# Лабораторна робота 2

# Тема: Особливості виконання операцій та операторів у мові JavaScript. Типи даних. Приведення типів.

# Мета роботи: вивчити відмінності у роботі операцій та операторів у мові JavaScript від інших мов; ознайомитись з базовими типами даних; вивчити особливості приведення типів.

**Хід роботи**

**1.** Визначте, які значення будуть виведені на екран:

**var *a*** = 2, ***b*** = 3, ***c***, ***d***;  
 ***c*** = ++***a***; *alert*(***c***);*//3* ***d*** = ***b***++; *alert*(***d***);*//3* ***c*** = (2+ ++***a***); *alert*(***c***);*//6* ***d*** = (2+ ***b***++); *alert*(***d***);*//6  
 alert*(***a***);*//4  
 alert*(***b***);*//5*

**2.** Визначте, яке значення х буде отримано в результаті виконання коду:

**var *a*** = 2;  
**var *x*** = 1 + (***a*** \*= 2);*//x=5*

**3.** Які значення будуть виведені наступним кодом і чому?

*alert*( **null** || 2 || **undefined** );*//2  
alert*( *alert*(1) || 2 || *alert*(3) );*//1 and 2  
alert*( 1 && **null** && 2 );*//null  
alert*( *alert*(1) && *alert*(2) );*//1 and undefined  
alert*( **null** || 2 && 3 || 4 );*//3***if** (-1 || 0) *alert*( **'первое'** );*//первое***if** (-1 && 0) *alert*( **'второе'** );*//nothing***if** (**null** || -1 && 1) *alert*( **'третье'** );*//третье*

**4.** Пояснити які значення і яких типів будуть давати наступні вирази:

**""** + 1 + 0*//10***""** - 1 + 0*//-1***true** + **false***//1*6 / **"3"***//2***"2"** \* **"3"***//6*4 + 5 + **"px"***//9px***"$"** + 4 + 5*//$45***"4"** - 2*//2***"4px"** - 2*//NaN*7 / 0*//inf***" -9\n"** + 5*//-9 next line 5***" -9\n"** - 5*//-14*5 && 2*//2*2 && 5*//5*5 || 0*//5*0 || 5*//5***null** + 1*//1***undefined** + 1*//NaN***null** == **"\n0\n"***//false*+**null** == +**"\n0\n"***//true*

5. Створіть скрипт lab2-1.js, у якому:

* реалізуйте функцію isInteger(num), яка повертає true, якщо num – ціле число, інакше – false.
* реалізуйте функцію findPrimes(a, b), яка приймає числа a і b, і знаходить усі прості числа, які знаходяться на інтервалі [a, b]. Функція повинна результат виводити у консоль.

Лістинг lab2-1.js

*isInteger*(+*prompt*(**"Input begin"**,**""**))?*alert*(**"false"**):*alert*(**"true"**);  
*findPrimes*(+*prompt*(**"input start namber for search"**,**""**),+*prompt*(**"input end namber for search"**,**""**));  
**function** *isInteger*(inputNumber) {  
 **if**(inputNumber%1){  
 **return true**;  
 }**else**{  
 **return false**;  
 }  
}  
**function** *findPrimes*(beginSearch, endSearch) {  
 **for**(**var** i=beginSearch;i<=endSearch;i++){  
 **for**(**var** divider=0,counter=0;counter<=2&&divider<=i;divider++){  
 **if**(i%divider==0) {  
 counter++;  
 }  
 }  
 **if**(counter!=3)***console***.log(i);  
 }  
}

Створіть скрипт lab2-2.js, у якому реалізуйте такі функції:

* isIPAddress(ip) Перевіряє, чи є вказаний рядок ІР-адресою (повертає true або false)
* findRGBA(text) Знаходить у заданому тексті рядок формату rgba(r, g, b, a)що відповідає кольору у форматі RGBA і повертає його. Якщо такого рядка не знайдено, то повернути null.
* findHexColor(text) Знаходить у заданому тексті колір у форматі #ABC або #ABCDEF. Якщо не знайдено, то повернути null.
* findTags(text, tag) Знаходить у заданому тексті усі теги із заданим ім’ям. Приклад використання: var tags = findTags(s, "img");
* findPosNum(text) Знаходить у тексті усі додатні числа
* findDates(text) Знаходить у тексті усі дати записані у форматі РРРР-ММ-ДД

Лістинг index.html

<!DOCTYPE **html**>  
<**html lang="en"**>  
<**head**>  
 <**meta charset="UTF-8"**>  
 <**title**>lab2</**title**>  
</**head**>  
<**body**>  
<**script src="lab2-1.js"**></**script**>  
<**script type="lab2-2.js"**></**script**>  
</**body**>  
</**html**>

Лістинг lab2-1.js

*isInteger*(+*prompt*(**"Input begin"**,**""**))?*alert*(**"false"**):*alert*(**"true"**);  
*findPrimes*(+*prompt*(**"input start namber for search"**,**""**),+*prompt*(**"input end namber for search"**,**""**));  
**function** *isInteger*(inputNumber) {  
 **if**(inputNumber%1){  
 **return true**;  
 }**else**{  
 **return false**;  
 }  
}  
**function** *findPrimes*(beginSearch, endSearch) {  
 **for**(**var** i=beginSearch;i<=endSearch;i++){  
 **for**(**var** divider=0,counter=0;counter<=2&&divider<=i;divider++){  
 **if**(i%divider==0) {  
 counter++;  
 }  
 }  
 **if**(counter!=3)***console***.log(i);  
 }  
}

Лістинг lab2-2.js

*isIPAddress*(prompt(**"Input ip adress"**,**"My ip addres 192.168.0.1 "**));  
**function** *isIPAddress*(str) {  
 *alert*( /\b(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\b/.test(str) )  
 **return** str  
}  
*findRGBA*(prompt(**"Input RGBA"**,**"rgba(255, 142, 30, 0.5) color"**));  
**function** *findRGBA*(str) {  
 *alert*( /(rgba\((\d{1,3}%?,\s?){3}(1|0?\.\d+)\)|rgba\(\d{1,3}%?(,\s?\d{1,3}%?){2}\))/.test(str) )  
 **return** text  
}  
*findHexColor*(prompt(**"Input hex color"**,**" Color is #ABCDEF"**));  
**function** *findHexColor*(str) {  
 *alert*( /#([A-Fa-f0-9]{6}|[A-Fa-f0-9]{3})/.test(str) )  
 **return** str  
}  
*findPosNum*(prompt(**"Input pos number"**,**" asdas 1ewqwqwq 2jhjhghg 3 ddfgr5sd 8 dashjkd "**));  
**function** *findPosNum*(str) {  
 *alert*( /(!-)\b\d+\b|(\+\d+)/g.test(str) )  
 **return** str  
}  
*findDates*(prompt(**"Input date"**,**"1939-01-09 it's date 2018-03-19"**));  
**function** *findDates*(str) {  
 *alert*( /([12]\d{3}-(0[1-9]|1[0-2])-(0[1-9]|[12]\d|3[01]))/g.test(str) )  
 **return** str  
}  
*findTags*(prompt(**"Input tag"**,**"Hello my favourite color is red,red"**));  
**function** *findTags*(text, tag){  
 **var** regexp = **new *RegExp***(tag, **'g'**);  
 **var** result = **""**;  
 **var** res = **""**;  
  
 **while** (result = regexp.exec(text)) {  
 res += result[0] + **" "**;

}

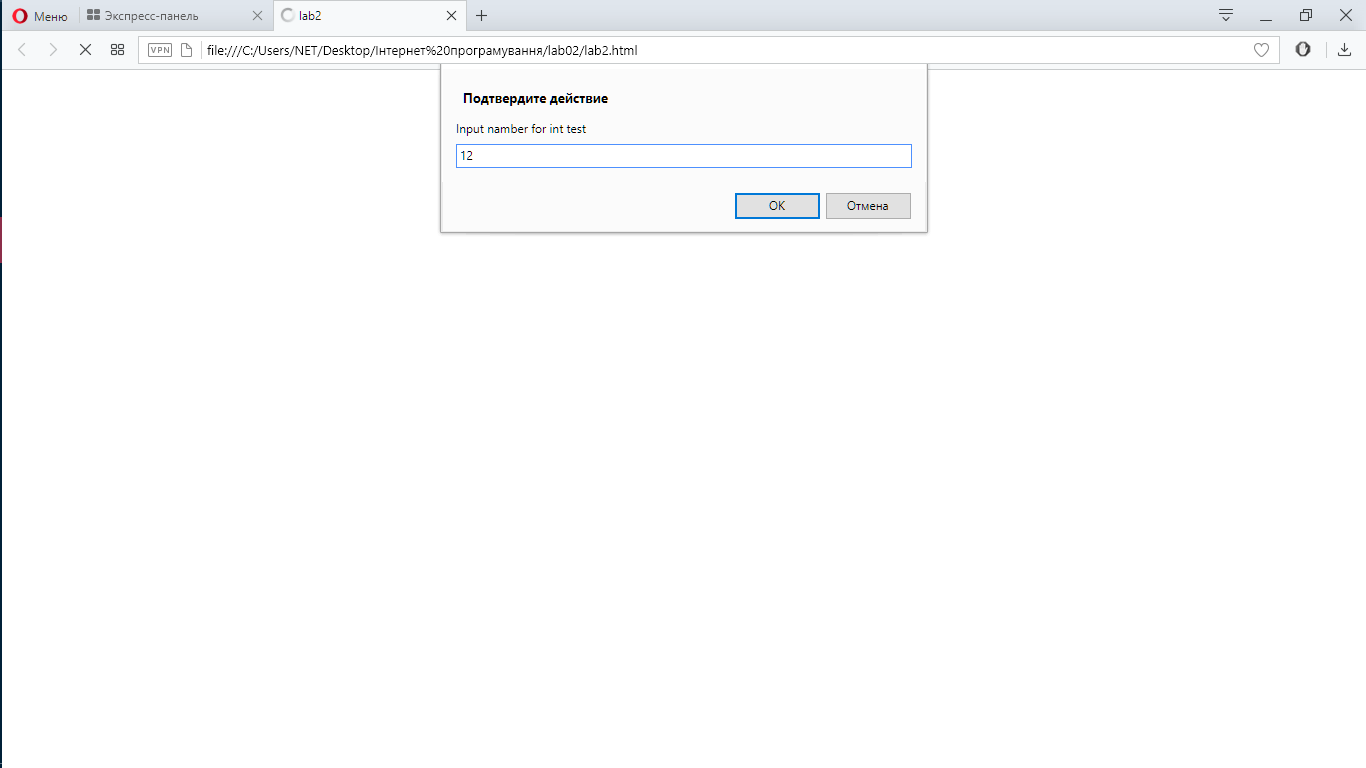


Рис. 1 – Введення числа для перевірки на ціле

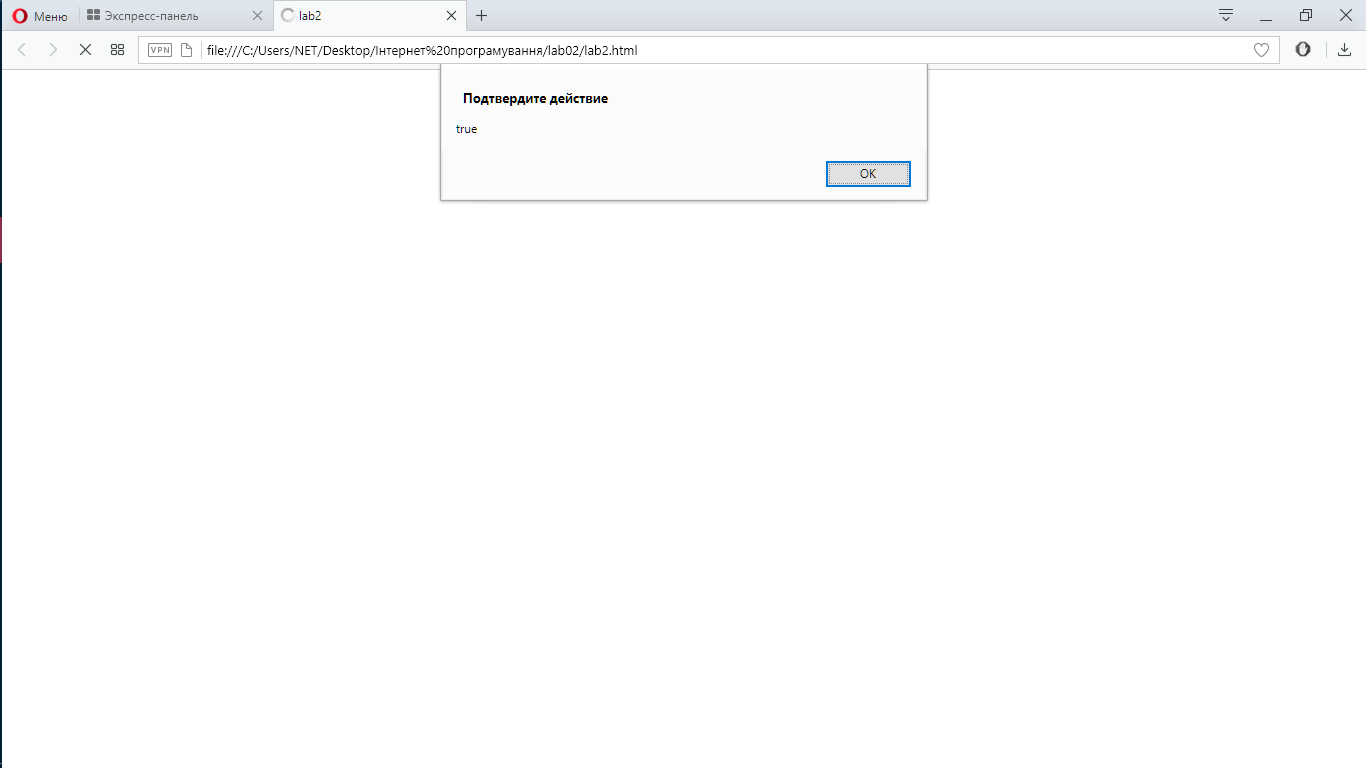


Рис. 2 – Виведення результату виконання програми

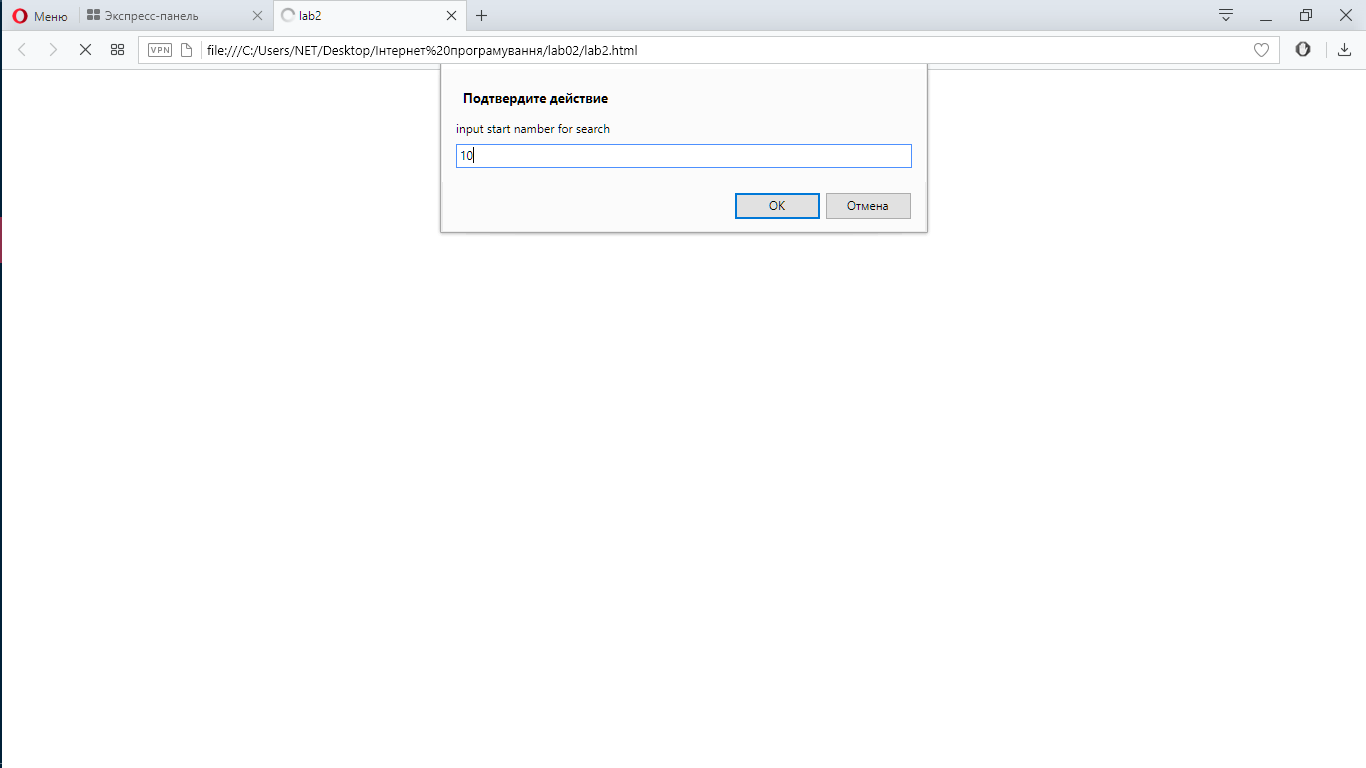


Рис. 3 – Введення початку діапазону

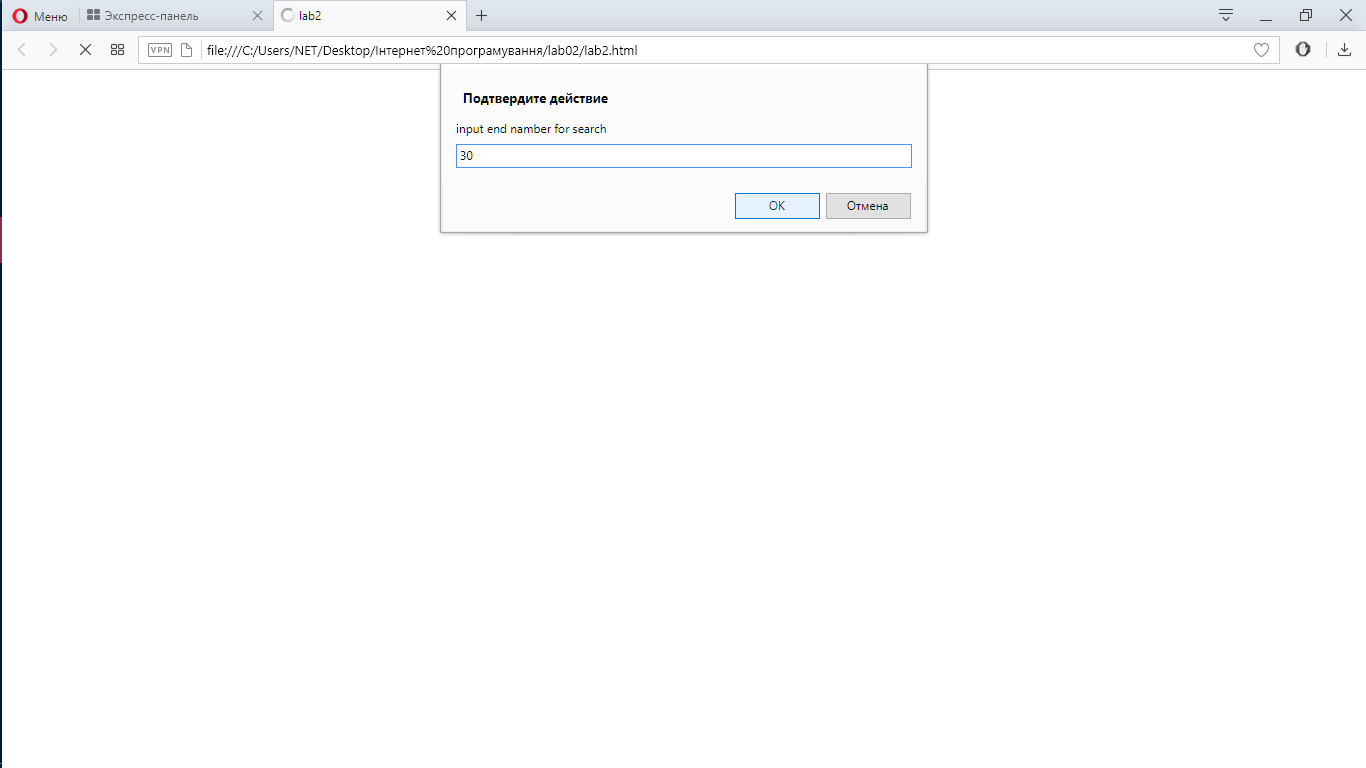


Рис. 4 – Введення кінцевого діапазону

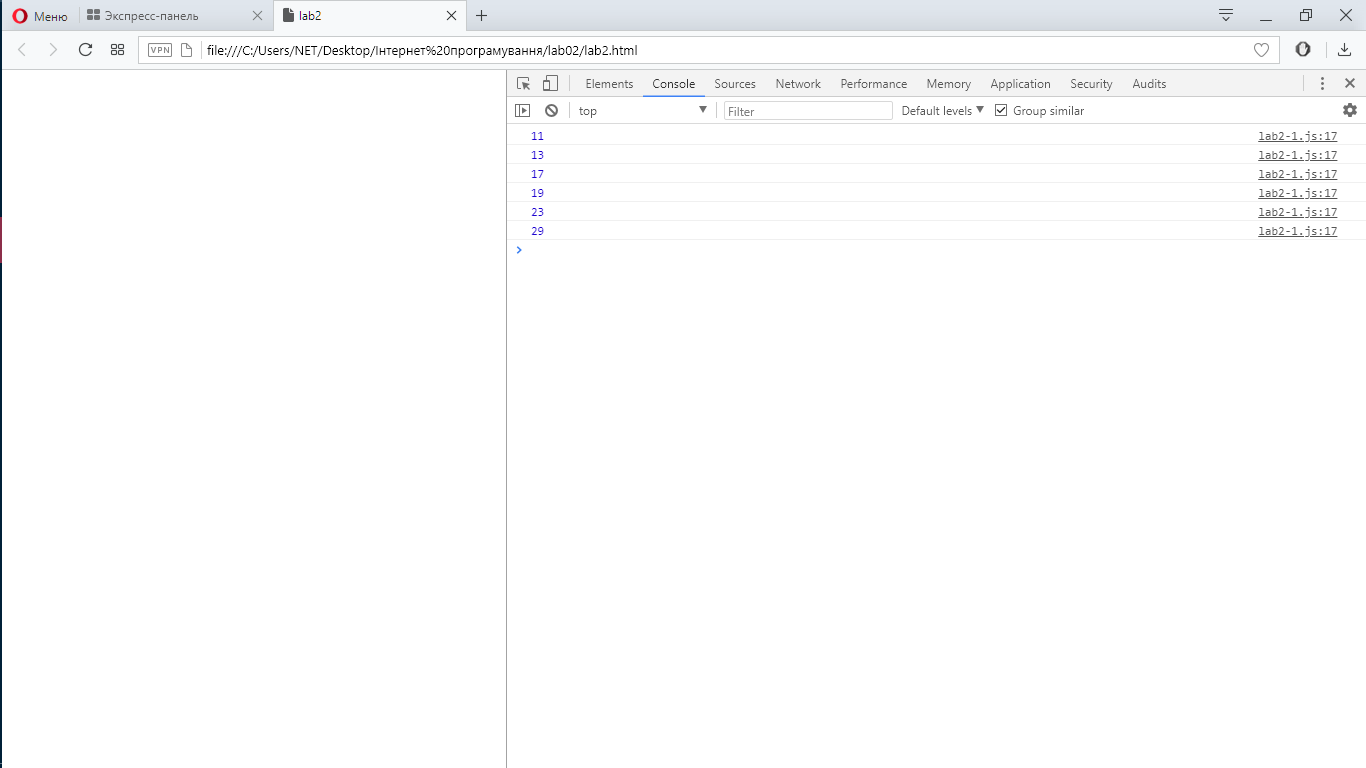


Рис. 5 – Виведення всіх простих чисел в заданому діапазоні

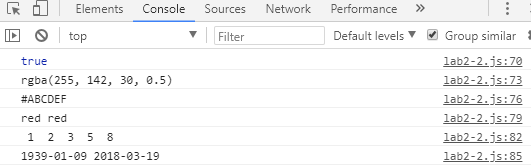


Рис. 6 – Виведення результату виконання регулярних виразів