

# Практичне заняття 2

---

## Тема:

Створення та виконання скриптів у Windows (BAT-файли) та Linux (Shell-скрипти).

## Мета:

- Навчитися створювати прості сценарії у Windows та Linux.
- Освоїти базові конструкції: послідовність команд, змінні, введення-виведення, умови, цикли.
- Ознайомитися з передачею аргументів, роботою з файлами та обробкою помилок.
- Навчитися запускати скрипти з командного рядка.

## Звітність:

- Знімки екрану кожного завдання
- Скріншот коду, А файл з кодом додаткового завдання

## Частина 1. Windows (BAT-файли)

### Завдання 1. «Hello, World!»

Створіть файл hello.bat з таким вмістом:

```
@echo off
echo Hello, World!
pause
```

### Завдання 2. Змінні та введення користувача

Створіть файл user.bat з таким вмістом:

```
@echo off
set /p name=Enter your name:
echo Hello, %name%!
pause
```

### Завдання 3а. Умови (if)

Створіть файл if\_example.bat з таким вмістом:

```
@echo off
set /p num=Enter a number:
if %num% GTR 10 (
    echo The number is greater than 10
)
```

```
) else (  
    echo The number is not greater than 10  
)  
pause
```

### **Завдання 3б. Цикли (for)**

Створіть файл loop.bat з таким вмістом:

```
@echo off  
for /l %%i in (1,1,5) do echo %%i  
pause
```

### **Завдання 4. Передача аргументів у BAT-файл:**

```
@echo off  
echo First argument: %1  
echo Second argument: %2  
pause
```

### **Завдання 5. Арифметика:**

```
@echo off  
set /p a=Enter first number:  
set /p b=Enter second number:  
set /a c=%a% + %b%  
echo Sum: %c%  
pause
```

### **Завдання 6. Робота з файлами:**

```
@echo off  
echo This is a new line > test.txt  
notepad test.txt
```

### **Завдання 7. Масив і цикл:**

```
@echo off  
for %%n in (John Alice Peter) do (  
    echo Hello, %%n!  
)  
pause
```

## **Частина 2. Linux (Shell-скрипти)**

chmod +x <ім'я фалу>.sh - дозволити запускати

## **Завдання 1. «Hello, World!»**

Створіть файл hello.sh з таким вмістом:

```
#!/bin/bash
echo "Hello, World!"
```

## **Завдання 2. Змінні та введення користувача**

Створіть файл user.sh з таким вмістом:

```
#!/bin/bash
echo -n "Enter your name: "
read name
echo "Hello, $name!"
```

## **Завдання 3а. Умови (if)**

Створіть файл if\_example.sh з таким вмістом:

```
#!/bin/bash
echo -n "Enter a number: "
read num
if [ $num -gt 10 ]; then
    echo "The number is greater than 10"
else
    echo "The number is not greater than 10"
fi
```

## **Завдання 3б. Цикли (for)**

Створіть файл loop.sh з таким вмістом:

```
#!/bin/bash
for i in {1..5}
do
    echo $i
done
```

## **Завдання 4. Передача аргументів у скрипт:**

```
#!/bin/bash
echo "First argument: $1"
echo "Second argument: $2"
```

## **Завдання 5. Арифметичні обчислення:**

```
#!/bin/bash
echo -n "Enter first number: "
read a
```

```
echo -n "Enter second number: "  
read b  
sum=$((a+b))  
echo "Sum: $sum"
```

### **Завдання 6. Робота з файлами:**

```
#!/bin/bash  
echo "Enter file name:"  
read filename  
if [ -f "$filename" ]; then  
    echo "File exists"  
else  
    echo "File not found, creating..."  
    echo "New file" > $filename  
fi
```

### **Завдання 7. Масив і цикл:**

```
#!/bin/bash  
names=("John" "Alice" "Peter")  
for n in "${names[@]}"  
do  
    echo "Hello, $n!"  
done
```

## **Додаткове завдання**

У цьому завданні потрібно створити скрипт, який:

1. **Отримусь прізвище студента з аргументу командного рядка.**  
(Приклад запуску: `grade.bat Smith` або `./grade.sh Smith`)
2. **Запитує у користувача бажану оцінку за роботу.**  
(Користувач вводить число з клавіатури.)
3. **Запитує, чи потрібно зберегти результат у файл.**  
(Відповідь користувача: `yes` або `no`.)

Якщо користувач відповів `yes`, скрипт створює файл `result.txt`, у який записує:

`Прізвище Оцінка`

Наприклад:

Smith 95

4. Якщо користувач відповів **no**, скрипт виводить повідомлення, що результат не збережено.