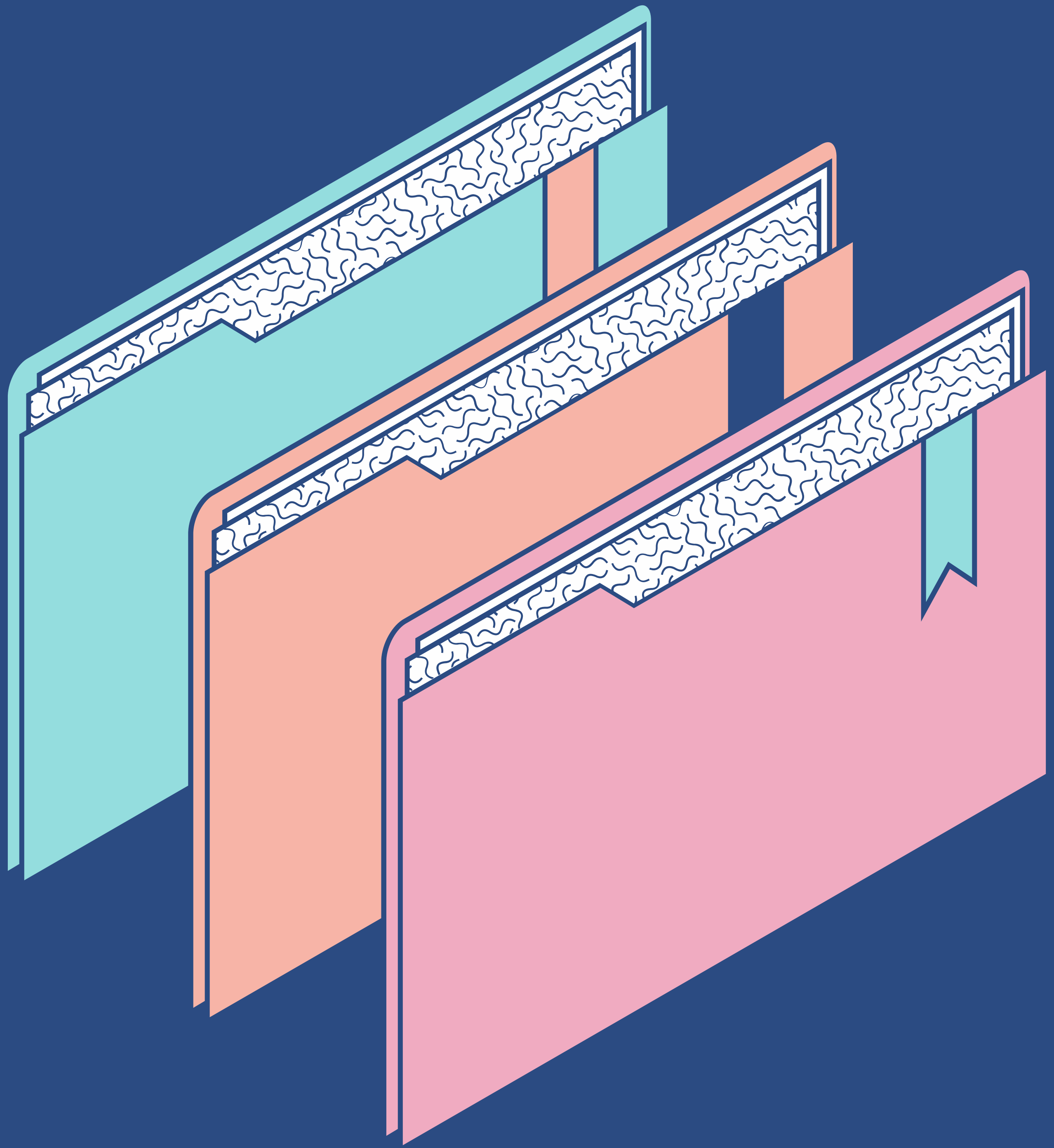
BASH SCRIPTING - BEGINNER TO INTERMEDIATE LEVEL

```
#!/usr/bin/env bash
echo "Hello world."
```

SHRAVAN DWARKA

System Administrator



Agenda

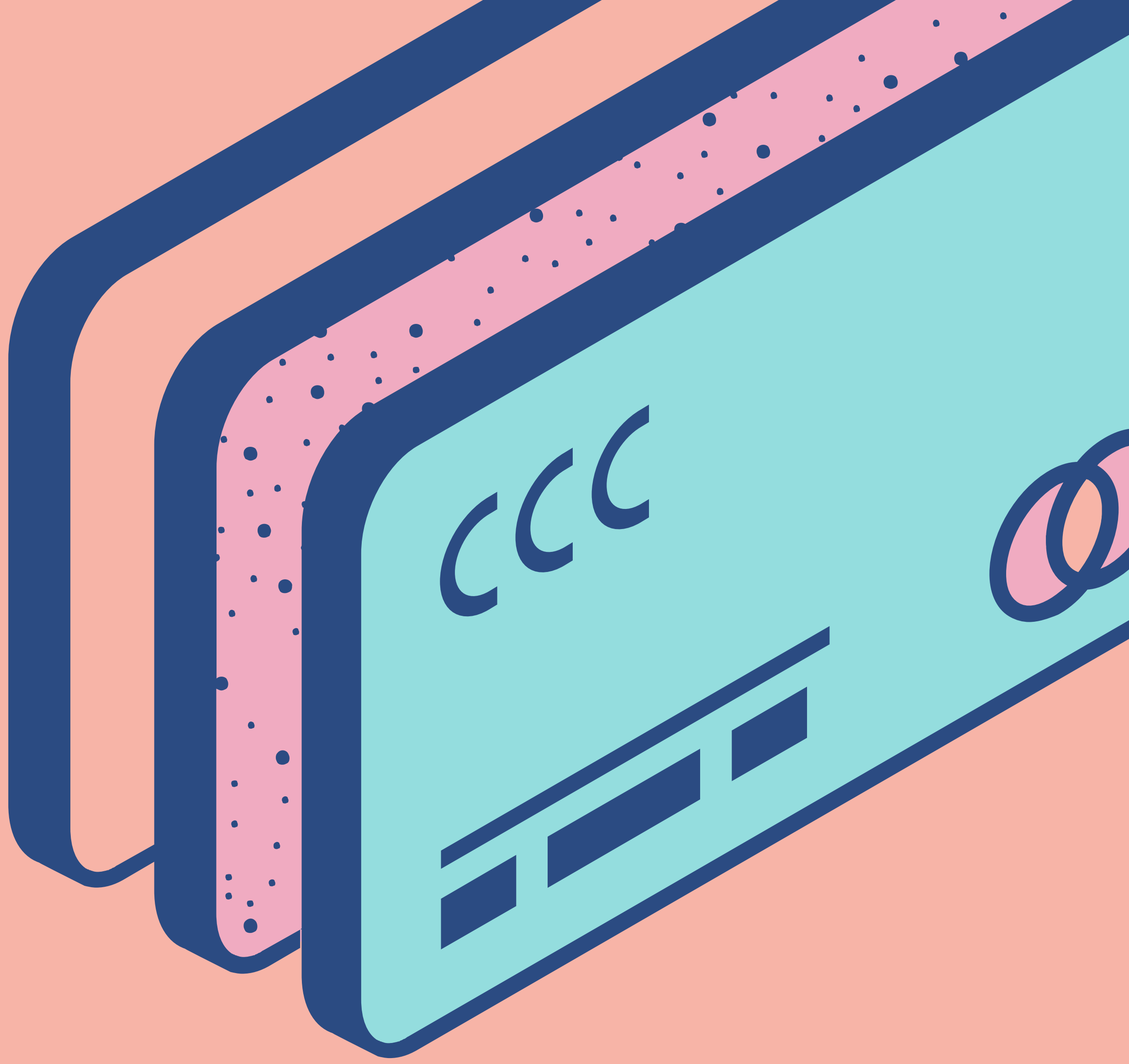
KEY TOPICS DISCUSSED IN THIS PRESENTATION

- What is Bash?
- Sets of commands in Bash.
- How to start writing a Bash script?
- Tip & tricks around Bash (levelling up 📈).
- Demonstration.
- Question time.

BASH - Bourne Again SHell

- Command language interpreter for GNU OS
- Application to run commands to instruct the OS
- Default shell on current Unix systems
- Can be used to write scripts and perform automated tasks - more or less same concepts as .bat file in Windows.

Set of commands in Bash



Command	Description
cd	change directory, `cd /home/shravan/Desktop`
ls	list, `ls /home/shravan` will display contents inside /home/shravan/directory
pwd	present working directory, pwd will return the current directory you're in
cat	concatenate, outputs the contents of a file on the terminal
less	opens a file so you can use your cursor to browse through
mv	move, either move or rename a file
rm	remove, deletes a file

How to start writing Bash scripts?

LET'S TRY HELLO WORLD

Shebang

A script needs to know what shell to use to execute the commands in the script. Hence, the use of shebang.
#!/bin/bash or #!/usr/bin/env bash, will inform the script to use Bash to execute its commands.

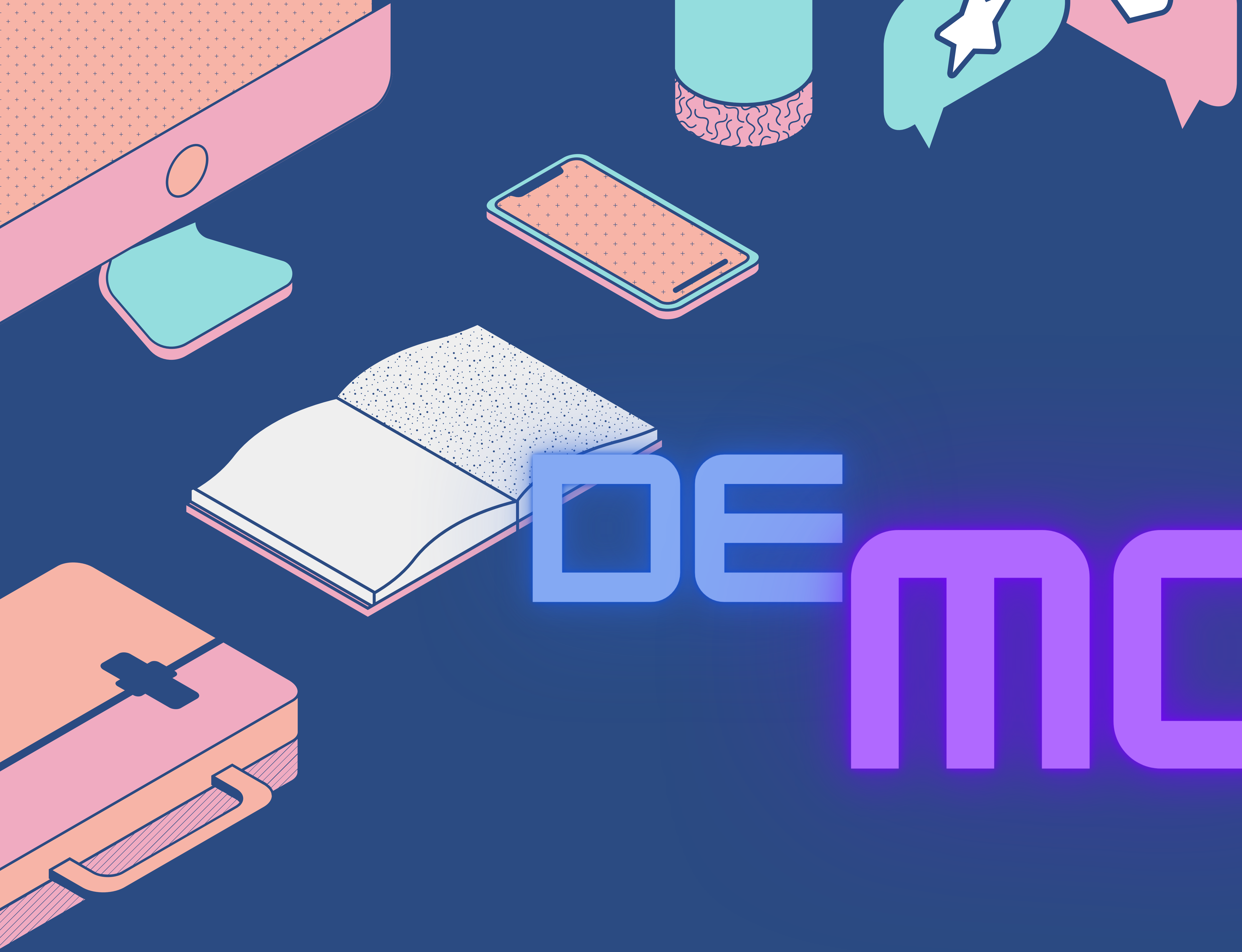
Commands

We can use either the full path of the command or only the command name, depending on whether the path is defined in the $PATH variable. Or, we can use some built-in shell commands.

```
#!/usr/bin/env bash
echo "Hello world!"
```

Tips & tricks

Loops	Using for and while loops to do repetitive task
Aliases	Stored in ~/.bash_aliases, your shortcuts
Functions	One function to perform multiple task (example, purchasing a song and downloading it, and placing it in your Musics directory)
Arrays	One of my most used features in scripting Usage: readarray -t arrayName < <(cmd) Test: for i in ${arrayName[@]}; do echo $i; done
Logs	Keep track of what your script is doing on a daily basis, and add monitoring on those logs to keep you aware



DEMO

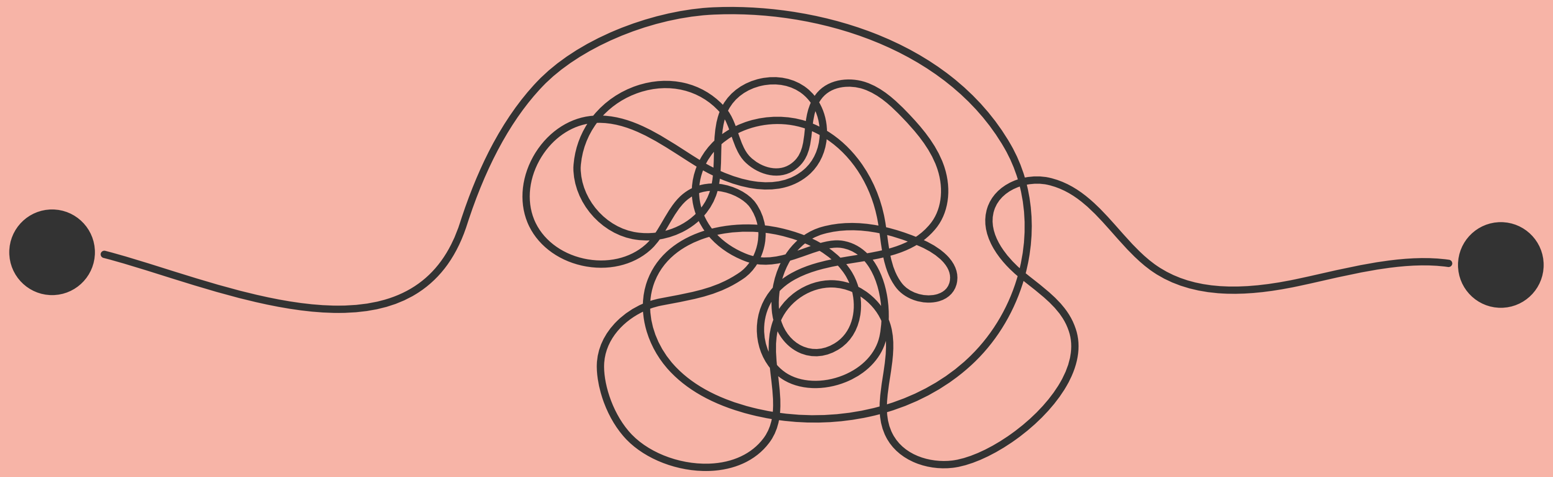
The last slide contains all the scripts used for this presentation.

Scenario

You are a Database Administrator. You have to perform daily backups for your database.

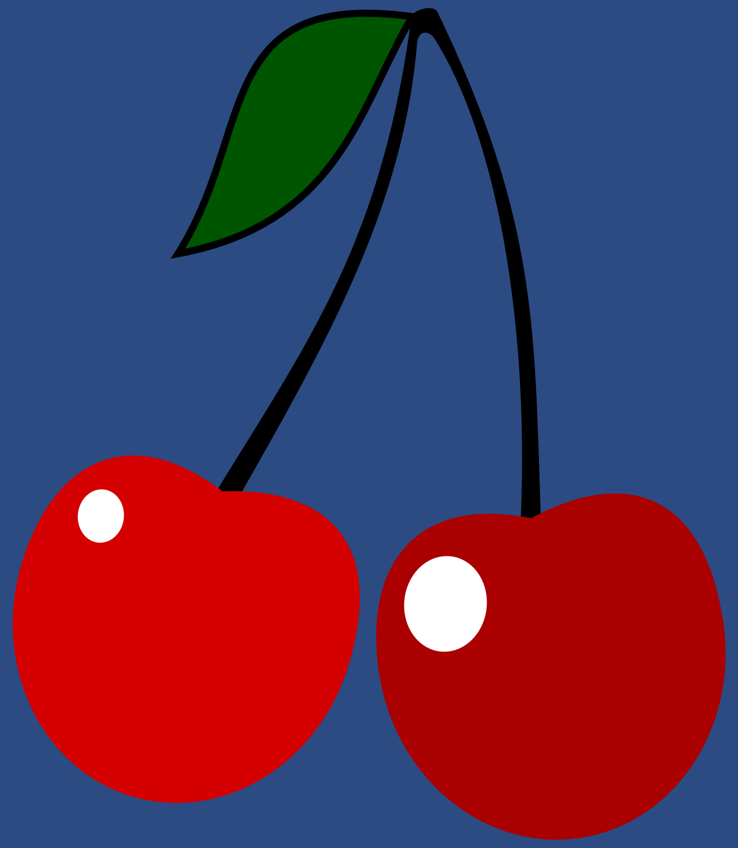
Ideally with little or no human interaction.

Your BASH script should be able to also handle re-arranging and retention so that the server's disk is filled with backups.



Cherry on the cake

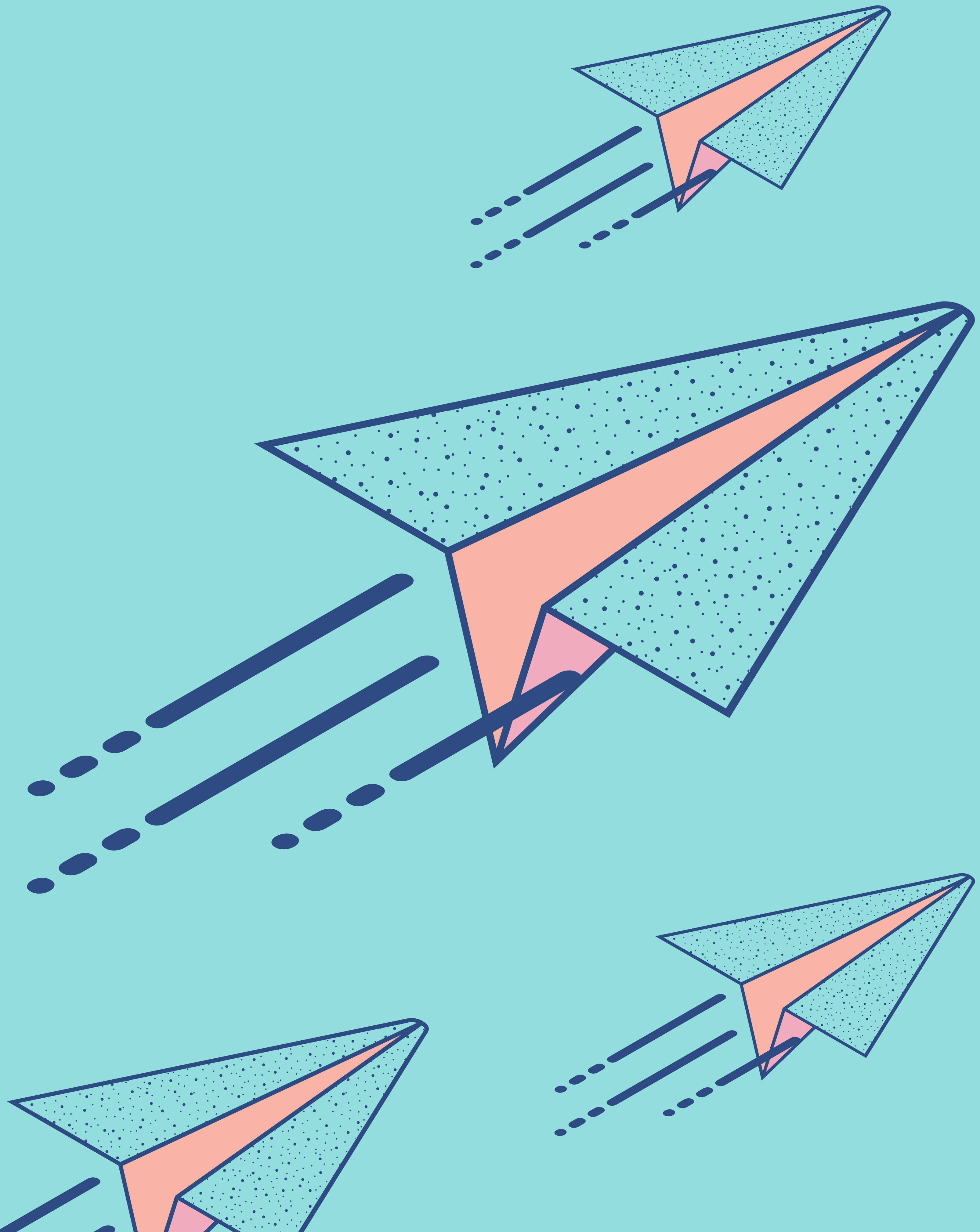
SHELLCHECK !



Y U NO
USE BASH
imgflip.com

Do you have any questions?

Avez-vous des questions ?



1.Hello World script

```
#!/usr/bin/env bash
echo "Hello world!
```

2. For loop

```
#!/usr/bin/env bash
MAX=10

for i in $(seq 1 $MAX); do
 echo "Value: $i"
done
```

3. While loop

```
#!/usr/bin/env bash
MAX=10
i=1

while [[ "$i" -le "$MAX" ]]; do
 echo "Value: $i"
 ((i++))
done
```

4. Trap (clean up)

```
#!/usr/bin/env bash
trap "fnCleanUp" EXIT

fnCleanUp() {
 if ls tmp/2_*.tmp > /dev/null 2>&1; then
  echo "Temporary files found."
  rm -vf tmp/2_*.tmp
  echo "Temporary files deleted."
 fi
}

for i in {1..5}; do
 tmpName=$(LC_ALL=C /usr/bin/tr -dc 'A-Za-z0-9' < /dev/urandom | head -c 5)
 touch tmp/2_file-"$tmpName".tmp
 echo "File created in tmp/2_file-$tmpName.tmp."
done

sleep 60
```

5. Bash options

```
#!/usr/bin/env bash

set -e
set -o nounset
set -o pipefail

executeThisCommandWhichDoesNotExist # set -e will stop executing the script right away
echo "$toto" # set -o nounset will stop executing the script here

# Scenario for pipefail
cat tmp/3/3_test.txt 2>&1 | grep -c "something"
if [[ "$?" -eq 0 ]]; then
 echo "Pipe succeeded"
else
 echo "Pipe failed"
fi
```

6. Logging

```
#!/usr/bin/env bash

log() {
 [[ "$DEBUG" -eq 1 ]] && echo "[DEBUG] $*"
}

log "Before if"
if [[ 2 -gt 1 ]]; then
 log "Inside if"
fi
log "After if"
```