ANNEX A · FIRST BASH SCRIPTS

NAME _____ **DATE** _____ **GROUP** _____

Work in the `deb-lab` VM, in a terminal. Type the commands, do not paste them. In `nano`: save with `Ctrl+O` then `Enter`, exit with `Ctrl+X`. Start with: `mkdir -p ~/scripts && cd ~/scripts`

TASK 1 · HELLO.SH

1. `nano hello.sh`, write these lines, save, exit.

```
#!/bin/bash
# hello.sh - my first script
echo "Hello from $(hostname)"
date
```

2. Run it: `./hello.sh`. It fails. Copy the error:

_____

3. `chmod +x hello.sh` then `ls -l hello.sh`.
   Which letter is new? _____
4. Run `./hello.sh` again. Then run it without the `./`.
   The shell answers:

_____

DONE WHEN `./hello.sh` prints the name of your machine and the date.

TASK 2 · VARIABLES AND QUOTES

1. `nano greet.sh`:

```
#!/bin/bash
name="Marta"
today=$(date +%A)
echo "Hello, $name. Today is $today."
echo 'Single quotes: $name'
```

2. **Before** you run it: what will the last line print?

_____

3. `chmod +x greet.sh` and run it. Change line 2 to `name = "Marta"` and run it again. Error:

_____

   Then fix it.

DONE WHEN It greets Marta with today's weekday, and you can explain the last line.

TASK 3 · ARGUMENTS

1. `nano args.sh`:

```
#!/bin/bash
echo "Script name: $0"
echo "First:  $1"
echo "Second: $2"
echo "How many: $#"
```

2. `chmod +x args.sh`. Run `./args.sh report.txt /tmp`.
   How many? _____
3. Run `./args.sh "my file.txt"`. How many? _____
   And without the quotes? _____

DONE WHEN You can say what `$1`, `$2` and `$#` hold.

TASK 4 · EXIT CODES AND IF

1. Run each command, then `echo $?`. Write the number.

   `true` _____ `ls /nope` _____

   `grep -q root /etc/passwd` _____

   `grep -q marta /etc/passwd` _____

   `nosuchcommand` _____

2. `nano check.sh`:

```
#!/bin/bash
if [ -d "$1" ]; then
    echo "$1 is a directory"
else
    echo "$1 is not a directory"
    exit 1
fi
```

3. `chmod +x check.sh`. Run `./check.sh /etc; echo $?` and `./check.sh /etc/passwd; echo $?`.
   Exit codes: _____ and _____

DONE WHEN It answers correctly for a folder and for a file, with exit codes 0 and 1.

TASK 5 · FOR

1. Create the sample folders of Tramuntana Apps:

```
mkdir -p ~/tramuntana/{dev,design,admin}
echo "print(42)" > ~/tramuntana/dev/app.py
echo "<svg/>" > ~/tramuntana/design/logo.svg
echo "2026-001;Hotel Bellver;1200" > ~/tramuntana/admin/invoices.csv
```

2. Type this loop in the terminal (`Enter` after each line). How many lines does it print? _____

```
for f in ~/tramuntana/*/*; do
    echo "$f: $(wc -c < "$f") bytes"
done
```

DONE WHEN You see one line per file, with its size in bytes.

TASK 6 · BACKUP.SH (IN PAIRS)

1 `nano backup.sh`. Copy the script and fill the six gaps (a)–(f).

```
#!/bin/__(a)__
# backup.sh - back up one folder into a dated .tar.gz
# Usage: backup.sh SOURCE DEST

if [ $# -ne __(b)__ ]; then
    echo "Usage: $0 SOURCE DEST"
    exit __(c)__
fi

src="$1"
dest="$2"

if [ ! __(d)__ "$src" ]; then
    echo "Error: $src is not a directory"
    exit 2
fi

mkdir -p "$dest"
stamp=$(__(e)__ +%Y%m%d-%H%M%S)
name=$(basename "$src")
file="$dest/$name-$stamp.tar.gz"

if tar -czf "$file" -C "$(dirname "$src")" "$name"; then
    __(f)__ "$file" > "$file.sha256"
    echo "$(date '+%F %T') OK $file"
else
    echo "$(date '+%F %T') ERROR tar failed for $src"
    exit 3
fi
```

2 `chmod +x backup.sh`. Run it three times; after each run write the exit code (`echo $?`):

`./backup.sh` _____ `./backup.sh ~/nothing ~/backups` _____

`./backup.sh ~/tramuntana/dev ~/backups` _____

3 Check the result: `ls ~/backups` then `sha256sum -c ~/backups/*.sha256` then `tar -tzf ~/backups/dev-*.tar.gz`

DONE WHEN The three exit codes are 1, 2 and 0, `sha256sum -c` answers `OK`, and `tar -tzf` lists `dev/app.py`.

EXIT TICKET

Tick one answer for each question.

1 What is the first line `#!/bin/bash` for?

- ☐ A. It is a comment with the author's name.
- ☐ B. It says which program must read the file.
- ☐ C. It makes the file executable.
- ☐ D. It imports the Bash commands.

2 `./backup.sh` answers "Permission denied". What do you do?

- ☐ A. Rename it to `backup.exe`.
- ☐ B. Run `bash --version`.
- ☐ C. Run `chmod +x backup.sh`.
- ☐ D. Delete the first line.

3 Why does `name = "Pau"` fail?

- ☐ A. The spaces make `name` look like a command.
- ☐ B. Variables must be in capital letters.
- ☐ C. The quotes must be single quotes.
- ☐ D. A variable needs `$` when you create it.

4 A command ends with exit code 2. What does that tell you?

- ☐ A. It worked, with two results.
- ☐ B. It failed.
- ☐ C. It ran twice.
- ☐ D. It needs two arguments.

5 In `./backup.sh ~/tramuntana/dev ~/backups`, what is `$#`?

- ☐ A. `./backup.sh`
- ☐ B. `~/backups`
- ☐ C. 2
- ☐ D. 3