Одеський коледж комп’ютерних технології «Сервер»

Звіт з лабораторної роботи №3

«Об’єкти та ООП у JavaScript»

Виконував :

Студент групи К21.1

Машков Володимир

Перевірив:  
Рудниченко Н.Д.

Одеса 2024

# Зміст

[Зміст 2](#_Toc160112606)

[Введення 3](#_Toc160112607)

[Теоретична частина 4](#_Toc160112608)

[Практична частина 7](#_Toc160112609)

[Завдання 1 7](#_Toc160112610)

[Завдання 2 8](#_Toc160112611)

[Завдання 3 9](#_Toc160112612)

[Завдання 4 10](#_Toc160112613)

[Завдання 5 12](#_Toc160112614)

[Завдання 6 13](#_Toc160112615)

[Підсумок 17](#_Toc160112616)

[Додаткові матеріали 18](#_Toc160112617)

# Введення

Вивчити та застосувати на практиці знання та навички про роботу із об’єктами ООП в JS. Вивчити та застосувати базових функції для роботи із об’єктами. Вивчити типи та різновиди видів побудування об’єктів.

# Теоретична частина

У javaScript існує кілька видів об'єктів:

вбудовані об'єкти;

об'єкти браузера;

об'єкти, які програміст створює самостійно (користувацькі).

Вбудовані об'єкти - це зумовлені об'єкти number , string , array ...

Познайомимося з користувацькими об'єктами в javaScript.

1. Object(об'єкти);
2. Number (обробка чисел);
3. String (обробка рядків);
4. Array (масиви);
5. Math (математичні формули, функції та константи);
6. Date (робота з датами та часом);
7. RegExp;
8. Global (його властивості Infinity, NaN, undefined);Function.

Існує 2 способи створення об'єктів:

1. використання ініціалізатора об'єкта (або створення об'єктів-колекцій)
2. використання конструктора об'єктів (створення класів-конструкторів)

Функція-конструктор дозволяє наділити об'єкт властивостями за замовчуванням. Ці властивості будуть у кожного створюваного екземпляра об'єкта

Додавання властивостей до класу об'єктів. Значення властивості можна додати для:

1. конкретного екземпляра об'єкта;
2. цілого класу об'єктів.
3. Додавання властивостей до конкретного (екземпляра) об'єкта: ім'я\_об'єкта.ім'я\_властивості = значення

Прототип - об'єкт, що визначає структуру

Git — це система контролю версій, яка використовується для відстеження змін у коді програми. З його допомогою розробники можуть створювати коміти (записи про зміни), створювати гілки для розробки нових функцій, об'єднувати зміни з різних гілок та зберігати історію змін.

Наприклад, git commit -m "Describe changes" зберігає зміни до репозиторію, а git branch feature-branch створює нову гілку з назвою "feature-branch".

Приклад команд Git:

# Ініціалізація нового репозиторію

git init

# Додавання всіх файлів для відстеження змін

git add .

# Створення коміту, що описує зміни

git commit -m "Изменения в таблице HTML"

# Створення нової гілки

git branch new-feature

# Перехід до створеної гілки

git checkout new-feature

# Об'єднання гілок (після внесення змін)

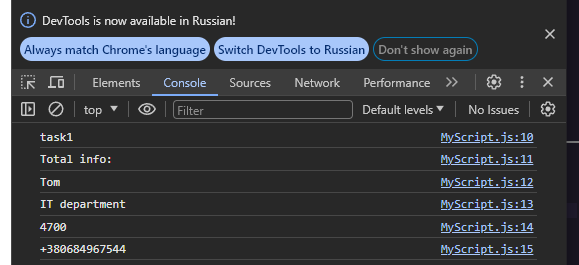
git merge new-feature

# Практична частина

# Завдання 1

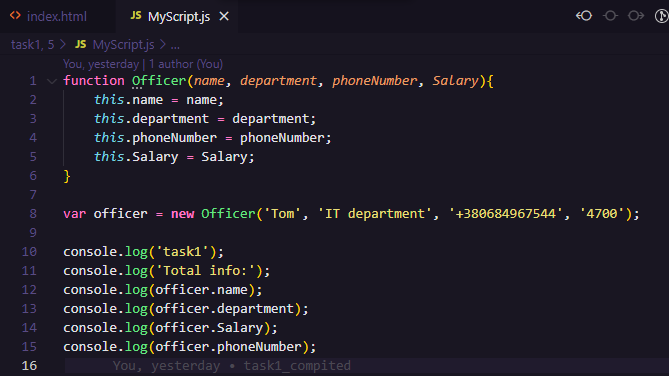
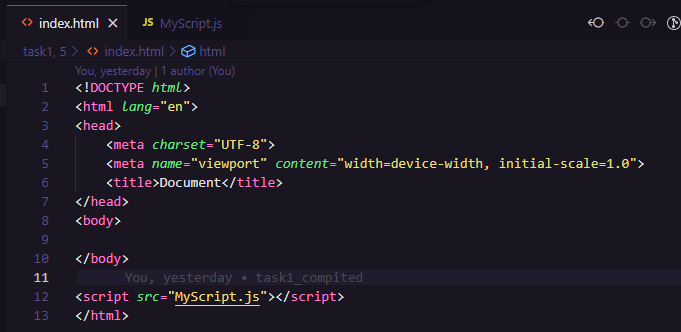
Умова : Побудуйте об’єкт Співробитник, який має наступні поля : ім’я, відділ, телефон та зарплатня. Для ініціалізації класу використовувати функцію-конструктор та ключове слова this. Побудуйте екземпляр класу.

Результат :



Малюнок 1 : Результат роботи з першого завдання

Код програми :

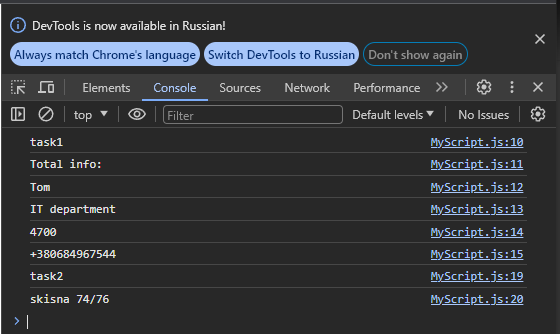


Малюнок 2-3 : код програми із першого завдання

# Завдання 2

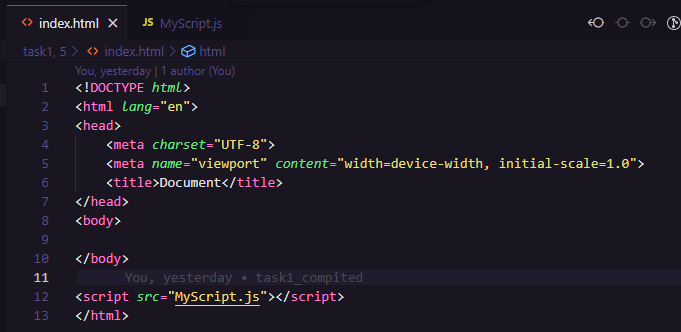
Умова : Для завдання 1, через вже існуючий єкзепляр об’єкта Співробитник додати свойство «адреса» до класу об’єктів

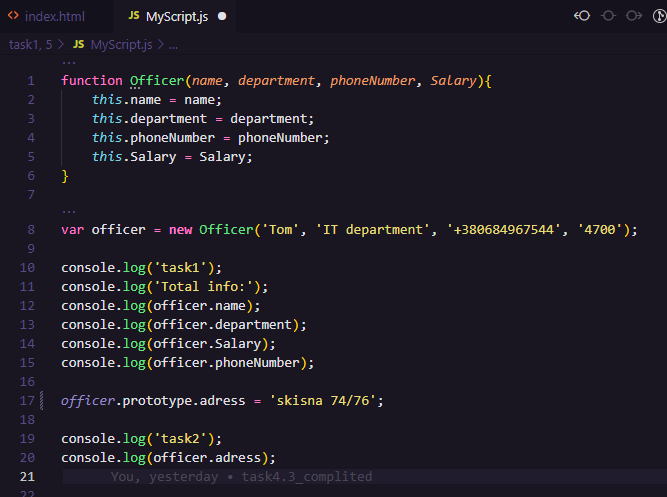
Результат :



Малюнок 4 : результат виконання другого завдання

Код завдання :





Малюнок 5-6 : код програми другого завдання

# Завдання 3

Умова : Побудувати клас об’єктів (Tour) для роботи туристичного агентства із методом підрахунку вартості поїздки із урахуванням : кількість людей \* кількість дній \* тариф держави. Побудувати екземляр об’єкта із значенням свойств. Вивести усі свойства об’єкта tyrkeyTour на екран. Метод об’єкта розрахування побудувати на основі функції.

Результат :

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Малюнок 7 : результат виконання третього завдання

Код програми :

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Малюнок 8 : код програми третього завдання

# Завдання 4

Умова : додати у код програми для виконання завдання : До вбудованного класу string додати метод PrintMe, який виводить слово «Ура!», як заголовок, який обирає користувач у випадаючему меню.

Результат :

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Изображение выглядит как снимок экрана, текст, электроника, схема

Автоматически созданное описание

Малюнок 9-11 : результат виконання четвертого завдання

Код програми :

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Малюнок 12 : код програми четвертого завдання

# Завдання 5

Умова : 5. Створити об'єкт-колекцію Співробітник, який містить відомості проспівробітника деякої фірми, такі як Ім'я, Відділ, Телефон, Зарплата і відображаєдані про цього співробітника (створити метод об'єкта для відображення даних)

Результат :

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Малюнок 13 : результат роботи програми п’ятого завдання

Код програми :

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Малюнок 14 : код програми п’ятого завдання

# Завдання 6

Умова : Створити меню з пунктів-гіперпосилань:

Визначити клас MenuItem.

Визначити властивості: title (назва пункту меню або текст гіперпосилання), url (значення атрибута href для гіперпосилання), метод show (виведення на екран створеного меню).

Додати 4 пункти меню у вигляді списку (тег li)

Результат :

Изображение выглядит как текст, снимок экрана, программное обеспечение, мультимедиа

Автоматически созданное описание Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, веб-страница

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, Значок на компьютере

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, веб-страница

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, Значок на компьютере

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, карта

Автоматически созданное описание

Малюнок 15-23 : результат роботи шостого завдання

Код програми :

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Малюнок 24 : код програми шостого завдання

# Підсумок

У результат роботи та виконання цієї роботи, ми навчилися працювати із класами, їх екземплярами, додавати додаткові поля для конкретного екземпляру класу, або для всіх екземплярів класу. Додавати до сторінки нові елементи, на основі тих даних, які ми отримуємо від користувача.

# Додаткові матеріали

Мій GitHub :

<https://github.com/ActaniumTime/web_pr4>