OIM3690 - Web Technologies

JavaScript - Form Validation

Processing form data

- check if `input` is blank
  - `.value.length` – the number of characters typed in the form `input`
  - Example:

```
if (myForm["userName"].value.length == 0){
  ...
}
// or if(myForm["userName"].value=="")
```

- check if the entered value is a number
  - `isNaN()`
  - Example: `isNaN(myForm["age"].value)`
    - returns `true` if the user entered a **non-numerical** value
    - returns `false` if the user typed a **numerical** value

Exercise: *ex16-1.html*

- Download *ex16-1.html* from GitHub (*OIM3690/resources/templates*) and write JavaScript
- When a user **clicks** the button, JavaScript will
  - capture the user inputs from the text fields and dropdown list
  - display all the inputs in `textarea`
  - check to make sure that the user **has entered something** in the text fields
  - check to make sure that the age value **is a numerical value**
- Can you **hide** age input when the `checkbox` (commented) is *checked*?

(Remote) Pair Programming

- **Driver**
  - typing code
  - sharing screen
- **Navigator**
  - paying close attention to the code
  - providing guidance and suggestions whenever possible
- Ideally, *Driver* and *Navigator* will switch roles

Exercise: *ex16-2.html* (pair programming)

- Creating a Celsius (C) - Fahrenheit (F) converter
  - create a form with **two text inputs** - one for F and the other for C
  - add one button `"CONVERT"` and one `"RESET"`
  - write a function that will convert one from the other, both ways
    - $F = C * (9/5) + 32$
    - $C = (F–32) * (5/9)$
  - validate to make sure that the user does not "*try anything funny*"
    - if the user does **not** enter any value, let the user know
    - if the user enters a **non-numeric** value, let the user know

Pair Programming

1. Switch *Driver* and *Navigator*
2. Write **pseudo-code** (8 mins)
    i. **DO NOT** start coding immediately
    ii. Write pseudo-code in Google doc playground
3. We will be back and summarize
4. Write **JavaScript code** in VS Code (10 mins)
    i. Copy pseudo-code into VS Code
    ii. Try to write JavaScript code based on pseudo-code

Benefits of (Remote) Pair Programming

1. Constant feedback
2. Reduced frustration
3. Increased focus
4. Social interaction
5. Accountability
6. Collaborative skills
7. Real-world experience
8. Mentorship

source: How Remote Pair Programming Works & Why it Can Change Your Life

Questions?