

# RPS DAY 4 Assignment

## Assignment 1

**Name: Akshada Baad**

**Batch - CPPE**

**1: Ensure the script checks if a specific file (e.g., myfile.txt) exists in the current directory. If it exists, print "File exists", otherwise print "File not found".**

**Step1: Create a file using the touch command**

```
touch file1.txt
```

**Step2: create one variable and store the filename in that**

```
filename="file1.txt"
```

**Step3: Check if the file exists in the current directory**

```
if test -f "$filename"; then
```

```
    echo "File exists"
```

```
else
```

```
    echo "File not found"
```

```
fi
```



The screenshot shows a web-based interface for an online Bash shell. At the top, there's a navigation bar with icons for Run, Debug, Stop, Share, Save, and Beautify. Below this is a code editor with a dark theme. The code in the editor is a Bash script that creates a file, sets a variable, and checks if the file exists. The script is as follows:

```
1 # Online Bash Shell.
2 # Code, Compile, Run and Debug Bash script online.
3 # Write your code in this editor and press "Run" button to execute it.
4
5
6 # touch file1.txt
7
8 filename="file1.txt"
9
10 # Check if the file exists in the current directory
11 if test -f "$filename"; then
12     echo "File exists"
13 else
14     echo "File not found"
15 fi
16
17
```

Below the code editor is a terminal window. It shows the output of the script: "File exists". At the bottom of the terminal, it says "...Program finished with exit code 0" and "Press ENTER to exit console.".

## Assignment 2

**2: Write a script that reads numbers from the user until they enter '0'. The script should also print whether each number is odd or even.**

**#Step1: Write a Function to check if a number is odd or even**

```
#!/bin/bash
check_odd_even() {
    local number=$1
    if (( number % 2 == 0 )); then
        echo "The number $number is even."
    else
        echo "The number $number is odd."
    fi
}
```

**#Step2: Create an Infinite loop to read numbers from the user**

```
while true; do
```

**#Step3: Read a number from the user**

```
    read -p "Enter a number (0 to stop): " number
```

**#Step4: Check if the user entered 0**

```
    if [ "$number" -eq 0 ]; then
        echo "Exiting..."
        break
    fi
```

**#Step5: Check if the entered value is a valid number**

```
    if ! [[ "$number" =~ ^-[0-9]+$ ]]; then
        echo "Invalid input. Please enter a valid number."
        continue
    fi
```

**#Step6: Call the function to check if the number is odd or even**

```
    check_odd_even $number
done
```

onlinegdb.com/online\_bash\_shell

RunDebugStopShareSaveBeautify

main bashfile1.txt

```
5
6 #!/bin/bash
7
8 # Function to check if a number is odd or even
9 check_odd_even() {
10     local number=$1
11     if (( number % 2 == 0 )); then
12         echo "The number $number is even."
13     else
14         echo "The number $number is odd."
15     fi
16 }
17
18 # Infinite loop to read numbers from the user
19 while true; do
20     # Read a number from the user
21     read -p "Enter a number (0 to stop): " number
22
23     # Check if the user entered 0
24     if [ "$number" -eq 0 ]; then
25         echo "Exiting..."
26         break
27     fi
28
```

input

```
Enter a number (0 to stop): 25
The number 25 is odd.
Enter a number (0 to stop): 63
The number 63 is odd.
Enter a number (0 to stop): 55
The number 55 is odd.
Enter a number (0 to stop): 2
The number 2 is even.
Enter a number (0 to stop): 0
Exiting...
```