МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

**Лабораторна робота №1**

з дисципліни «**Технології програмування**»

на тему: «JavaScript.Basics»

Варіант №9

**Виконала:**

студентка гр. ІР-21

Аріна ЛАЗАРЧУК

**Перевірив:**

Ольга БОНДАРЕНКО

Зараховано від

\_\_\_.\_\_\_.\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис викладача)

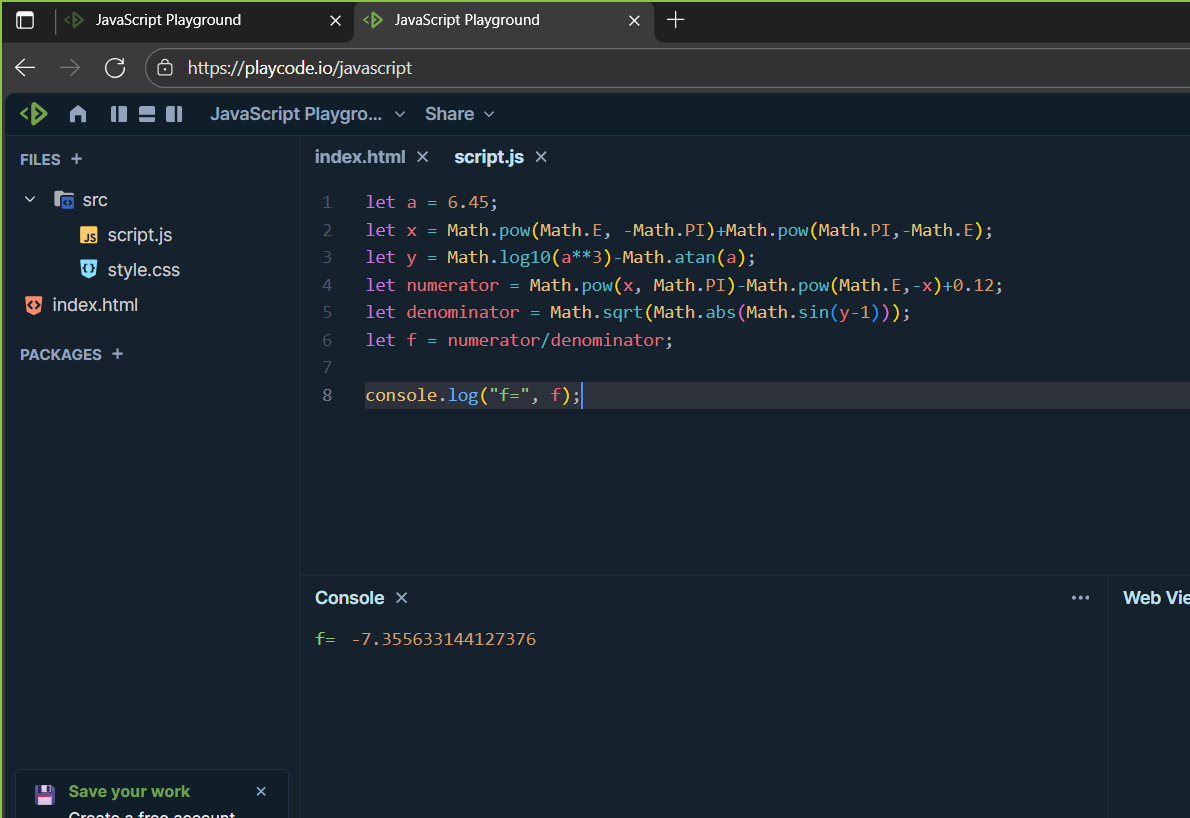
Київ-2025

**Мета:** Ознайомлення з базовими конструкціями мови JavaScript для обробки числових даних.

**Завдання**

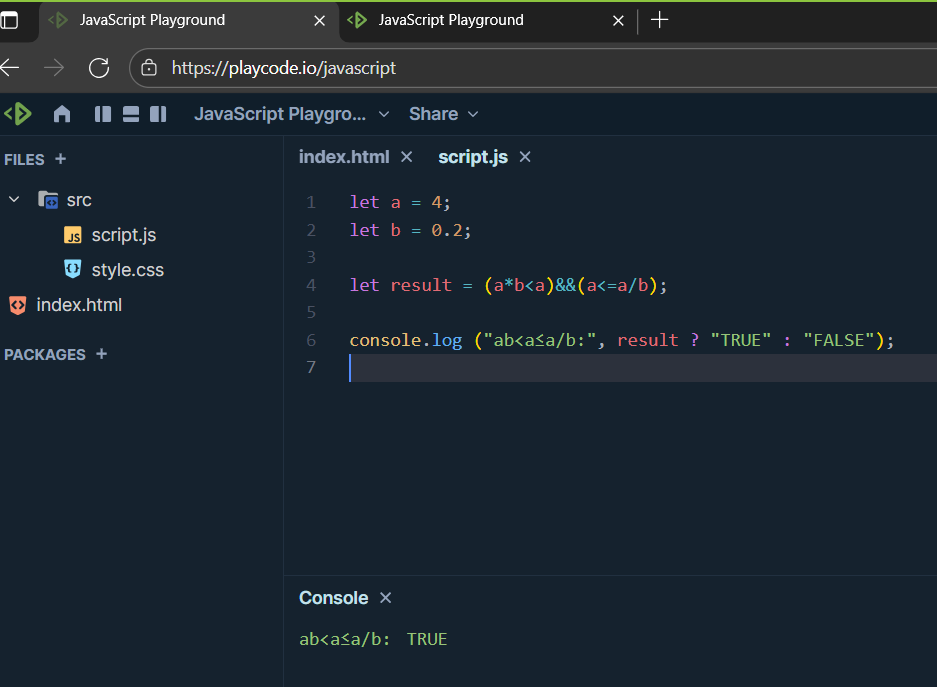
***Завдання 1:*** Обчислити значення функції та вивести результат





***Завдання 2:*** Записати мовою програмування даний логічний вираз і визначити значення результату логічних операцій TRUE або FALSE при вказаних значеннях змінних





**Контрольні запитання**

**1. Що таке змінна в JavaScript і як її оголосити?**

Змінна в JavaScript — це іменована комірка пам'яті, що використовується для зберігання даних. Її можна оголосити за допомогою трьох ключових слів:

* **var** (старий стандарт, має глобальну або функціональну область видимості).
* **let** (сучасний стандарт ES6+, має блокову область видимості).
* **const** (сучасний стандарт ES6+, має блокову область видимості, але її значення не можна змінити після ініціалізації).

**2. Які основні етапи створення змінної?**

Процес створення змінної включає два основні етапи:

1. **Оголошення** (Declaration): виділення місця в пам'яті та присвоєння імені змінній.
2. **Ініціалізація** (Initialization): присвоєння початкового значення змінній.

**3. Які правила іменування змінних у JavaScript?**

Правила іменування змінних:

* Назва повинна починатися з літери, знака підкреслення \_ або знака долара $.
* Назва може містити літери, цифри, \_ та $.
* Назви є чутливими до регістру (myVar і myvar — різні змінні).
* Не можна використовувати зарезервовані слова JavaScript (наприклад, if, for, let).
* Загальноприйнятою є нотація **camelCase** (наприклад, myVariableName).

**4. Чим відрізняється оператор == від ===?**

* Оператор **==** (оператор нестрогої рівності) порівнює значення, але перед цим намагається привести їх до одного типу. Наприклад, 10 == '10' поверне true.
* Оператор **===** (оператор строгої рівності) порівнює і значення, і тип даних. Наприклад, 10 === '10' поверне false, оскільки типи різні (число та рядок). Рекомендується використовувати ===, щоб уникнути несподіваної поведінки.

**5. Для чого використовується унарний та бінарний оператор +?**

* **Унарний оператор +**: ставиться перед операндом і намагається перетворити його на число. Наприклад, +'10' перетворить рядок '10' на число 10.
* **Бінарний оператор +**: використовується для додавання чисел або конкатенації (об'єднання) рядків. Наприклад, 5 + 3 поверне 8, а 'Hello' + ' World' поверне 'Hello World'.

**6. Яка різниця між префіксною та постфіксною формами інкременту (++i і i++)?**

* **Префіксна форма ++i**: спочатку **збільшує** значення змінної на 1, а потім повертає нове, змінене значення.
* **Постфіксна форма i++**: спочатку **повертає** поточне значення змінної, а вже потім збільшує його на 1.

**7. Які існують методи округлення чисел у JavaScript (назвати не менше двох)?**

Найпоширеніші методи округлення:

* **Math.round()**: округлює число до найближчого цілого. Наприклад, Math.round(4.7) поверне 5, а Math.round(4.3) поверне 4.
* **Math.floor()**: округлює число до найближчого меншого цілого. Наприклад, Math.floor(4.7) поверне 4.
* **Math.ceil()**: округлює число до найближчого більшого цілого. Наприклад, Math.ceil(4.3) поверне 5.

**8. Що повертає оператор % і наведіть приклад його використання.**

Оператор **%** (оператор залишку від ділення) повертає залишок від ділення першого операнда на другий.

**9. Як працює умовна конструкція if і які значення вважаються «хибними» (false)?**

Умовна конструкція **if** виконує блок коду, якщо умова в дужках є істинною (true).

**10. Що таке тернарний оператор ? і наведіть приклад його використання.**

Тернарний оператор ? — це скорочена форма умовної конструкції **if-else**. Він має три операнди і використовується для присвоєння значення змінній залежно від умови. Синтаксис: умова ? вираз\_якщо\_true : вираз\_якщо\_false;