ANNEX A · PRACTICAL 4.8 · SCHEDULING BACKUP.SH

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

You need `deb-lab` with your `~/scripts/backup.sh`, and later `win11`. Write your answers on this sheet. **Hand in** on Classroom: `practical-4.8-surname.md` with the outputs marked **REPORT**. Deadline: Thursday 11/02, 23:59. Marks:

Task 1, 2 · Part A, 3 · Part B, 2.5 · Part C, 1.5 · table and report, 1.

TASK 1 · READ AND WRITE CRON LINES

The five fields: **minute · hour · day of month · month · day of week**. When does each line run?

`0 8 * * *` _____

`*/15 * * * *` _____

`30 2 * * 1-5` _____

`0 0 1 * *` _____

Write the five fields for each schedule.

Every day at 20:00 _____

Every Sunday at 03:30 _____

Every 10 minutes _____

At 14:30, Monday to Friday _____

PART A · CRON (DEB-LAB)

1. Check that the script works by hand: `~/scripts/backup.sh ~/tramuntana/dev ~/backups; echo $?` Exit code: _____
2. Open your crontab with `crontab -e` (choose nano). Add this line at the end, in **one** line. Save and exit.

```
*/2 * * * * $HOME/scripts/backup.sh $HOME/tramuntana/dev $HOME/backups >> $HOME/backup.log 2>&1
```

3. Wait about four minutes (do Task 1 meanwhile). Then: `cat ~/backup.log` and `ls ~/backups`.
   How many `OK` lines? _____ How many minutes between them? _____
4. Why did we write `$HOME/scripts/backup.sh` and not just `backup.sh`?

   _____

5. `crontab -e` again: change the five fields so that it runs **every day at 20:00**. Check with `crontab -l`. **REPORT**

DONE WHEN `backup.log` has at least two `OK` lines two minutes apart, and `crontab -l` shows the 20:00 line.

PART B · SYSTEMD TIMER (DEB-LAB)

1. `mkdir -p ~/.config/systemd/user`. Then `nano ~/.config/systemd/user/backup.service`:

```
[Unit]
Description=Back up the dev folder

[Service]
Type=oneshot
ExecStart=%h/scripts/backup.sh %h/tramuntana/dev %h/backups
```

2. `nano ~/.config/systemd/user/backup.timer`:

```
[Unit]
Description=Run backup.service on a schedule

[Timer]
OnCalendar=*:0/2
Persistent=true

[Install]
WantedBy=timers.target
```

3. Tell systemd to read the new files: `systemctl --user daemon-reload`
4. Test the **what**, without waiting: `systemctl --user start backup.service` then `journalctl --user -u backup.service -n 3`.
   Do you see an `OK` line? _____
5. Switch on the **when**: `systemctl --user enable --now backup.timer`
6. `systemctl --user list-timers`. Wait two minutes and run it again. What changed in the `LAST` column?

   _____

7 Change `OnCalendar` to `*-*-* 20:00:00`. Then `systemctl --user daemon-reload` and `systemctl --user restart backup.timer`. Paste `systemctl --user list-timers`. **REPORT**

Now cron **and** the timer run at 20:00. Which one do you keep, and why? Delete the other (`crontab -e`, or `systemctl --user disable --now backup.timer`).

DONE WHEN `list-timers` shows `backup.timer` with the next run at 20:00, and the journal has an `OK` line.

PART C · TASK SCHEDULER (WIN11)

Open **Command Prompt** in `win11`.

1 Create a folder to protect, a folder for the copies, and a one-line script (a **batch file**) that makes the copy:

```
mkdir C:\tramuntana\dev C:\backup
echo print(42) > C:\tramuntana\dev\app.py
echo robocopy C:\tramuntana C:\backup\tramuntana /E /LOG+:C:\backup\backup.log > C:
\backup\backup.bat
```

2 Create the task that runs the batch file every day at 20:00:

```
schtasks /create /tn "Tramuntana backup" /tr C:\backup\backup.bat /sc daily /st 20:00
```

3 Do not wait until 20:00. Run it now: `schtasks /run /tn "Tramuntana backup"`

4 Check: `dir C:\backup\tramuntana\dev`. Is `app.py` there? _____

5 Press `Win+R`, type `taskschd.msc`. In **Task Scheduler Library** find your task.

Trigger: _____ Action: _____ Last Run Result: _____

Take a screenshot of the task. **REPORT**

DONE WHEN `app.py` is in `C:\backup\tramuntana\dev` and the task is in the Library with a daily trigger at 20:00.

COMPARE THE THREE TOOLS

	cron	systemd timer	Task Scheduler
Where is the schedule written?			
The machine was off at 20:00. What happens?			
Where does the output go?			

EXIT TICKET

1 When does `0 7 * * 1` run?

- ☐ A. Every Monday at 07:00.
- ☐ B. Every day at 07:01.
- ☐ C. On day 1 of every month at 07:00.
- ☐ D. Every 7 minutes on Mondays.

2 A script works by hand but does nothing from cron. Most likely cause?

- ☐ A. cron only runs scripts owned by root.
- ☐ B. The script has too many lines.
- ☐ C. It calls files or commands without the full path.
- ☐ D. cron needs the script to end in `.cron`.

3 The laptop was off at 20:00. Which tool can run the missed task later?

- ☐ A. cron, always.
- ☐ B. A systemd timer with `Persistent=true`.
- ☐ C. `crontab -l`.
- ☐ D. None of them can.

4 In Windows Task Scheduler, what is the trigger?

- ☐ A. The program that the task starts.
- ☐ B. The user who owns the task.
- ☐ C. The log file of the task.
- ☐ D. The condition that starts the task, such as a time.