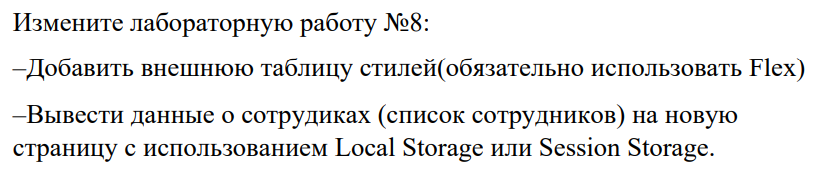
Лабораторная работа №10

Выполнил: Тарасенко Станислав

Цель работы: Использование объектной модели DOM

Код программы:

<body>

    <div class="container">

        <div>

            <div class="col-md-12 text-center">

                <h1 class="animate-charcter">Task: Write a script that, for a given date, determines how many days

                    passed after some event.

                    The questionnaire contains information about seven employees: last name, date

                    conclusion of a contract. Exactly eleven months after the start of work

                    Employees are granted leave of 24 working days.

                    Create a data entry form. Write a script that defines:

                    vacation start and end dates for each employee;

                    names of employees who will be granted leave in a given

                    month (month is selected by the user);

                    names of employees whose vacation will be completed in a given quarter

                    (number of the quarter is selected by the user);

                    names of employees who will be granted leave in the next

                    calendar year; Additional tasks: add external stylesheet (be sure to use Flex);

                    display employee data (list of employees) to a new

                    page using Local Storage or Session Storage.</h1>

            </div>

        </div>

    </div>

    <div id="newcon">

    <div id="top">

        <form>

            <div class="forma">

                <div class="cell">Number employee</div>

                <div class="cell">Second Name</div>

                <div class="cell">The date conclusion of a contract</div>

                <div class="cell">Vacation</div>

            </div>

        </form>

        <form [formGroup]="form1" class="form">

            <div class="forma">

                <div class="cell">Employee 1</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form1.value.vacation }}</div>

            </div>

        </form>

        <form [formGroup]="form2">

            <div class="forma">

                <div class="cell">Employee 2</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form2.value.vacation }}</div>

            </div>

        </form>

        <form [formGroup]="form3">

            <div class="forma">

                <div class="cell">Employee 3</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form3.value.vacation }}</div>

            </div>

        </form>

        <form [formGroup]="form4">

            <div class="forma">

                <div class="cell">Employee 4</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form4.value.vacation }}</div>

            </div>

        </form>

        <form [formGroup]="form5">

            <div class="forma">

                <div class="cell">Employee 5</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form5.value.vacation }}</div>

            </div>

        </form>

        <form [formGroup]="form6">

            <div class="forma">

                <div class="cell">Employee 6</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form6.value.vacation }}</div>

            </div>

        </form>

        <form [formGroup]="form7">

            <div class="forma">

                <div class="cell">Employee 7</div>

                <input class="cell" type="text"  formControlName="secondName">

                <input class="cell" type="date" formControlName="date">

                <div class="cell" formControlName="vacation">{{ form7.value.vacation }}</div>

            </div>

        </form>

        <form>

            <div class="forma1">

                <div class="cell">Month number</div>

                <div class="cell">Quarter</div>

            </div>

        </form>

        <form [formGroup]="form8">

            <div class="forma">

                <input class="cell" type="text" formControlName="monthNumber">

                <input class="cell" type="text" formControlName="kvartal">

            </div>

        </form>

    </div>

    <form>

    <div id="column" class="animate-charcter">

        <p id="secondName">In selected month</p>

        <p id="kvartal">Ends in this quarter</p>

        <p id="year">In next year</p>

    </div>

    </form>

    </div>

    <button (click)="calculateVacation()">Calculate</button>

    </body>

import { Component } from '@angular/core';

import { FormGroup, FormControl } from '@angular/forms';

@Component({

  selector: 'app-localstorage-ten',

  templateUrl: './localstorage-ten.component.html',

  styleUrls: ['./localstorage-ten.component.css']

})

export class LocalstorageTenComponent {

  form1 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form2 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form3 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form4 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form5 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form6 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form7 = new FormGroup({

    secondName: new FormControl(),

    date: new FormControl(),

    vacation: new FormControl()

  });

  form8 = new FormGroup({

    monthNumber: new FormControl(),

    kvartal: new FormControl(),

  });

  forms:FormGroup[]=[

    this.form1,

    this.form2,

    this.form3,

    this.form4,

    this.form5,

    this.form6,

    this.form7

  ];

  printSecondNameInMonth(){

    document.getElementById("secondName")!.innerHTML = 'In selected month';

    for(let form of this.forms) {

      let date = new Date(form.value.date)

      if(date.getMonth() === (this.form8.value.monthNumber-1)) {

        document.getElementById("secondName")!.innerHTML += ` ${form.value.secondName}`;

      }

   }

};

  getVacationNextYear(vacationStart:Date, secondName:string){

  let date = new Date();

  if(date.getFullYear() < vacationStart.getFullYear()){

    document.getElementById("year")!.innerHTML += ` ${secondName}`;

  }

  }

printSecondNameInQuarter(vacationEnd:Date) {

  for(let form of this.forms) {

    let date = new Date(form.value.vacation)

    console.log(date.getMonth())

    console.log(this.form8.value.monthNumber)

    if(this.form8.value.kvartal == 1){

      if(vacationEnd.getMonth() === 0 || vacationEnd.getMonth()==1 || vacationEnd.getMonth() == 2) {

        document.getElementById("kvartal")!.innerHTML += ` ${form.value.secondName}`;

        continue;

      }

    }

    if(this.form8.value.kvartal == 2){

      if(vacationEnd.getMonth() == 3 || vacationEnd.getMonth()==4 || vacationEnd.getMonth() == 5) {

        document.getElementById("kvartal")!.innerHTML += ` ${form.value.secondName}`;

        continue;

      }

    }

    if(this.form8.value.kvartal == 3){

      if(vacationEnd.getMonth() == 6 || vacationEnd.getMonth()==7 || vacationEnd.getMonth() == 8) {

        document.getElementById("kvartal")!.innerHTML += ` ${form.value.secondName}`;

        continue;

      }

    }

    if(this.form8.value.kvartal == 4){

      if(vacationEnd.getMonth() === 9 || vacationEnd.getMonth()===10 || vacationEnd.getMonth() === 11) {

        document.getElementById("kvartal")!.innerHTML += ` ${form.value.secondName}`;

        continue;

      }

    }

 }

}

calculateVacation(){

  let currentDate:any = new Date();

  document.getElementById("year")!.innerHTML = `In next year`;

  document.getElementById("kvartal")!.innerHTML = "";

  document.getElementById("kvartal")!.innerHTML = `Ends in this quarter`;

  this.printSecondNameInMonth()

  for(let form of this.forms) {

      let startDate:any = new Date(form.value.date)

      let vacationStart = new Date(startDate);

      vacationStart.setFullYear(currentDate.getFullYear())

      vacationStart.setMonth(vacationStart.getMonth()+11);

      let vacationEnd = new Date(vacationStart);

      this.printSecondNameInQuarter(vacationEnd);

      this.getVacationNextYear(vacationStart, form.value.secondName);

     vacationEnd.setMonth(vacationEnd.getMonth()+1)

      form.setValue({

      secondName: form.value.secondName,

      date:form.value.date,

      vacation: vacationStart.toDateString()

     })

    }

    this.SetListOfEmployes();

  };

  SetListOfEmployes(){

    const obj:any = {

      list: []

    }

      for(let form of this.forms) {

        obj.list.push({

          name: form.value.secondName,

          date: form.value.date

      })

      };

      localStorage.setItem("list", JSON.stringify(obj))

   }

}

Css:

body{

    height: 1000px;

}

#container{

    display: flex;

    flex-direction: column;

}

#newcon{

    display: flex;

    flex-direction: row;

}

#column{

    display: flex;

    flex-direction: column;

    margin-top: 100px;

    margin-left: 130px;

    font-size: 40px;

    width: 700px;

}

.checkbox{

    width: 40px;

    height: 40px;

}

.text{

    border-color: #1976d2;

    background-color: #1976d2;

}

input, select, textarea{

    color: white;

}

table{

    border: 5px solid #1976d2;

}

label{

    color: #1976d2;

}

th, td {

    border: 3px solid #1976d2;

    text-align: center;

    color: #1976d2;

 }

.forma{

    display: flex;

}

.forma1{

    display: flex;

    margin-top: 30px;

}

.cell{

    width: 150px;

    border: 2px solid #1976d2;

    text-align: center;

    color: #1976d2;

}

#top{

    margin-top: 20px;

    width: 600px;

}

button{

    margin-top: 30px;

    margin-left: 170px;

    width: 200px;

    height: 40px;

    font-size: 20px;

    display: flex;

    flex-direction: column;

    align-items: center;

    padding: 6px 20px;

    font-family: -apple-system, BlinkMacSystemFont, 'Roboto', sans-serif;

    border-radius: 30px;

    border: none;

    color: #fff;

    background: linear-gradient(180deg, #4B91F7 0%, #367AF6 100%);

}

.position{

    flex-direction: column;

}

.animate-charcter

{

  background-image: linear-gradient(

    -225deg,

    #367AF6 0%,

    #364cf5 29%,

    #9f99f5 67%,

    #090274 100%

  );

  background-size: auto auto;

  background-clip: border-box;

  background-size: 200% auto;

  color: #fff;

  background-clip: text;

  -webkit-background-clip: text;

  -webkit-text-fill-color: transparent;

  animation: textclip 2s linear infinite;

  display: inline-block;

  font-size: 20px;

}

.container{

margin-top: 30px;

}

@keyframes textclip {

  to {

    background-position: 200% center;

  }

}

Вторая страница:

<button value="Show list" (click)="showList()">Show employes</button>

<div class="wrapper">

    <p class="animate-charcter" id="elem1"></p>

    <p class="animate-charcter" id="elem2"></p>

    <p class="animate-charcter" id="elem3"></p>

    <p class="animate-charcter" id="elem4"></p>

    <p class="animate-charcter" id="elem5"></p>

    <p class="animate-charcter" id="elem6"></p>

    <p class="animate-charcter" id="elem7"></p>

</div>

import { Component } from '@angular/core';

@Component({

  selector: 'app-additional-ten',

  templateUrl: './additional-ten.component.html',

  styleUrls: ['./additional-ten.component.css']

})

export class AdditionalTenComponent {

  showList(): void {

    this.createList(this.getListFromLS())

    }

 createList(content:any) {

  let i:number = 1;

  console.log(content)

    for(let elem of content.list){

      if(i==8)

      {

        return;

      }

      let stroka: string = "";

      stroka+=`Second name: ${elem.name};  Hire date: ${elem.date}`

      stroka+='\n';

      document.getElementById(`elem${i}`)!.innerHTML= stroka;

      i++

    }

}

 getListFromLS() {

    return JSON.parse(localStorage.getItem("list")!);

}

}

Результат работы программы:

