Hello World & Basics

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

Variables

```
name="Mayank"
age=30
echo "My name is $name and I am $age years old."

readonly constVar=100
unset name
```

User Input

```
read username
echo "You entered: $username"

read -p "Enter age: " age
read -sp "Enter password: " password
```

Command Substitution

```
today=$(date)
user=`whoami`
echo "Today is $today and logged in as $user."
```

Arithmetic Operations

```
a=10
b=3
echo $((a + b))
echo $((a - b))
echo $((a * b))
echo $((a / b))
echo $((a % b))

echo "scale=2; 5/3" | bc
```

Conditional Statements

```
num=10
if [ $num -gt 5 ]; then
  echo "Greater than 5"
elif [ $num -eq 5 ]; then
  echo "Equal to 5"
else
  echo "Less than 5"
fi
```

Case Statement

```
read -p "Enter a character: " char
case $char in
  [a-z]) echo "Lowercase";;
  [A-Z]) echo "Uppercase";;
  [0-9]) echo "Number";;
  *)     echo "Other";;
esac
```

Loops

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

count=1
while [ $count -le 5 ]; do
  echo "Count: $count"
  ((count++))
done

x=1
until [ $x -gt 5 ]; do
  echo "x=$x"
  ((x++))
done
```

Functions

```
greet() {
  echo "Hello, $1"
}
greet "Mayank"

add() {
  echo $(( $1 + $2 ))
}
sum=$(add 5 7)
echo "Sum: $sum"
```

Arrays

```
fruits=("apple" "banana" "cherry")
echo ${fruits[0]}
echo ${fruits[@]}
echo ${#fruits[@]}
fruits+=("mango")
unset fruits[1]
```

String Operations

```
str="Hello World"
```

```
echo ${#str}
echo ${str:0:5}
echo ${str/World/Bash}
echo ${str,,}
echo ${str^^}
```

File Handling

```
filename="test.txt"
while read line; do
  echo "$line"
done < "$filename"

echo "New Line" > file.txt
echo "Append Line" >> file.txt
```

Command Line Arguments

```
echo "Script Name: $0"
echo "First Arg : $1"
echo "Second Arg: $2"
echo "All Args  : $@"
echo "Arg Count : $#"
```

Exit Status & Error Handling

```
ls /etc/passwd
echo $?
command || echo "Command failed"
command && echo "Command succeeded"
```

Debugging

```
bash -x script.sh
set -x
set +x
```

Useful One-Liners

```
for f in *.txt; do echo "$f"; done
find . -type f -name "*.log"
ps aux | grep bash
```

Associative Arrays

```
declare -A capitals
capitals[India]="New Delhi"
capitals[France]="Paris"

echo ${capitals[India]}
echo ${!capitals[@]}   # Keys
```

```
echo ${capitals[@]}    # Values
```

Environment Variables

```
echo $PATH
export MYVAR="Hello"
echo $MYVAR
```

HereDocs and HereStrings

```
cat <<EOF
This is a multiline text block.
Line 2
EOF

cat <<< "Single line input using HereString"
```

Redirection & Pipes

```
echo "Hello" > file.txt     # Overwrite
echo "World" >> file.txt    # Append
cat file.txt | grep "Hello" # Pipe
ls notexist 2> errors.log   # Redirect stderr
```

Process Management

```
sleep 100 &      # Run in background
jobs            # List jobs
fg %1           # Bring job to foreground
bg %1           # Resume job in background
kill -9 1234    # Kill process with PID
```

Trap Signals

```
trap "echo Interrupted; exit" SIGINT SIGTERM

while true; do
  echo "Running..."
  sleep 2
done
```

Subshells & Grouping

```
(cd /tmp && ls)    # Subshell
{ echo "one"; echo "two"; } > file.txt
```

Regular Expressions

```
str="hello123"
if [[ $str =~ ^hello[0-9]+$ ]]; then
```

```
  echo "Pattern matched"
fi
```

Advanced Parameter Expansion

```
var="hello world"
echo ${var:-default}     # If var unset, use default
echo ${var:0:5}          # Substring
echo ${var/world/bash}   # Replace
echo ${var#hello}        # Remove prefix
echo ${var%world}        # Remove suffix
```