The read command in a shell script is used to read input from the user and store it in a variable. Here are some examples of how you can use the read command in various situations:

Basic Input:

#!/bin/bash

echo "Enter your name:"

read name

echo "Hello, $name!"

1.Prompt with a Message:

#!/bin/bash

read -p "Enter your age: " age

echo "You are $age years old."

2.Reading Multiple Values:

#!/bin/bash

read -p "Enter your first name: " first\_name

read -p "Enter your last name: " last\_name

echo "Full name: $first\_name $last\_name"

Reading Passwords (with hidden input):

#!/bin/bash

read -s -p "Enter your password: " password

echo # Print a new line after password input

echo "Password accepted."

3.Reading Input into an Array:

#!/bin/bash

echo "Enter your favorite fruits, separated by spaces:"

read -a fruits

echo "Your favorite fruits are: ${fruits[@]}"

4.Setting a Timeout for Input:

#!/bin/bash

read -t 5 -p "You have 5 seconds to enter something: " user\_input

if [ -z "$user\_input" ]; then

echo "Time's up!"

else

echo "You entered: $user\_input"

fi

5.Using a Default Value:

#!/bin/bash

read -p "Enter your favorite color [blue]: " favorite\_color

favorite\_color=${favorite\_color:-blue}

echo "Your favorite color is $favorite\_color."

7.Using a Delimiter:

#!/bin/bash

IFS=',' # Set the Internal Field Separator to comma

read -p "Enter a list of items separated by commas: " item\_list

IFS=' ' # Reset the Internal Field Separator to default space

items=($item\_list) # Convert the input string into an array

echo "You entered ${#items[@]} items:"

for item in "${items[@]}"; do

echo "- $item"

done