 5
		tomatoes: 2	eggs: 1
16 - 50	Naapuritie 12	sausages: 2	drinks: 3
51 -	Juhlasalintie 34	tomatoes: 1	eggs: 0.5

Print address (element **address**) and shopping list (element **shopping\_list**). To get a line break in the print use <br> in the print statement. Example prints:

Guests:

Shopping list

Shopping list:  
 - Sausages 36 pcs  
 - Drinks 60 bottles  
 - Tomatoes 24 pcs  
 - Eggs 12 pcs

Naapuritie 1

Guests:

Shopping list

Shopping list:  
 - Sausages 50 pcs  
 - Drinks 75 bottles  
 - Tomatoes 25 pcs  
 - Eggs 12.5 pcs

Naapuritie 12

Guests:

Shopping list

Shopping list:  
 - Sausages 120 pcs  
 - Drinks 180 bottles  
 - Tomatoes 60 pcs  
 - Eggs 30 pcs

Juhlasalintie 34