



Debugging Bash Shell Scripts

NarendraP

Learn How to Automate Common Tasks with **Bash** Shell Scripting

Debugging Bash Shell Scripts

- Debugging is determining the cause which fails the script.
- Why script fails ?
 - Because of some errors.
- This is because of two type of errors.
 - Syntax Errors
 - Runtime Errors
- Syntax Errors stops script execution and run time errors don't stop script.
- Actually we don't have good debugging procedures with shell scripting, but we can try with some commands.
- And there are different commands for debugging and we will work with set command.
- One more thing our bash is an interpreter.

Learn How to Automate Common Tasks with **Bash** Shell Scripting

Debugging Bash Scripts...

- We can go with **set** command and We have different options with set command.
- Syntax:
 - `set [options]`
 - **No Options:** To list system defined variables
 - `set -n` No Execution, Purely for syntax check.
 - `set -x` Prints the command before executing it in script
 - **`set -e` Exit Script if any command fails**

Thank you

Debugging Bash Scripts

- We are working with shell scripting and we are using bash as our shell, means OS is executing our shell scripts using bash shell.
- Now here, important point is bash is an interpreter.
- Interpreter is a kind of a program, which is used to execute code/script.
- **How Interpreter(Bash) executes script(s) ?**
- It reads and executes our code/script line by line.
- Sometimes it throws an exception/error while executing code/script.
- This is because of two type of errors.
 - Syntax Errors
 - Runtime Errors
- Syntax Errors stops our script execution and run time errors don't stop our script.
- Here: Debugging is simply, determining the cause which fails the script.
- Actually we don't have good debugging procedures with shell scripting, but we can try with some commands.
- And there are different commands for debugging.

Learn How to Automate Common Tasks with **Bash** Shell Scripting