Linux for Data Engineers – Group Practical Activity Brief

Writing an Advanced Automation Script in Linux

Creating a System Health Monitoring Script

1. **Writing the Script**

- Open a text editor and write the following script:

```
#!/bin/bash
# System Health Monitoring Script

# Variables
OUTPUT_FILE="/path/to/system_health_report.txt"
DATE=$(date '+%Y-%m-%d %H:%M:%S')

# Collect System Information
UPTIME=$(uptime -p)
DISK_USAGE=$(df -h /)
MEMORY_USAGE=$(free -h)
```

```
# Write to the output file
echo "System Health Report - $DATE" > $OUTPUT_FILE
echo "" >> $OUTPUT_FILE
echo "Uptime: $UPTIME" >> $OUTPUT_FILE
echo "" >> $OUTPUT_FILE
echo "Disk Usage:" >> $OUTPUT_FILE
echo "$DISK_USAGE" >> $OUTPUT_FILE
echo "" >> $OUTPUT_FILE
echo "Memory Usage:" >> $OUTPUT_FILE
echo "$MEMORY_USAGE" >> $OUTPUT_FILE

# Output result
echo "System health report generated at $OUTPUT_FILE"
```

2. Making the Script Executable

- Change the script's permissions to make it executable:

```
chmod +x /path/to/system_health_monitor.sh
```

3. Running the Script Manually

- Execute the script:

```
./path/to/system_health_monitor.sh
```

4. Scheduling the Script

- Add the script to the crontab to run daily:

```
crontab -e
# Add the following line
0 4 * * * /path/to/system_health_monitor.sh
```