Shell Script

1. Breve Histórico
   * Criadores: O Shell Script foi desenvolvido como parte do sistema Unix por Ken Thompson e Dennis Ritchie nos anos 1970.
   * Surgiu em 1977 com o Bourne Shell (sh), criado por Stephen Bourne.
   * Evolução:
     + Korn Shell (ksh) nos anos 1980, por David Korn.
     + GNU Bash (bash) em 1989, amplamente utilizado em sistemas Linux.
   * Objetivo: Automatizar tarefas repetitivas e simplificar interações com o sistema operacional.
2. Paradigmas de Programação

* Imperativo:
  + Baseado em comandos sequenciais para manipulação de arquivos, execução de programas e controle de fluxo.
* Funcional:
  + Suporte a funções reutilizáveis para modularidade e organização.
* Procedural:
  + Scripts organizados em blocos de código que executam tarefas específicas.

1. Tipagem de dados
   * Tipagem dinâmica:
     + Variáveis não precisam ser declaradas com tipos específicos.
     + Exemplo:

nome=”Shell Script”

Echo $nome

* Tipos Comuns:
  + Strings, números inteiros e arrays.
* Manipulação de dados:
  + Uso de comandos como awk, sed, e grep para processamento de texto.

1. Aplicação e Diferenciais da Linguagem

Aplicações:

* Automação de tarefas administrativas (backup, monitoramento).
* Configuração de ambiente de desenvolvimento.
* Execução de scripts para compilação e deploy de software.

Diferenciais:

* Integração direta com o sistema operacional.
* Sintaxe simples e semelhantes do terminal.
* Portabilidade entre sistemas Unix / Linux.
* Permite controle avançado sobre processos e arquivos.

1. Exemplos de código

Hello World

#!/bin/bash

echo “Hello World”

Automação Simples (Backup)

#!/bin/bash

# Script para backup automático

data=$(date +%Y-%m-%d)

destino=”/backup/$data”

mkdir -p $destino

cp -r /origem/\* $destino

echo “Backup concluído em $destino”

Controle Condicional

#!/binbash

echo “Digite um número:”

read numero

if [ $numero -gt 10 ]; then

echo “O número é maior que 10.”

else

echo “O número é menor ou igual a 10.”

fi