**Control flow**

**DEADLINE:** 10/02/2019

**FOLDER STRUCTURE**

|  |  |
| --- | --- |
| FE\_9\_7\_homework\_control-flow/\*     └─ task/      └─ FE\_9\_7\_homework\_control-flow.docx  └─ homework/\*     └─ src/  └─ js/  └─ task1.js\*  └─ task2.js\*  └─ task1.html\*  └─ task2.html\* | \* ­­­- required |

**TASK**

## Task #1. Check the user

Write the code which verify user rights.

Step 1. Check login

* Ask user for a login // use prompt()
* If the input is an empty line or Esc – show “Canceled.” // for showing - use alert()
* If the input length less than 4 symbols - show “I don't know any users having name length less than 4 symbols”.
* If it’s another string – then show “I don’t know you”.
* If the visitor enters "User" or "Admin", then prompt for a password.

Step 2. Check password:

* For an empty string or cancelled input, show “Canceled.”
* For login “User” correct password is “UserPass”, for “Admin” correct password is “RootPass”. In other case, show “Wrong password”.

Step 3. Greets the user appropriately:

* If the current time in hours is less than 20: // current hours – new Date().getHours()
  + For “User” show “Good day, dear User!”
  + For “Admin” show “Good day, dear Admin!”
* If the current time in hours is more or equals 20
  + For “User” show “Good evening, dear User!”
  + For “Admin” show “Good evening, dear Admin!”

## Task #2. Guessing game

Your task is to write a game.

Requirements:

Step 1:

* Create a prompt window (use confirm()). Show the message inside the window ‘Do you want to play a game?’.
* In case the user clicks the 'Cancel' button, the message 'You did not become a millionaire, but can.' should be shown (use alert).

Step 2:

* If user clicked ‘Ok’ – start a game: randomly (use Math.random()) choose a number in range [0; 5] and ask user to enter a number. (use prompt()).
* User has 3 attempts to guess a number.
* If user guessed number on 1-st attempt prize is 10$ (maximum prize for current numbers range), 2-nd attempt – 5$, 3-rd attempt – 2$.
* If user did not guess a number show the message ‘Thank you for a game. Your prize is: …’ (Use alert) and ask if he wants to play again (use confirm).

Step 3:

* If user did guess - Show the message ‘Congratulation! Your prize is: … Do you want to continue?’.
* If user does not want to continue – show the message ‘Thank you for a game. Your prize is: …’ (Use alert) and ask if he wants to play again (use confirm).
* If user does want to continue, make number range twice as big as the previous one (for example [0; 5] -> [0; 10]), and three times bigger maximum prize (for example on 1-st attempt prize will be 30$, 2-nd attempt – 15$, 3-rd attempt – 7$). Prize must be added to the previous one and number of attempts should be set to 3 (user should have 3 attempts to guess a number for each numbers range)
* Each time you ask user to enter a number you should show him a range of numbers, how much attempts he has left, his total prize and possible prize on current attempt. See Figure 1:

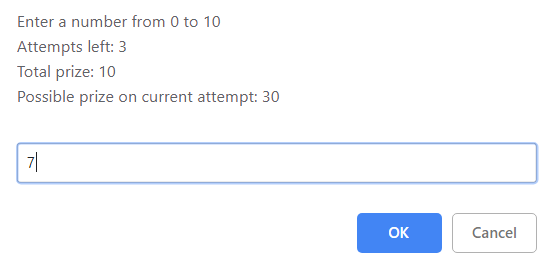


Figure 1 – The prompt window

* All these stuffs should be repeated until user lose or decide to quit

**BEFORE SUBMIT**

* In order to use npm package manager you should install nodejs (https://nodejs.org/ )
* Install eslint to check your code (npm install -g eslint)

- open a terminal(or cmd)

- go to *src* folder

- run eslint (i.e. eslint ./js/task1.js)

Code should be without ‘errors’

* Verify that all functionality is implemented according to requirements
* Format your code (remove redundant spaces, lines of code etc.)
* Add comments if necessary

**SUBMIT**

* The folder should be uploaded to github repository 'FLX' into master branch