



boa3444 / Linux_Lab



<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

[Linux_Lab](#) / [Assignments](#) / [LAB_extra.md](#) 

boa3444 Update LAB_extra.md

a5f8dd8 · now



141 lines (103 loc) · 3.46 KB

Preview

Code

Blame



Raw



🔗 LAB5 - Starter Kit & Automation

Objective: Build a starter project environment automatically.



starter_kit.sh – Script

```
#!/bin/bash
```



```
# Create project directory structure
```

```
mkdir -p project/scripts
```

```
mkdir -p project/docs
```

```
mkdir -p project/data
```

```
# Add placeholder README.md in each subfolder
```

```
touch project/scripts/README.md
```

```
touch project/docs/README.md
```

```
touch project/data/README.md
```

```
# Final message
```

```
echo "Starter Kit Ready!"
```



LAB_extra.md – Documentation

```
# LAB 5 - Starter Kit & Automation
```



```
## Objective
```

```
Automatically set up a starter project environment with organized folders a
```

🛠️ Purpose of `starter_kit.sh`

This script simplifies the setup of a new project by:

- Creating a standard folder structure: `scripts/`, `docs/`, `data/` inside
- Adding a `README.md` file in each folder to encourage documentation from
- Printing a confirmation message once setup is complete

Example Run

```
```bash
$ bash starter_kit.sh
Starter Kit Ready!
```

## Directory Structure After Execution:

```
project/
├── scripts/
│ └── README.md
├── docs/
│ └── README.md
└── data/
 └── README.md
```



## Screenshots:

```
vboxuser@ubuntu: ~/Documents/Linux_Lab/starter_folder/project
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$ ls
starter_kit.sh
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$./starter_kit.sh
bash: ./starter_kit.sh: Permission denied
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$ chmod +x starter_kit.sh
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$./starter_kit.sh
Starter Kit Ready!
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$ ls
project starter_kit.sh
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$ cd project
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder/project$ ls
data docs scripts
```

```
vboxuser@ubuntu:~/Documents/Linux_Lab/starter_folder$ ls -R project
project:
data docs scripts

project/data:
README.md

project/docs:
README.md

project/scripts:
README.md
```

## ? Extra Questions:

### Q1. What does `mkdir -p` do?





- `mkdir -p` creates **nested directories** in one command.
- If the parent directory already exists, it **doesn't throw an error**.
- Example:

```
mkdir -p project/scripts
```



This creates both `project/` and `scripts/` if they don't exist.

### Q2. Why is automation useful in DevOps?

-  **Consistency:** Ensures environments are set up the same way every time.
-  **Speed:** Saves time by eliminating manual setup.
-  **Reproducibility:** Makes it easy to recreate environments for testing or deployment.
-  **Scalability:** Supports large teams and CI/CD pipelines by automating repetitive tasks.



## Appendix: Raw Markdown Source

To ensure reproducibility and peer learning, the full Markdown source of this lab report is included below.

# LAB 5 - Starter Kit & Automation



## ## Objective

Automatically set up a starter project environment with organized folders a

## ## 🛠 Purpose of `starter\_kit.sh`

This script simplifies the setup of a new project by:

- Creating a standard folder structure: `scripts/`, `docs/`, `data/` inside
- Adding a `README.md` file in each folder to encourage documentation from
- Printing a confirmation message once setup is complete

## ## Example Run

```
```bash
```

```
$ bash starter_kit.sh
```

```
Starter Kit Ready!
```

```
```
```

**\*\*Directory Structure After Execution:\*\***

```
```
```

```
project/
```

```
|— scripts/
```

```
|   |— README.md
```

```
|— docs/
```

```
|   |— README.md
```

```
|— data/
```

```
|   |— README.md
```

```
```
```

**\*\*Screenshots:\*\***

\*(Insert your screenshot here showing terminal output and folder tree)\*

## ## Extra Questions

### ### Q1. What does `mkdir -p` do?

- `mkdir -p` creates **\*\*nested directories\*\*** in one command.
- If the parent directory already exists, it **\*\*doesn't throw an error\*\***.

### ### Q2. Why is automation useful in DevOps?

- 🗂 **\*\*Consistency\*\***
- ⚡ **\*\*Speed\*\***
- 🧪 **\*\*Reproducibility\*\***
- 📦 **\*\*Scalability\*\***

