

Web Applications Development: Javascript

2017

Allotted time: 90 minut. You have to send the solution *before* 15:50.
Points total: 25

1 Sending the solution

Pack all (.js i .css) files into zip and name it by replacing login with your lab login. `login.zip`.

Send email:

To: jan.karwowski@mini.pw.edu.pl

Subject: [WebApp] JS2 2017

Attachments: `login.zip`

Empty message body.

2 The task: city planning simulator

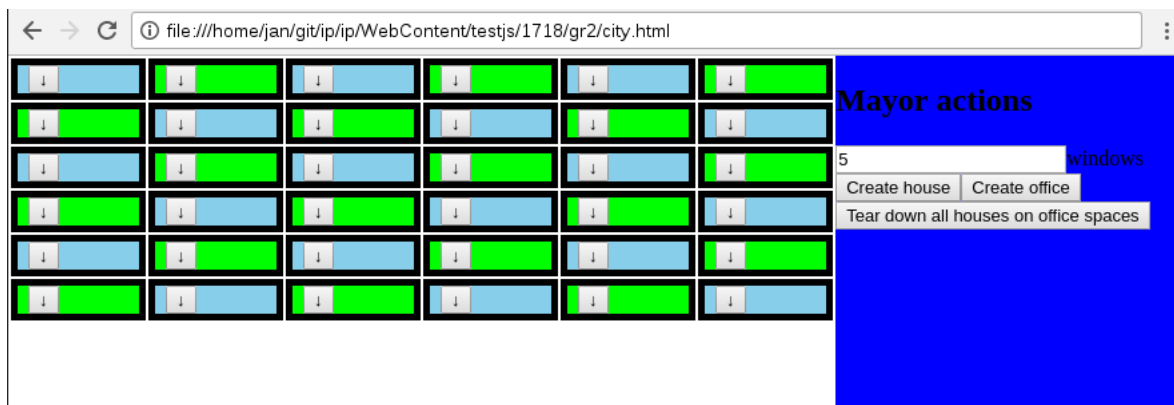
You have to develop simple application to aid planning of a city.

2.1 Common requirements

You have to send only your .css and .js files (no html, no jquery). Your solution will be checked against the original timers.html and has to work with it. If your solution will not work with the original html, it will be considered incorrect.

Creation of global variables in javascript code is strictly forbidden. **Use of global variables is strictly forbidden.**

2.2 Detailed requirements



Rysunek 1: The loaded page

- 1 point The page is divided to two areas white on the right, 70% wide and blue on the left. Both areas cover the whole height line in the Figure 1.
- 4 points When the page is loaded control buttons are created on the right panel and the left one is populated with 6×6 grid of rectangular city districts. District are colored in chessboard-like fashion. Blue districts are commercial ones and green are residential. Each district has a button with an arrow pointing down.
- 3 points When *create house* or *create office* is clicked a new building is created, house and office respectively. (The images that are needed are enclosed with the task). Newly created building are put in the top-leftmost



Rysunek 2: Adding three buildings: House (3 windows), House (0 windows), Office (5 windows).

district. The number input field by the buttons is used to define how many windows that can be opened are installed in the new building. This number is drawn on the building image. See Figure 2.

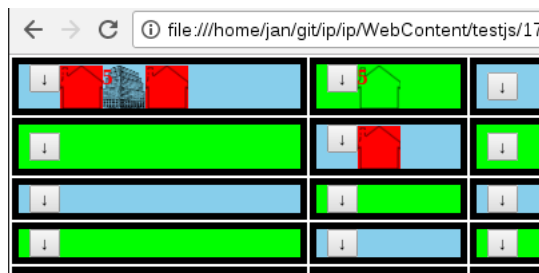
4. 2 points After clicking the building becomes selected (depicted with red border around it). Only one building can be selected at a time. If another one is selected, the previous one returns its state to normal. A selected building is presented in Figure 3.



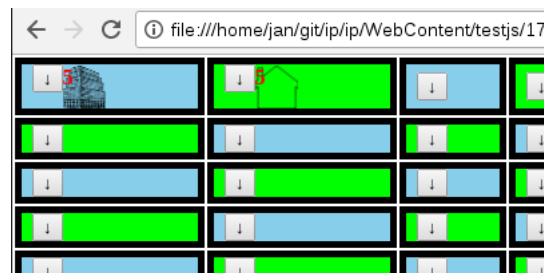
Rysunek 3: A building with 0 windows is selected.

5. 2 points Clicking a button within a district causes the selected building to be moved to that district. The buildings becomes not selected. If there is no selected building nothing happens.
6. 4 points Implement a planing aid for a „corporate-friendly city” approach. A button labeled *Tear down all houses on office spaces* destroys all buildings that are currently on commercial (blue) districts. The process is as follows:
 1. Change background of all buildings to destroy to red.
 2. Wait three seconds.
 3. Remove those buildings from the documents.

Until the operation completes the button *Tear down all houses on office spaces* is inactive. The process is visualized in Figure 4.



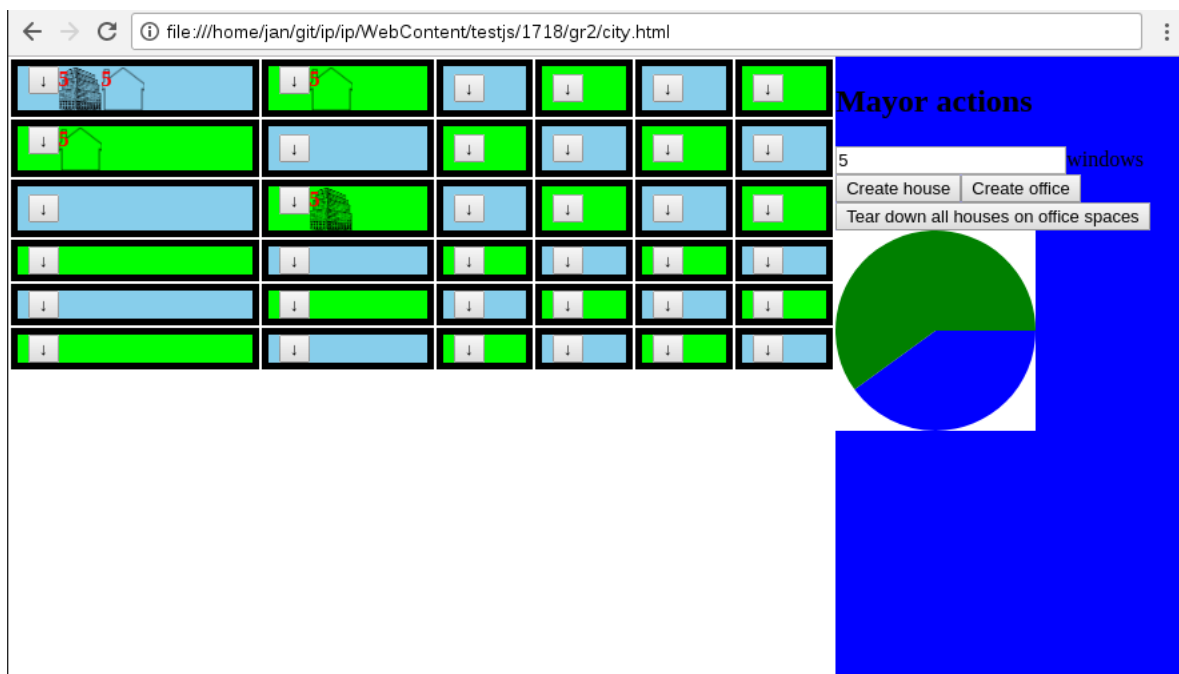
(a) Just after pressing the button.



(b) After three seconds.

Rysunek 4: A process of tearing down houses in commercial district.

7. 2 points A building with red background (during tear-down) can be rescued by moving a mouse cursor over it while holding shift key. The red background is removed and building won't be destroyed.
8. 5 points Under a controls on the right a pie chart appears. The chart shows a portion of office buildings in the city (in blue) and houses (in green). The chart is updated every 5 seconds. Example situation is presented in Figure 5.
9. 2 points When there are more than 30 buildings in the city, *the system* breaks down and city state returns to the initial (empty) one.



Rysunek 5: A building count pie chart.