BASH Scripting & AWS

BASH- **B**ourne **A**gain **SH**ell
it is command line interpreter and scripting language commonly used in UNIX-Based Systems like Linux and Mac OS.
it Allows users to automate the tasks, Manage Syatem Opertaions, and Write Scripts for repetitive task
SHEBANG- #! /bin/bash
Step:1 Create a File

```
nano hello.sh
```

Step:2 Create Code and Execute

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

Step:3 Make the Script Executable

```
chmod +x hello.sh //hello.sh will be the name of your file with extension
```

Step:4 Run The Scrip

```
./hello.sh
```

Output:

```
Hello World!
```

Example :2

```
#!/bin/bash

echo "Enter Your Name"
read user_name

echo "Hello , $user_name"
```

Example:3 (IF-Else Statement)

#!/bin/bash

```
echo "Enter a Number:"
read num

if [ $num -gt 10 ]; then
echo "The Number is greater than 10"
else
echo "The Number is 10 or less"
fi
```

Example:4 (For Loop)

```
#!/bin/bash

for i in {1..5}; do
echo "Iteration $i"
done
```

Example:5 (While Loop)

```
#!/bin/bash
count=1
while [ $count -le 5 ]; do
echo "Count: $count"
((count++))
done
```

Example:6 Function in Bash

```
#!/bin/bash

greet(){
echo "Hello. $1!"
}

greet "Nikunj"
```

Example:7 Working with Arguments

```
#!/bin/bash
```

```
echo "Script Name:$0"
echo "First Argument:$1"
echo "Third Argument:$2"


run the commnad: ./script.sh arg1 arg2
```

Example:8 File Operation

```
#!/bin/bash

file="example.txt"

if [ -f "$file" ]; then
echo "File Exist"
else
echo "File Does not Exist."
fi
```

Example: 9 Read Data Line By Line

```
#!/bin/bash

while read line; do
echo "$line"
done <example.txt
```

Project: Time 10-15 Minutes
Demonstrating writing a Simple Bash Script to automate repetitive task (Eg. generating multiple files);

1. **Step:1** Create Automatic Directory
2. **Step:2** Create Script That Will Automates n no of file creation within the Directory
3. **Step:3** Print the message of File Completion

Solution:

```
GNU nano 7.2 project.sh #!/bin/bash


DIR="my_files"
```

```
NUM_OF_FILES=4


mkdir -p $DIR

for i in $(seq 1 $NUM_OF_FILES); do
FILE="$DIR/file_$i.txt"
echo "This is file number $i" > $FILE
echo "Created $FILE"
done

echo "File creation completed!"
```