**Лабораторна робота № 4**

**Тема:** Введення в JavaScript розробку. TypeScript ООП

**Мета:** Навчитись працювати з TypeScript ООП

**Завдання**

№7

Облік успішності (студент, оцінка, дисципліна, вид оцінки)

**Хід роботи**

Index.html

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>ЛМВ Лр4</title>  
</head>  
<body>  
<script src="script.js"></script>  
</body>  
</html>

Script.ts

class Student {  
 private static *newId*: number = 1;  
 public id: number;  
 public surname: number;  
 public name: number;  
 public middle\_name: number;  
 public faculty: number;  
 public group: number;  
  
 public constructor(surname, name, middle\_name, faculty, group) {  
 this.id = Student.newId++;  
 this.surname = surname;  
 this.name = name;  
 this.middle\_name = middle\_name;  
 this.faculty = faculty;  
 this.group = group;  
 }  
}  
  
class Discipline {  
 public id: number;  
 public name: number;  
  
 public constructor(id, name) {  
 this.id = id;  
 this.name = name;  
 }  
  
 static Math: Discipline = new Discipline(1, 'Math');  
 static Ukrainian: Discipline = new Discipline(2, 'Ukrainian');  
 static English: Discipline = new Discipline(3, 'English');  
}  
  
class MarkType {  
 public minScore: number;  
 public maxScore: number;  
 public increase: number;  
  
 constructor(minScore, maxScore, increase) {  
 this.minScore = minScore;  
 this.maxScore = maxScore;  
 this.increase = increase;  
 }  
  
 static MT1\_5 = new MarkType(1, 5, 1);  
 static MT1\_12 = new MarkType(1, 12, 1);  
}  
  
class Mark {  
 public mark: number;  
 public markType: MarkType;  
 public discipline: Discipline;  
 public student: Student;  
  
 constructor(mark, markType, discipline, student) {  
 this.mark = mark;  
 this.markType = markType;  
 this.discipline = discipline;  
 this.student = student;  
 }  
  
 static marks: Mark[] = [];  
  
 static addMark(mark: Mark): void {  
 this.marks.push(mark);  
 }  
  
 static marksBy(by: string) {  
 const lists = {};  
 this.marks.forEach((item: Mark) => {  
 if (lists[item[by].id]) {  
 lists[item[by].id].marks.push(item);  
 } else {  
 const o = {marks: [item]};  
 o[by] = item[by];  
 lists[item[by].id] = o;  
 }  
 });  
 return lists;  
 }  
}  
  
const Liza: Student = new Student('Цапик', 'Єлизавета', 'Романівна', 'ПІ', 181);  
const Ania: Student = new Student('Хуршудян', 'Аня', 'Юріївна', 'ПМ', 192);  
const Roman: Student = new Student('Зейлик', 'Роман', 'Вікторович', 'ПІ', 182);  
const Sasha: Student = new Student('Муштин', 'Олександр', 'Романович', 'ПІ', 211);  
const Olga: Student = new Student('Кулачук', 'Ольга', 'Романівна', 'ПІ', 181);  
const Masha: Student = new Student('Кулачук', 'Марія', 'Романівна', 'КІ', 202);  
const Vania: Student = new Student('Городовий', 'Іван', 'Петрович', 'ПМ', 201);  
  
const students: Student[] = [Liza, Ania, Roman, Sasha, Olga, Masha, Vania];  
Mark.*addMark*(new Mark(5, MarkType.*MT1\_5*, Discipline.*Ukrainian*, students[0]));  
Mark.*addMark*(new Mark(5, MarkType.*MT1\_5*, Discipline.*Math*, students[0]));  
Mark.*addMark*(new Mark(4, MarkType.*MT1\_5*, Discipline.*Math*, students[1]));  
Mark.*addMark*(new Mark(4, MarkType.*MT1\_12*, Discipline.*English*, students[6]));  
Mark.*addMark*(new Mark(7, MarkType.*MT1\_12*, Discipline.*Ukrainian*, students[6]));  
***console***.log('Marks by students', Mark.*marksBy*('student'));  
***console***.log('Marks by discipline', Mark.*marksBy*('discipline'));

**Висновок:** Навчилась працювати з класами в TypeScript