Лабораторна робота №1  
JavaScript. Конструктори та прототипи

*Мета роботи:* вивчити техніки визначення псевдокласів в JavaScript.

*Знати:* особливості використання техніки визначення псевдокласів в JavaScript.

*Уміти:* cстворювати свої типи даних в JavaScript; визначати функції-конструктори; працювати з прототипами; використовувати загальні методи класу Object.

*Обладнання:* технічне забезпечення: персональний комп’ютер, програмне забезпечення: текстовий редактор Notepad++, Visual Studio Code, браузер.

***Самостійна підготовка до виконання лабораторної роботи***

1. Записати в зошит тему, мету, обладнання, практичні завдання свого варіанту(при наявності).
2. За літературою до лабораторної роботи вивчити необхідний теоретичний матеріал та письмово відповісти на запитання для самостійної підготовки.
3. За інструкцією до лабораторної роботи ознайомитись з порядком виконання роботи.

***Короткі теоретичні відомості***

У мові JavaScript не існує класів, але за допомогою функцій-конструкторів і прототипів можна реалізувати подібну поведінку з використанням класів в інших ООП мовах програмування. Якщо в контексті JavaScript використовується термін «клас», під ним мається на увазі використання прототипу або функції-конструктора.

Функції, які застосовуються спільно з оператором new називаються функціями-конструкторами. Завдання оператора **new** – створити порожній об'єкт і передати його функції конструктору як значення ключового слова this. Завдання **функції-конструктора** – заповнити об'єкт властивостями і методами.

Приклад створення об'єкта – new Date().

Об'єкт – структура даних, яка містить іменовані фрагменти даних, а також може містити методи для роботи з фрагментами даних. Об'єкт групує пов'язані значення і дії в єдиний набір, який, як правило, полегшує процес програмування. У таких мовах як C# або C++ клас визначає структуру об'єкта. В JavaScript замість класу будуть використовуватися конструктори, які налаштовують по одному і тому ж принципу порожні об'єкти.

Кожен об'єкт має свій набір **властивостей екземпляра**. Якщо є 10 об'єктів створених функцією-конструктором, то у кожного об'єкта буде своя властивість (всього буде 10 копій властивостей). Властивість екземпляра – властивість, яку було задано через функцію конструктор.

**Метод екземпляра** – те ж саме що і властивості екземпляра, але замість значення в ньому зберігатися функція.

Якщо в методі екземплярапотрібно звернутися до властивостей екземпляра обов'язково потрібно використовувати ключове слово **this**.

У мові програмування C# є поняття статичний метод та статична властивість, тобто значення або дія, яке є загальним для всіх, а не належить, конкретному екземпляру. Для того, що б імітувати статичні методи в JavaScript, метод або властивість потрібно визначити на функції-конструкторі, а не на об'єкті.

Методи функції-конструктора не можуть використовувати ключове слово this, так як вони не пов'язані з конкретним об'єктом.

**Прототип** – це об'єкт, з якого створюється новий об'єкт, успадковується. Будь-який об'єкт в JavaScript успадковує властивості і методи його прототипу. Якщо додати значення до прототипу, він буде доступний для всіх об'єктів, створених за допомогою функції конструктора. Щоб отримати доступ до прототипу, використовується наступний синтаксис Конструктор.prototype.

Коли в JavaScript створюється нова функція-конструктор, вона додає в об’єкт наперед визначений набір властивостей.

Загальні методи прототипу Object:

**toString()** – метод для перетворення об'єкта в рядкове представлення.

**valueOf()** – багато в чому схожий на toString(), але викликається коли в JavaScript Ви бажаєте перевести об'єкт до елементарного типу.

**equals()** – метод (не є методом Object) використовується для порівняння двох об'єктів за значенням.

**compareTo(obj)** – метод (не є методом Object) використовується для визначення відносин між об'єктами. Якщо метод повертає значення менше нуля об'єкт на якому був проведений виклик менше ніж об'єкт, який був переданий в якості параметра. Якщо метод поверне значення більше нуля – об'єкт, на якому зроблений виклик, більше ніж той, який переданий в параметрах. Результат методу дорівнює нулю якщо об'єкти рівні.

***Порядок виконання роботи***

1. Вивчіть основні конструкції та поняття, які розглянуті на лекції.
2. Створіть функцію-конструктор Box (коробка) для створення об'єкта з властивостями Height (висота), Width (ширина), Depth (глибина) і Material (матеріал). Також в об'єкті повинні бути методи Volume – для отримання об’єму та Equals () – для порівняння об'єктів, об'єкти повинні вважатися рівними якщо містять однакові значення в властивості Material.
3. Створіть функцію конструктор, яка створює об'єкт з властивістю екземпляру, властивістю функції-конструктора, з методом екземпляру і методом функції-конструктора, імена виберіть довільно.
4. Розробіть функцію-конструктор, яка буде створювати об'єкт Human (людина) створіть масив об'єктів і реалізуйте функцію, яка буде сортувати елементи масиву за значенням властивості Age по зростанням або за спаданням.
5. Розробіть функцію-конструктор, яка буде створювати об'єкт Human (людина) додайте на свій розсуд властивості і методи в цей об'єкт. Подумайте які методи і властивості слід зробити рівня екземпляра, а які рівня функції-конструктора.
6. Створіть приклад, який демонструє призначення методу toString ().

***Питання для самостійної роботи***

1. Що таке функція-конструктор?

2. Чи правильно стверджувати що в JavaScript є класи?

3. Яке призначення оператора new при використанні спільно з функцією конструктором?

4. В чому різниця між властивістю та методом?

5. В чому різниця між властивістю екземпляра та властивістю функції-конструктора?

6. У чому полягає різниця між toString та valueOf?

7. Опишіть дії, які потрібно зробити, якщо потрібно створити об'єкти, які підтримують порівняння за значенням.

***Література:***

1. Об’єкти:

<http://learn.javascript.ru/object>

1. Об’єкти: <https://developer.mozilla.org/ru/docs/Web/JavaScript/Guide/Working_with_Objects>

Звіт до роботи

1. Оформити лабораторну роботу в зошиті. Записати тему, мету, знати, уміти, обладнання.
2. Письмово відповісти на запитання для самостійної підготовки.
3. Виконати завдання згідно свого варіанту.
4. Зробити висновок.