

Question 1: Write a shell script that prints "Hello, World!" to the terminal.

Ans- `echo "Hello, World!"`

Question 2: Declare a variable named "name" and assign the value "CDAC Mumbai" to it. Print the value of the variable.

Ans-
`name = "CDAC MUMBAI"`

`echo $name`

Question 3: Write a shell script that takes a number as input from the user and prints it.

Ans- `echo "enter a number:"`
`read number`
`Echo "the entered number is : $number"`

.

Question 4: Write a shell script that performs addition of two numbers (e.g., 5 and 3) and prints the result

```
.
num1=5

num2=3

result=$((num1 + num2))

echo "The sum of $num1 and $num2 is: $result"
```

Question 5: Write a shell script that takes a number as input and prints "Even" if it is even, otherwise prints "Odd".

```
echo "Please enter a number:"
read number
if (( number % 2 == 0 ));
then
echo "Even"
else
echo "Odd"
fi
```

Question 6: Write a shell script that uses a for loop to print numbers from 1 to 5.

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

Question 7: Write a shell script that uses a while loop to print numbers from 1 to 5.

```
i=1
while [ $i -le 5 ]
do
    echo $i
    ((i++))
done
```

Question 8: Write a shell script that checks if a file named "file.txt" exists in the current directory. If it does, print "File exists", otherwise, print "File does not exist".

```
if [ -e "file.txt" ];
then
    echo "File exists"
else
    echo "File does not exist"
fi
```

Question 9: Write a shell script that uses the if statement to check if a number is greater than 10 and prints a message accordingly.

```
echo "Please enter a number:"
read number
if [ $number -gt 10 ];
then
echo "The number is greater than 10."
else
echo "The number is not greater than 10."
fi
```

Question 10: Write a shell script that uses nested for loops to print a multiplication table for numbers from 1 to 5. The output should be formatted nicely, with each row representing a number and each column representing the multiplication result for that number

```
for i in {1..5}
do
for j in {1..5}
do
printf "%2d" $((i * j))
```

```
done
    echo
done.
```

Question 11: Write a shell script that uses a while loop to read numbers from the user until the user enters a negative number. For each positive number entered, print its square. Use the break statement to exit the loop when a negative number is entered.

```
echo "Please enter a number:"
    read number
    if [ $number -lt 0 ];
    then
        echo "Negative number entered. Exiting..."
        break
    fi
    square=$((number * number))
    echo "The square of $number is: $square"
done
```