# Лабораторна робота №2

# Основні конструкції мови JavaScript

#### Мета роботи: вивчити основні конструкції мови JavaScript.

## Теоретична частина

## Умовні конструкції

Умовні конструкції дозволяють виконати ті чи інші дії, залежно від певних умов.

### Оператор if

Оператор if належить до найбільш популярних і застосовується наступним чином:

if (умова) {

[оператори]

}

Як умова може вказуватися будь-який логічний вираз. Якщо результат умови дорівнює true, виконуються оператори, продовжується виконання іншого програмного коду. Якщо умова повертає false, оператори ігноруються.

### Конструкція if … else

Іноді однієї конструкції if виявляється недостатньо. Часто потрібно зарезервувати набір операторів, які виконуватимуться у разі, коли умовний вираз повертає false. Це можна зробити, додавши ще один блок безпосередньо після блоку if:

if (умова) {

[оператори]

}

else {

[оператори]

}

### Конструкція switch ... case

Конструкція switch ... case є альтернативою використанню конструкції if ... else і дозволяє обробити відразу кілька умов.

Після ключового слова switch у дужках йде порівнюваний вираз. Значення цього виразу послідовно порівнюється зі значеннями, розміщеними після оператора сase. І якщо збіг буде знайдено, то виконуватиметься певний блок сase.

Наприкінці кожного блоку сase ставиться оператор break, щоб уникнути виконання інших блоків. Якщо ми хочемо також обробити ситуацію, коли збігу не буде знайдено, можна додати блок default:

let income = 300;

switch (income) {

 case 100:

 console.log("Дохід дорівнює 100"); break;

 case 200:

 console.log("Дохід дорівнює 200"); break;

 case 300:

 console.log("Дохід дорівнює 300"); break;

 default:

 console.log("Дохід невідомої величини"); break;

}

### Тернарна (потрійна) операція

Тернарна операція складається з трьох операндів і має такий вираз:

[перший операнд - умова]? [Другий операнд]: [третій операнд]

Залежно від умови тернарна операція повертає другий чи третій операнд: якщо умова дорівнює true, то повертається другий операнд; якщо умова дорівнює false, то третя.

## Цикли

Цикли дозволяють залежно від певних умов виконувати деяку дію багато разів. JavaScript має такі види циклів:

* for;
* for ... in;
* for ... of;
* while;
* do ... while.

### Цикл for

Цикл for має такий вираз:

for ([ ініціалізація лічильника ]; [ умова ]; [ зміна лічильника ]) {

// дії

}

### Цикл for ... in

Цикл for..in призначений для перебору масивів та об'єктів. Його формальне визначення:

for (індекс in об'єкт) {

// дії

}

### Цикл for...of

Цикл for...of схожий на цикл for...in і призначений для перебору колекцій, наприклад, масивів:

for (індекс of масив) {

// дії

}

### Цикл while

Цикл while виконується до тих пір, поки деяка умова є істинною. Його визначення:

while (умова) {

// дії

}

### Цикл do ... while

У циклі do спочатку виконується код циклу, потім відбувається перевірка умови в інструкції while. І доки ця умова є істинною, цикл повторюється. Його вираз:

do {

// дії

}

while (умова)

### Оператори continue та break

Іноді буває необхідно вийти із циклу до його завершення. І тут можна скористатися оператором break. Якщо потрібно просто пропустити ітерацію, але не виходити з циклу, можна застосовувати оператор continue.

## Практична частина

Для виконання практичної частини уважно ознайомтеся з наведеними прикладами. Як індивідуальне завдання дані приклади мають бути модифіковані і збагачені іншими конструкціями.

Застосувати власні функції. При виведенні інформації передбачити форматування документа, виведення тексту завдання та виведення інформації про розробника скрипта. У всіх скриптах, у заголовку вікна браузера зазначити групу і прізвище студента.

### Оператори умовного переходу

**Завдання 1.** Створіть файл lab2\_1.html, що містить наступний скрипт, який демонструє можливості використання умовного оператора IF:

<!DOCTYPE HTML>

<html>

 <head>

 <meta charset="UTF-8">

 <title>Використання оператора IF</title>

 </head>

 <body>

 <script>

 document.write(

 "<p style='text-align:center'>Використання оператора <strong>IF</strong></p>"

 );

 let age = 24;

 if (age < 18){

 document.write("Ви занадто молоді для перегляду цього сайту"); // нічого не виводиться

 }

 age = 10;

 if (age < 18){

 document.write("Ви занадто молоді для перегляду цього сайту"); // виводиться повідомлення

 }

 document.write("<p style='text-align:center'>Використання оператора <strong>IF</strong> з блоком <strong>ELSE</strong></p>");

 let s = "погано";

 if (s == "добре"){

 document.write("Я в хорошому настрої!");

 }

 else document.write("Мені ", s);

 document.write("<p style='text-align:center'>Використання оператора <strong>IF</strong> з блоками <strong>ELSE IF</strong></p>");

 if (s == "добре") {

 document.write("Я в хорошому настрої!");

 }

 else if (s == "погано") {

 document.write("He впадайте у відчай!");

 }

 else {

 document.write("Незрозуміло, просто", s);

 }

 </script>

 </body>

</html>

Для виконання завдання застосувати інший приклад.

Наприклад, задати змінну lang, яка може набувати значень «ua», «en», «pl» або «de», введених користувачем. Використовуючи оператори if-else-elseif, забезпечити виведення на екран повної назви мови (українська, англійська тощо) залежно від того, що привласнено змінній lang. Обов'язково передбачити випадок невірного завдання значення змінної lang - тоді має виводитися напис «Мову невизначено».

**Завдання 2.** Створіть файл lab2\_2.html, що містить наступний скрипт, що демонструє знаходження більшого числа серед трьох чисел з використанням умовного оператора IF та умовної операції:

<!DOCTYPE HTML>

<html>

 <head>

 <meta charset="UTF-8">

 <title>Використання оператора IF</title>

 </head>

 <body>

 <script>

 "use strict";

 //Приклад знаходження більшого среди трех чисел

 document.write("<p style='text-align:center'>Приклад знаходження більшого числа</p>");

 let a = 1; let b = 22; let c = 3;

 if (a > b && a > c) document.write("Найбільше серед трьох чисел - a=", a);

 else

 if (b > c) document.write("Найбільше серед трьох чисел - b=", b);

 else document.write("Найбільше серед трьох чисел - c=", c);

 //Использование тернарной операции

 let max = a > b ? a : b; max = max > c ? max : c;

 document.write("<p>Значення найбільшого серед трьох чисел -", max);

 </script>

 </body>

</html>

**Завдання 3.** Створіть файл lab2\_3.html, що містить скрипт, що демонструє можливості використання оператора SWITCH:

<!DOCTYPE HTML>

<html>

 <head>

 <meta charset="UTF-8">

 <title>Використання оператора IF</title>

 </head>

 <body>

 <script>

 "use strict";

 let age = parseInt(prompt("Введіть ваш вік"));

 let b = age % 10;

 let s;

 switch (b) {

 case 1:

 s = "рік";

 break;

 case 2:

 case 3:

 case 4:

 s = "роки";

 break;

 default: s = "років";

 }

 if (age > 10 && age < 20)

 s = "років";

 document.write("Вам - ", age, s);

 </script>

 </body>

</html>

### Оператори циклу

**Завдання 4.** Створіть файл lab2\_4.html, що містить скрипт, який демонструє використання операторів циклу for, while, do while:

<!DOCTYPE HTML>

<html>

 <head>

 <meta charset="UTF-8">

 <title>Використання оператора IF</title>

 </head>

 <body>

 <script>

 "use strict";

 document.write("<p>Використання оператора циклу <b>FOR</b></p>");

 // Зведення числа x у ступінь

 let x = parseInt(prompt("Введіть число x")); // задане число

 let y = parseInt(prompt("Введіть ступінь y")); // ступінь, до якого треба звести число

 let z = x; // результат

 for (let i = 2; i <= Math.abs(y); i++) z = z \* x; z = y > 0? z: 1/z;

 document.write("Число ", x, " в ступені ", y, " дорівнює ", z);

 document.write("<p>Використання оператора циклу <b>WHILE</b></p>");

 let z1 = x; let i = 2;

 while (i <= Math.abs(y)) {

 z1 = z1 \* x;

 i++;

 }

 z1 = y > 0? z1: 1/z1;

 document.write("Число ", x, " в ступені ", y, " дорівнює ", z1);

 document.write("<p>Використання оператора циклу <b>DO WHILE</b></p>");

 let z2 = x; i = 2;

 do {

 z2 = z2 \* x; i++;

 }

 while (i <= Math.abs(y)); z2 = y> 0? z2: 1/z2;

 document.write("Число ", x, " в ступені ", y, " дорівнює ", z2);

 </script>

 </body>

</html>

**Завдання 5.** Створіть файл lab2\_5.html, що містить скрипт, що демонструє використання оператора циклу for in:

<!DOCTYPE HTML>

<html>

 <head>

 <meta charset="UTF-8">

 <title>Використання оператора IF</title>

 </head>

 <body>

 <script>

 "use strict";

 let propertyInfo = "";

 for (let propertyName in document) {

 propertyInfo=propertyName + "-" + document[propertyName];

 document.write(propertyInfo + "<br>");

 }

 </script>

 </body>

</html>

**Завдання 6.** Створіть файл lab2\_6.html, що містить скрипт, який демонструє використання оператора циклу while у грі з користувачем:

<!DOCTYPE HTML>

<html>

 <head>

 <meta charset="UTF-8">

 <title>Використання оператора IF</title>

 </head>

 <body>

 <script> "use strict";

 let guess = Math.random() \* 100; // генерація випадкового числа

 guess = Math.floor(guess); // відкидання дробової частини

 let f = true;

 let number;

 let count = 0;

 let answer = confirm("Зіграємо?");

 if (answer) {

 while (f) {

 number = parseInt(prompt("введіть число"));

 count++;

 if (number == guess) {

 alert("Ви вгадали\n кількість спроб: "+ count);

 f = false;

 }

 else

 if (number > guess) {

 alert("Число менше, спробуйте ще раз");

 }

 else {

 alert("Число більше, спробуйте ще раз");

 }

 }

 }

 else alert("Шкода.\nМоже бути в інший раз?");

 </script>

 </body>

</html>

## Контрольні питання

1. Які оператори керування ви знаєте?
2. Для чого використовується оператор break в операторі switch?
3. Як записати складну умову?
4. У чому різниця між ключовими словами else та else if?
5. Що таке умовна операція?
6. Які оператори циклу ви знаєте?
7. У чому відмінність операторів циклу while та do while?
8. Що записується в заголовку оператор циклу for?
9. Де використовується цикл for in?
10. Як ініціалізується лічильник циклу?
11. Як задається крок зміни лічильника циклу?
12. Як примусово вийти із циклу?