

# INFORMÁTICA

Linux



Livro Eletrônico



# SUMÁRIO

|  |    |
|--|----|
| Apresentação .....                                     | 3  |
| Linux.....   | 4  |
| Introdução ao Linux.....                               | 4  |
| Licença GPL.....                                       | 4  |
| Ambientes Gráficos .....                               | 5  |
| Distribuições (Distros) GNU/LINUX .....                | 6  |
| Principais Características do Linux.....               | 8  |
| Kernel (Núcleo do Sistema Operacional) .....           | 9  |
| Shell (Interpretador de Comandos).....                 | 10 |
| Usuários e Grupos do Linux .....                       | 11 |
| Arquivos .....   | 12 |
| Identificação de Discos do GNU/Linux .....             | 13 |
| Estrutura de Diretórios do GNU/Linux .....             | 15 |
| Uso de Consoles no Linux .....                         | 18 |
| Comandos do Linux.....                                 | 19 |
| Wildcards (Caracteres Curinga).....                    | 38 |
| Conexão de Comandos.....                               | 39 |
| Redirecionamento de E/S (Entrada/Saída) de Dados ..... | 40 |
| Concatenação de Comandos .....                         | 41 |
| Instalação de Aplicativos .....                        | 42 |
| Instalar e Configurar o DNS .....                      | 42 |
| Navegadores Web Utilizados no Linux.....               | 42 |
| Resumo.....  | 43 |
| Questões Comentadas em Aula .....                      | 49 |
| Questões de Concurso .....                             | 50 |
| Gabarito.....  | 66 |
| Referências.....                                       | 67 |

O conteúdo deste livro eletrônico é licenciado para MARIO LUIS DE SOUZA - 41250799864, vedada, por quaisquer meios e a qualquer título, a sua reprodução, cópia, divulgação ou distribuição, sujeitando-se aos infratores à responsabilização civil e criminal.

## APRESENTAÇÃO

Olá, querido (a) amigo(a), tudo bem?

*"Seja a mudança que você quer ver no mundo".*

*(Mahatma Gandhi)*

Não se contente com a mediocridade e com a mesmice. Todos os dias faça algo para ser melhor na competição. Ao abrir os olhos pela manhã, agradeça por mais uma oportunidade de aperfeiçoar sua vida e repita com convicção a frase: "**sob todos os aspectos eu melhoro a cada dia**". Cada passo dado com coragem e nobreza de intenções, mais nos aproxima de nosso sonho. Reflita nisso.

Que Deus o(a) abençoe e rumo à aula sobre o **sistema operacional Linux**.

Em caso de dúvidas, acesse o fórum do curso ou entre em contato.

Um forte abraço!

*Prof<sup>a</sup> Patrícia Quintão*

*Instagram: @coachpatriciaquintao*

*WhatsApp: (31) 99442.0615*

# LINUX

## INTRODUÇÃO AO LINUX

- O sistema **GNU/Linux** é frequentemente chamado por **Linux**. Foi originalmente construído como um sistema de multitarefas para microcomputadores e mainframes (computadores de grande porte) no meio dos anos 70. Cresceu desde então e tornou-se um dos sistemas operacionais mais usados em qualquer lugar.
- **O Linux é um clone de Unix.** Foi criado como uma alternativa barata e funcional para aqueles que não estão dispostos a pagar o alto preço de um sistema Unix comercial ou não tem um computador muito potente.
- No ano de 1983, Richard Stallman fundou a FSF – *Free Software Foundation* (Fundação de Software Livre), e criou o projeto **GNU GPL (GNU General Public License – Licença Pública Geral GNU)**. O desafio do GNU era enorme. Havia a necessidade de desenvolver o “**Kernel**” (**núcleo do sistema operacional que controla o hardware**), utilitários de programação, de administração do sistema, de rede, comandos padrão. Mas, no final da década de 80, o projeto estava fracassando e apenas os utilitários de programação e os comandos padrão estavam prontos, mas o Kernel não!
- **Linus Benedict Torvalds** era aluno da Universidade de Helsinque, na Finlândia e estava disposto a construir um Kernel clone do Unix que possuísse memória virtual, multitarefa e capacidade de multiusuários. Era um trabalho gigantesco e, na prática, impossível para apenas uma pessoa concluir-lo.
- Em 5 de outubro de 1991, Linus Torvalds lançou a primeira versão “oficial” do Linux: o Linux 0.02. A partir dessa data, muitos programadores no mundo inteiro têm colaborado e ajudado a fazer do Linux o sistema operacional que é atualmente.

## LICENÇA GPL

A licença GPL segue 4 liberdades. São elas:



O conteúdo deste livro eletrônico é licenciado para MARIO LUIS DE SOUZA - 41250799864, vedada, por quaisquer meios e a qualquer título, a sua reprodução, cópia, divulgação ou distribuição, sujeitando-se aos infratores à responsabilização civil e criminal.

*Figura. Licenças GPL e as 4 Liberdades*

Software responsável por **permitir a escolha de qual sistema operacional será carregado ao ligarmos um computador**. Os mais famosos **gerenciadores de boot do Linux** são:

- **LILO** (mais simples);
- **GRUB**.

Ambos permitem que se tenham o Linux e o Windows instalados em um mesmo computador. Também possibilitam que se escolha uma entre várias distribuições de Linux.

A figura seguinte ilustra a interface do GRUB. Observe as opções disponíveis que aparecem na janela, a saber: distribuição Red Hat Linux e o Windows XP.

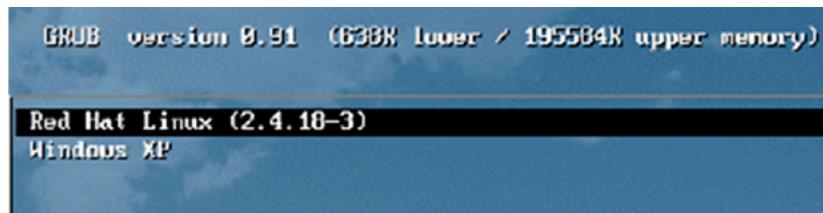


Figura. Interface do gerenciador de boot GRUB

A próxima figura mostra a interface do LILO. Observe as três opções disponíveis que aparecem na janela, a saber: Linux, LinuxOLD e Windows.



Figura. Interface do LILO

Após a seleção do sistema operacional desejado, o gerenciador de boot passa o controle do computador a esse sistema operacional.

## AMBIENTES GRÁFICOS

Há um número muito grande de **gerenciadores de janelas** (Windows Manager) que você pode instalar simultaneamente em uma máquina, possibilitando que cada usuário escolha aquele que mais lhe agrade.

Cada gerenciador difere do outro em muitos aspectos, como nível de customização da aparência e funcionalidades, configuração dos menus, meios gráficos para iniciar um software,

capacidade de utilizar múltiplos *desktops* e, principalmente, na quantidade de recursos que ele exige da máquina, entre outros.

Exemplos de **gerenciadores de janelas ou ambientes gráficos: GNOME, KDE** (*K Desktop Environment*), BlackBox, WindowMaker etc.

## DISTRIBUIÇÕES (DISTROS) GNU/LINUX

**Você já deve ter ouvido falar em Debian, RedHat, Slackware, SuSe, Ubuntu**, dentre outros.

Todos esses nomes são o que chamamos de **distribuições GNU/Linux**.

Várias empresas e organizações de voluntários decidiram juntar os programas do Linux em “pacotes” próprios aos quais elas fornecem suporte.

Uma **distribuição** é, portanto, uma versão do Linux empacotada por um determinado responsável (pessoa ou empresa), e que compreende um conjunto de programas formado pelo Kernel Linux e por mais alguns softwares distintos (como *shells*, aplicativos, jogos, utilitários etc.).

**Principais distribuições:**



### DICA!

**Não é necessário ficar decorando as diferenças entre cada uma delas, basicamente guardem o conceito de **distribuição** e **nomes** das principais.**

As distribuições podem:

- ser **produzidas em diferentes versões do Kernel**;
- incluir diferentes conjuntos de **aplicativos, utilitários, ferramentas e módulos de driver**;
- oferecer diferentes **programas de instalação e atualização para facilitar o gerenciamento do sistema**. Nesse caso, qualquer distribuição Linux irá possuir um **gerenciador de pacotes**, que cuidará de todos os detalhes necessários para instalar, desinstalar ou atualizar um programa que esteja no formato de um pacote RPM.

**Caso você não se identifique com nenhuma das distribuições, pode-se optar por criar a sua própria.** A partir desse ponto, foram surgindo diversas outras distribuições que de alguma forma se diferenciavam da filosofia do Slackware: como Debian ou RedHat, por exemplo.

Atualmente existem mais de 300 distribuições, algumas mais famosas que outras. Em sua maioria, mantidas por grandes comunidades de colaboradores, entretanto, há outras que são mantidas por empresas.

**As distribuições (ou distros) podem ser divididas em duas categorias básicas: livres e corporativas.**

- **Distribuições Corporativas:** mantidas por empresas que VENDEM o suporte ao seu sistema. Exemplos são: RedHat, SuSe e Mandriva. Neste ponto vale ressaltar o fato de que o produto vendido pelas empresas que comercializam sistemas GNU/Linux são, na verdade, os serviços relacionados ao sistema vendido, como suporte técnico, garantias e treinamentos, ou seja, o conhecimento do sistema.

O fato de o produto não ser mais o software, mas sim o serviço, é devido à Licença GPL que garante as já citadas quatro liberdades básicas. Com isso, por mais que uma empresa queira fazer o seu próprio sistema GNU/Linux, enquanto ela estiver utilizando softwares registrados com GPL, serão obrigadas a distribuir o código fonte gratuitamente.

- **Distribuições Livres:** mantidas por comunidades de colaboradores SEM fins lucrativos. Exemplos são: Debian, Ubuntu, Slackware, Gentoo, CentOS, etc. Dentro do conjunto de Distribuições Livres, podemos dividi-las novamente em duas outras categorias: Convencionais e Live.
  - **Distribuições convencionais:** distribuídas da forma tradicional, ou seja, uma ou mais mídias que são utilizadas para instalar o sistema no disco rígido.
  - **Distribuições live:** distribuídas em mídias com o intuito de rodarem a partir delas, SEM a necessidade de instalar no HD. Ficaram famosas, pois têm a intenção de fornecer um sistema GNU/Linux totalmente funcional, de forma fácil e sem a necessidade de instalar na máquina. O fator que favoreceu essa abordagem é que em uma distribuição Live praticamente todos os componentes já vêm configurados, funcionando e com interfaces agradáveis aos usuários finais. Exemplos desse tipo de distribuição são o Knoppix, do qual se originaram diversas outras como **Kurumin** ou **Kalango**, que são versões brasileiras do Knoppix, e o Ubuntu, bastante difundido atualmente.

## DIRETO DO CONCURSO

**001. (CESPE/2018/FUB/CONHECIMENTOS BÁSICOS/CARGOS DE NÍVEL MÉDIO)** Acerca do sistema operacional Linux, do gerenciador de arquivos Windows Explorer e do Microsoft Excel 2010, julgue o item subsequente. Há uma diversidade grande de distribuições do sistema Linux, cada uma delas com estrutura de diretórios e kernel diferentes.



O **kernel** é o núcleo do sistema operacional, e possui várias funções além do gerenciamento da CPU. Ele gerencia a memória; gerencia dispositivos de hardware; diz que sistema de arquivos

o sistema operacional usa, como deve usar e como deve se comportar. Para um sistema funcionar, só se precisa do kernel, todo o resto é complemento.

Uma **distribuição** é uma versão do Linux empacotada por um determinado responsável (pessoa ou empresa), e que comprehende um conjunto de programas formado pelo Kernel Linux e por mais alguns softwares distintos (como *shells*, aplicativos, jogos, utilitários, etc.). O Linux é o **kernel** (núcleo do sistema operacional) apenas e demanda usar uma das distribuições (distros) disponíveis, como Debian, Ubuntu, etc., para funcionar adequadamente.

Assim, **há uma diversidade grande de distribuições do sistema Linux, e um kernel utilizado por todas.**

**Errado.**

## PRINCIPAIS CARACTERÍSTICAS DO LINUX

O Linux é um **sistema operacional multitarefa, multiusuário, interoperável, portável, flexível, estável e open source**. Quanto adjetivo! Além de tudo, ele **segue o padrão POSIX/Unix**, o qual garante que temos no Linux todo o sistema de proteção do núcleo do kernel; com isso, fica “quase” impossível um programa travar em Linux.

### Multitarefa

O sistema é **capaz de executar simultaneamente vários programas**, garantindo a melhor distribuição de recursos entre esses programas. Um único programa **não** deve ser capaz de monopolizar os recursos da máquina, ao contrário do que ocorria, por exemplo, no Windows 95. Exemplo: No Linux você pode imprimir uma carta enquanto trabalha na planilha de vendas.

### Multiusuário

O sistema é **capaz de atender a vários usuários interativos simultaneamente**.

### Open Source

**Programa que tem seu código-fonte aberto.** Qualquer um pode baixar esse código-fonte, estudá-lo ou mesmo aperfeiçoá-lo.

### Interoperável

**O Linux executa bem com a maioria dos protocolos de rede e sistemas operacionais** incluindo Windows, Unix, Mac OS da Apple.

### Portável

A maioria do código do Linux é escrito em linguagem C, a vantagem disso é que ele pode ser prontamente portado para um novo hardware de computador. O Unix evoluiu com o surgimento da linguagem C.

## Flexível

O Linux **pode ser usado para várias finalidades**, como um *host* de rede, roteador, estação gráfica de trabalho, servidor de arquivos, servidor Web etc.

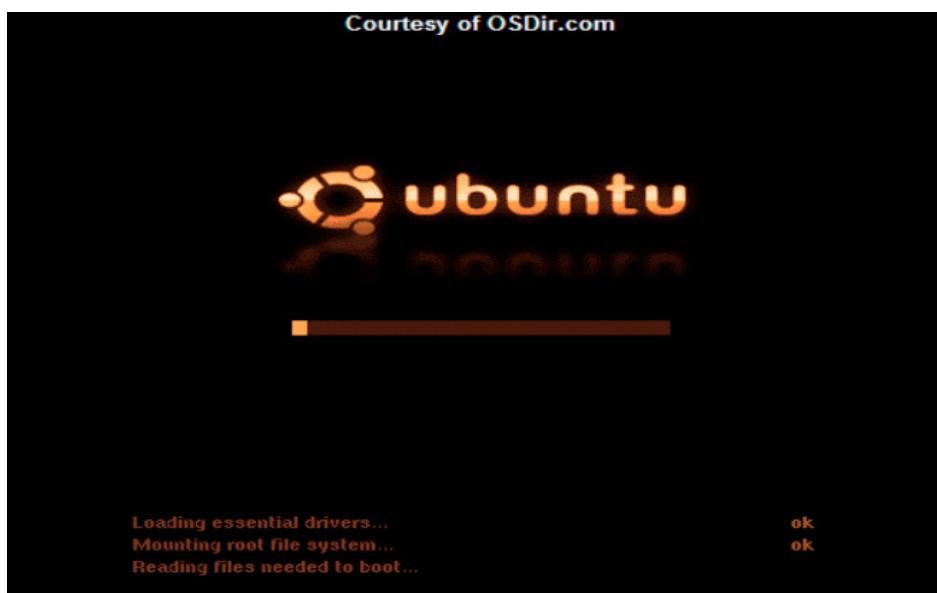
## Estável

O kernel do Linux atingiu um nível de maturidade muito bom. Não é raro encontrar relatos de servidores Linux que executaram durantes anos sem qualquer tempo de inatividade.

## KERNEL (NÚCLEO DO SISTEMA OPERACIONAL)

**Kernel** é a **parte central do sistema operacional** (ou seja, é o seu **núcleo**). Trata-se da parte do sistema operacional que “fala” diretamente com o hardware do computador.

Gerencia a memória; gerencia dispositivos de hardware; diz que sistema de arquivos o sistema operacional usa, como deve usar e como deve se comportar. Para um sistema funcionar, só se precisa do *kernel*, todo o resto é complemento. A tela seguinte destaca o carregamento do *kernel* durante o boot.



A próxima figura ilustra o Tux, símbolo do Kernel Linux.



Figura. Tux, o símbolo do Kernel Linux. Fonte: LinuxTage (<http://www.linuxtage.at/presse>)

## SHELL (INTERPRETADOR DE COMANDOS)

O **shell** é a interface entre o usuário e o kernel do sistema e por meio dele, podemos digitar os comandos.

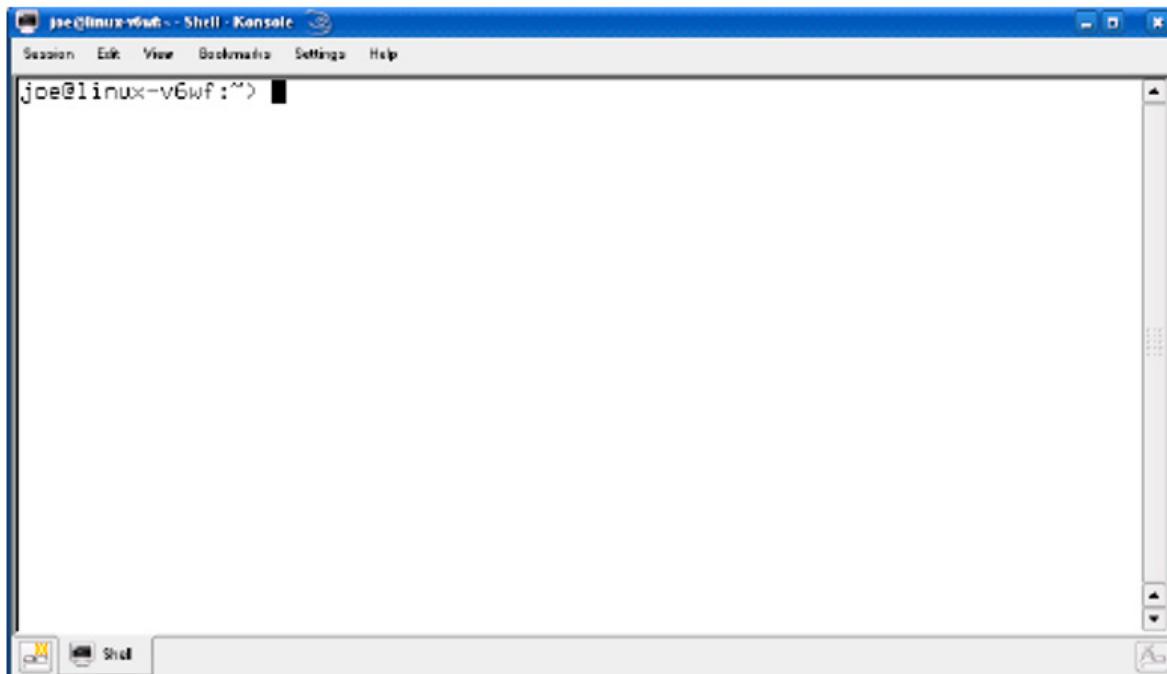
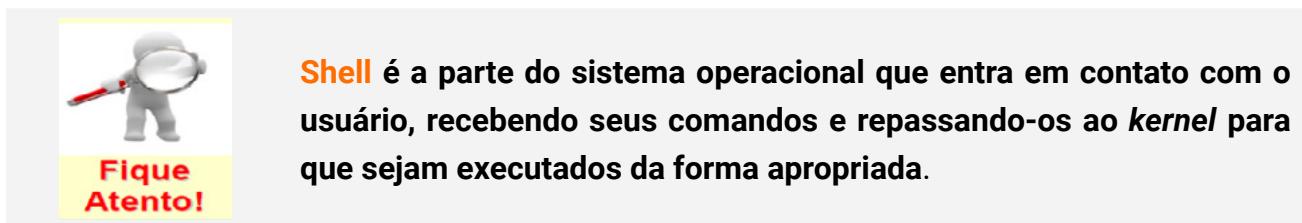
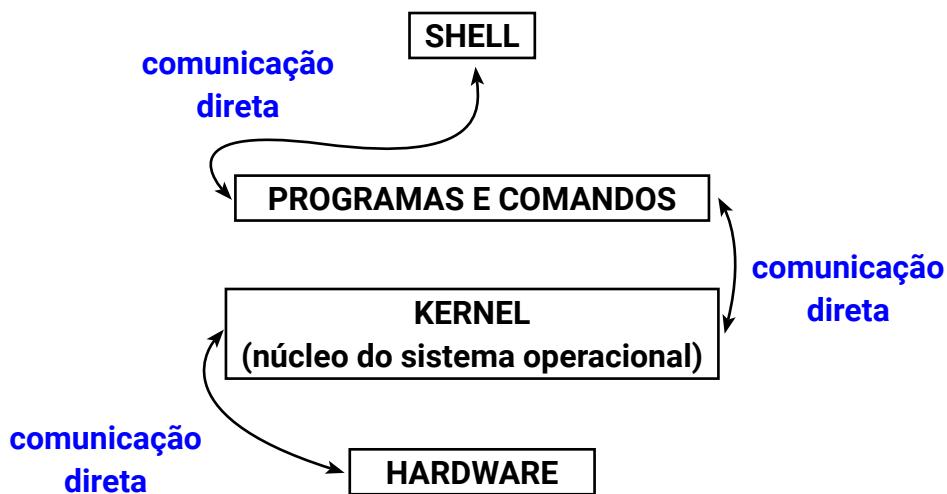


Figura. Ambiente para digitação de comandos no Linux

O *shell* padrão do GNU/Linux é o **bash**. Pode-se também ter outros *shells*, como, por exemplo, **csh**, **tcsh**, **ksh** e **zsh**.

Conforme pode ser visto na figura seguinte, **quanto mais baixo o nível, maior é a participação do kernel; quanto mais alto, maior a participação do shell**. Nível alto é o nível próximo ao usuário; nível baixo, próximo do hardware, da máquina.



Antes de vermos os principais comandos em si que podem ser executados no Linux, é necessário saber o que é **Linha de Comando**. Trata-se de um **modo de trabalho com caracteres, em que você digita o comando e o executa pressionando ENTER no teclado**.

Mas você também pode usar uma linha de comando em um ambiente gráfico. Se você usar o KDE, por exemplo, pode procurar o aplicativo KDE Terminal para abrir uma janela com linha de comando. Mas isso varia de acordo com a versão do seu Linux.

## USUÁRIOS E GRUPOS DO LINUX

Cabe destacar aqui, a diferença entre um **usuário comum** e o **superusuário** (conhecido como **root**), **MUITO cobrada em prova**:

|                                 |   |
|---------------------------------|---|
| <b>Super usuário<br/>(root)</b> | <p>É o administrador do sistema, e seu diretório (pasta) padrão é o /root, diferentemente dos demais usuários que ficam dentro de /home.</p> <p>O shell de um usuário root é diferente de um usuário comum. Antes do cursor, ele é identificado com "#" (jogo-da-velha).</p> <p><b>NOTA:</b> Podem ser criados outros usuários no sistema Linux com poderes de superusuários, que têm <u>privilégios administrativos completos</u> sobre o sistema.</p> |
| <b>Usuário comum</b>            | <p>Qualquer usuário do sistema que não seja root e não tenha poderes administrativos no sistema.</p> <p>Antes do cursor, o shell de um usuário comum é identificado com "\$" (cifrão). <b>ATENÇÃO!</b></p>  |

Vamos a um exemplo:

**[root@notebook:/documentos]#**

**Você sabe o que significa essa linha acima?**

Identificando a linha acima:

**root** = Usuário

**notebook** = nome da máquina

**:/documentos** = diretório atual

**#** = Indica que está logado como super usuário.

No Linux, os usuários, grupos de usuários e processos utilizam IDs para que o sistema operacional possa identificá-los durante as operações. Cada usuário, grupo ou processo possui um ID próprio.

Os **usuários** são identificados no sistema por um número chamado **UID (User IDentifier)**. O **root** é o usuário criado pelo sistema e o seu UID sempre será 0.

Em várias distribuições GNU/Linux, os **UID de 0 a 499 pertencem a usuários criados pelo sistema**. Assim, em vários GNU/Linux, os usuários terão como UID o número 500 em diante.

Os **grupos** criados no sistema são identificados por um número chamado **GID (Group IDentifier)**. Os grupos servem para permitir que vários usuários possam acessar um determinado arquivo ou diretório, sob certas condições e com determinadas permissões. Assim como o UID, o GID vai de 0 a 499 para grupos criados pelo sistema em várias distribuições.

O GNU/Linux trabalha com números de identificação para gerenciar usuários, grupos e processos. Esses números se chamam, respectivamente, **UID, GID e PID**.

O comando **#id** mostrará o UID e o GID do **usuário logado no sistema**.

## ARQUIVOS

O arquivo pode ser um texto, uma imagem, planilha etc. Os arquivos devem ser identificados por nomes para que sejam localizados por quem deseja utilizá-los.



**Fique Atento!**

Uma observação importante aqui é que o GNU/Linux é **case sensitive**, dessa forma **diferencia letras maiúsculas e minúsculas nos arquivos, comandos e diretórios (pastas)**.

O arquivo historia, por exemplo, é completamente diferente de Historia. Prefira, sempre que possível, usar letras minúsculas para identificar seus arquivos, pois quase todos os comandos do sistema estão em *minúsculas*.

## DIRETO DO CONCURSO

**002.** (CESPE/TRE-RJ/CBNS/2012) No Linux, em um mesmo diretório, não podem existir dois subdiretórios com o mesmo nome, contudo, em virtude de os nomes dos diretórios serem case sensitive, é possível criar dois subdiretórios de nomes /usr/TreRJ e /usr/trerj.



O Linux é **Case Sensitive**, pois diferencia letras maiúsculas e minúsculas nos arquivos, comandos e diretórios (pastas). Assim, pode-se criar os subdiretórios de nomes /usr/TreRJ e /usr/trerj, que são distintos para o sistema operacional. Observe que não podem existir dois arquivos com o mesmo nome em um diretório, ou um subdiretório com um mesmo nome de um arquivo em um mesmo diretório.

Certo.

## IDENTIFICAÇÃO DE DISCOS DO GNU/LINUX

Existem vários **tipos de unidades de disco**, utilizadas para armazenamento de dados. No GNU/Linux, elas são referenciadas de uma forma bastante particular, através de nomes de arquivos do **diretório /dev/**.

|                               |  |
|-------------------------------|--|
| Discos IDE                    | São os mais comuns, presentes na maioria dos computadores. Podem ser HDs, unidades de CD-ROM, CD-RW e DVDs.<br>No GNU/Linux, HDs são identificados como /dev/hdxy, em que x é a <u>letra que representa o número do disco</u> , e y é o número da partição.<br>CD-ROMs, CD-RWs e DVDs não têm partições, portanto são identificados apenas como /dev/hdx, sendo x a letra que representa o número do disco.    |
| Discos SCSI                   | <b>São discos de alto desempenho</b> , utilizados em servidores, por exemplo.<br>Exemplos: HDs, unidades de CD-RW e gravadoras de DVD.<br>No caso de HDs são referenciados como /dev/sdxy, sendo x a letra que indica o número do disco e y o número da partição.<br>CD-RWs e gravadoras de DVD não têm partições, sendo identificadas apenas como /dev/sdx, sendo x a letra que representa o número do disco. |
| Unidades de Disquete (Floppy) | Em desuso. São referenciadas como /dev/fdx, em que x é o número da unidade de disquete, começando por 0.   |

A seguir, destacamos algumas identificações de discos e partições em sistemas Linux:

|           |  |
|-----------|--|
| /dev/fd0  | Primeira unidade de disquetes.   |
| /dev/fd1  | Segunda unidade de disquetes.  |
| /dev/hda  | Primeiro disco rígido na primeira controladora IDE do micro ( <i>primary master</i> ). |
| /dev/hda1 | Primeira partição do primeiro disco rígido IDE.  |
| /dev/hdb  | Segundo disco rígido na primeira controladora IDE do micro ( <i>primary slave</i> )    |

O Linux nomeia os discos de acordo com um padrão que envolve o **tipo do disco** (Ex.: IDE, SCSI, Unidade de Disquete), **letra representando o número do disco** e **número da partição** (se existir).

Nota:

| Tipo do disco  | Qual disco rígido? | Qual partição? |
|--|--------------------|----------------|
| fd = Floppy Disk<br>hd = HD IDE<br>sd = HD SCSI (Serial ATA) | hda1, hdb1, hdc1   | hda1, hda2     |

### Exemplos:

/dev/fd0  
 /dev/fd1  
 /dev/hda1, /dev/hdb1, /dev/hda2, ...  
 /dev/sda1, /dev/sdb1, /dev/sda2, ...

Assim, diferentemente do Windows, no Linux os discos não recebem letras, mas siglas e números que os identificam, como exemplificado a seguir:

| Windows | Linux                  |
|---------|------------------------|
| A       | /dev/fd0               |
| B       | /dev/fd1               |
| C       | /dev/hda1 ou /dev/sda1 |

Os discos devem ser **montados** para que possam ser acessados. Atualmente os discos são montados automaticamente.

## ESTRUTURA DE DIRETÓRIOS DO GNU/LINUX

Um **diretório** nada mais é do que o local em que os arquivos são guardados no sistema.

A estrutura de diretórios também é conhecida como **árvore de diretórios** porque tem a forma de uma árvore. A seguir, destacamos os diretórios principais, mas a lista não se esgota por aqui!

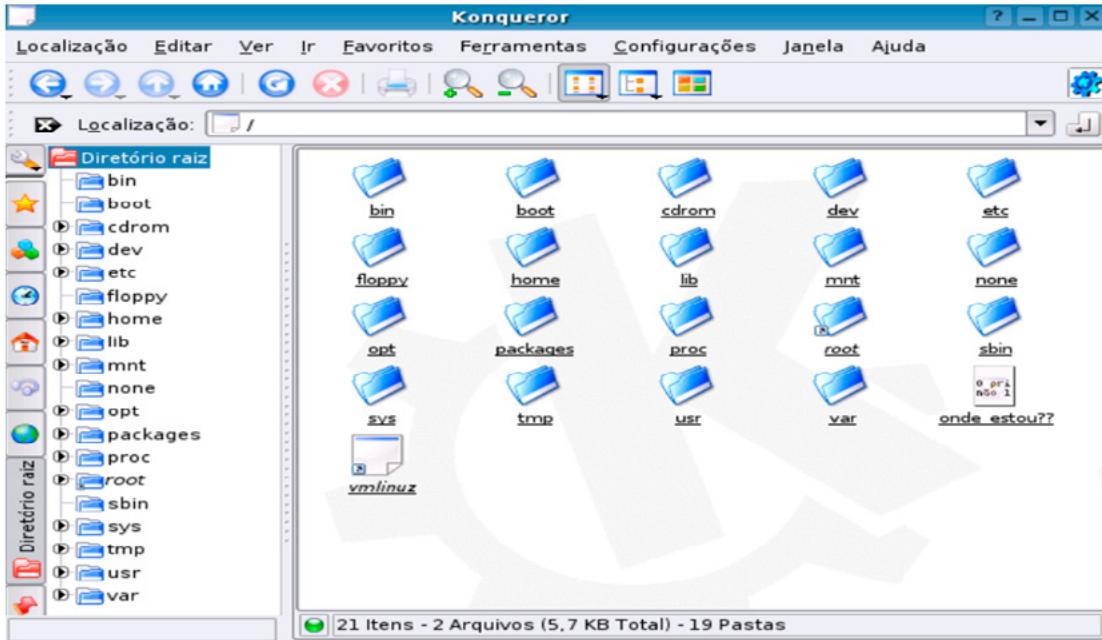
| Diretório | Descrição   |
|-----------|---|
| / (raiz)  | <p>Este é o <b>principal diretório</b> do GNU/Linux, e é representado por uma / (barra). No Linux, toda estrutura de diretórios <b>começa no barra (/), que significa início, portanto, é no diretório raiz</b> que ficam TODOS os demais diretórios do sistema!</p>                               |
| /bin      | <p>Contém arquivos executáveis de uso geral (<u>binários</u> ou não) que podem ser acessados pelos usuários. Guarda os <b>comandos essenciais para o funcionamento do sistema</b>. Esse é um diretório público, sendo assim, os comandos que estão nele podem ser utilizados por qualquer usuário do sistema. Entre os comandos, estão: <b>bash; ls; echo; cp; mkdir; rm, ...</b></p> |
| /boot     | <p>Contém o Kernel Linux e os arquivos que controlam a inicialização do sistema. Em outras palavras, guarda os <b>arquivos estáticos necessários à inicialização do sistema, e o gerenciador de boot</b>. O gerenciador de boot é um programa que carrega um sistema operacional e/ou permite escolher qual será iniciado.</p>  |

Figura. Diretório / (raiz) no Linux

| Diretório | Descrição  |
|-----------|--|
| /dev      | Contém <b>arquivos que servem de ligação com os dispositivos de hardware (devices) do computador</b> . O Linux faz a comunicação com os periféricos por meio de links especiais que ficam armazenados nesse diretório, facilitando assim o acesso aos mesmos.  |
| /etc      | Guarda os <b>arquivos de configuração do sistema</b> . Nesse diretório vamos encontrar vários arquivos de configuração, tais como: scripts de inicialização do sistema, tabela do sistema de arquivos, configuração padrão para logins dos usuários etc.   |
| /lib      | Contém os módulos do Kernel, drivers de dispositivo e as <b>bibliotecas (libraries)</b> utilizadas no momento da inicialização (boot) do sistema. As bibliotecas são funções que podem ser utilizadas por vários programas.  |
| /media    | <b>Ponto de montagem de mídias removíveis</b> , tais como: <b>CD-rom, DVD, disquete, pendrive, câmera digital, etc.</b> (Antes era usado o /mnt para isso, então ainda pode vir em prova o /mnt!)  |
| /mnt      | Utilizado para <b>montagem temporária de sistemas de arquivos</b> , tais como compartilhamentos de arquivos entre Windows e Linux, Linux e Linux etc.  |
| /temp     | Utilizado para <b>armazenamento de arquivos temporários</b> (guarda principalmente pequenas informações que precisam estar em algum lugar até que a operação seja completada, como é o caso de um download. Enquanto não for concluído, o arquivo fica registrado em /tmp, e, assim que é finalizado, é encaminhado para o local correto). |
| /usr      | Neste diretório encontra-se grande parte do Linux, nele estão programas, janelas gráficas, bibliotecas, fontes do Kernel, etc. <b>Guarda comandos que são de uso dos usuários em geral.</b>  |
| /var      | Contém <b>arquivos com conteúdo variável</b> , como logs, spool de impressoras (arquivos a serem impressos), caixas postais em servidores de e-mail etc.   |

A seguir, destacamos alguns diretórios opcionais, que podem estar disponíveis no sistema, mas não precisam obrigatoriamente possuir este nome.

| Diretório    | Descrição  |
|--------------|--|
| <b>/home</b> | Contém os diretórios pessoais dos usuários cadastrados no sistema. Por exemplo, o usuário 'patricia' terá todos os seus arquivos e suas configurações gravadas dentro do diretório /home/patricia que também pode ser representado como ~patricia                          |
| <b>/root</b> | O usuário <b>root</b> é o administrador do sistema, e pode alterar a configuração (dele), configurar interfaces de rede, manipular usuários e grupos, alterar a prioridade dos processos, entre outras. O <b>/root</b> é o diretório pessoal do superusuário <b>root</b> . |

**DICA:** Utilize uma conta de usuário normal em vez da conta root para operar seu sistema.

Uma razão para **EVITAR USAR PRIVILÉGIOS root** é por causa da **facilidade de se cometer danos irreparáveis como root**; além do que, **você pode ser enganado e rodar um programa malicioso**, como o Cavalo de Troia (que obtém poderes do super usuário) comprometendo a segurança do seu sistema sem a sua autorização!

O Linux usa uma estrutura diferente de organização em seu sistema de arquivos<sup>1</sup>. Por isso, em vez da sua pasta ser c:\arquivos\pasta\arquivo.txt, no Linux pode ser /home/pasta/arquivo.txt.

## DIRETO DO CONCURSO

**003. (CESPE/2011/EBC/CARGOS DE NÍVEL SUPERIOR/CONHECIMENTOS BÁSICOS/ EXCETO CARGO 4 ADVOCACIA)** Na árvore de diretórios do Linux, o /lib contém os programas necessários à inicialização do sistema, e o /home contém os arquivos e diretórios dos usuários.



O diretório **/lib** é reservado para armazenar os arquivos de biblioteca (no Windows são chamados de arquivos dll) e módulos do Kernel, compartilhados com frequência pelos programas instalados no sistema. O diretório que contém os programas e arquivos necessários para inicialização do Sistema é o **/boot**. O diretório **/home** contém os diretórios pessoais dos usuários cadastrados no sistema.

<sup>1</sup> **Sistema de arquivos:** é o local em que os arquivos e diretórios são guardados. Consiste em uma área formatada em um dispositivo como um HD. Exemplos de sistema de arquivo: ext2/ext3 (Linux), FAT (Windows), NTFS (Windows NT/2000/XP).

Por exemplo, o usuário 'patricia' terá todos os seus arquivos e suas configurações gravadas dentro do diretório /home/patricia que também pode ser representado como ~patricia.

**Errado.**

## Uso de CONSOLES no LINUX

**Console** é uma interface que permite a um operador realizar a comunicação com um sistema de computador, como um terminal do Linux por exemplo.

Por ser um sistema multitarefa, o Linux pode ser acessado por vários consoles ao mesmo tempo, assim como pode rodar vários programas ao mesmo tempo nesse sistema operacional. Para mudar o console do 1 a 6, utilize: **ALT+N** (Onde N representa o número do console desejado). Exemplo: ALT+1, ALT+2, ALT+3, ALT+4, ALT+5, ALT+6.

Agora você pode ir para o próximo console e o antecedente com:

- ALT+RIGHT (Vai para 1 console À FRENTE);
- ALT+LEFT (Vai para 1 console ATRÁS).

Se você quiser ir para outra sessão sem sair do console, utilize o comando **su**, que permite a troca de usuário no sistema.

A combinação de teclas CTRL+C no console de uma distribuição qualquer do Linux é usada para interromper um comando em execução (comando corrente).

## DIRETO DO CONCURSO

**004.** (CESPE/2015/TCU/TÉCNICO FEDERAL DE CONTROLE EXTERNO/CONHECIMENTOS BÁSICOS) A respeito dos sistemas operacionais Linux e Windows, do Microsoft PowerPoint 2013 e de redes de computadores, julgue o item a seguir.

No console de uma distribuição qualquer do Linux, como, por exemplo, o Ubuntu, é possível cancelar um comando em execução a partir do uso da combinação das teclas CTRL + C.



**Console** é uma interface que permite a um operador realizar a comunicação com um sistema de computador, como um terminal do Linux por exemplo.

Por ser um sistema multitarefa, o Linux pode ser acessado por vários consoles ao mesmo tempo, assim como pode rodar vários programas ao mesmo tempo nesse sistema operacional. A combinação de teclas CTRL+C no console de uma distribuição qualquer do Linux é usada para interromper um comando em execução (comando corrente).

**Certo.**

## COMANDOS DO LINUX

Como vimos, o **shell** é o responsável pela interação entre o usuário e o sistema operacional, interpretando os comandos. **É no shell que os comandos são executados.**

**Os comandos são pequenos programas, que podem ser executados para realizar tarefas específicas.**

De uma maneira geral o formato é: **comando -opções parâmetros**.

Em <http://bellard.org/jslinux/> temos um **emulador de Linux** feito em Javascript, que roda no seu navegador! Use-o para entender como é a linha de comandos do Linux e digite lá os principais comandos, como **pwd** (destaca em qual diretório o usuário se encontra), para ver o resultado como se fosse diretamente no Linux!

Vamos à descrição dos comandos mais cobrados em provas!

### cd

**Entra ou sai de diretório.**

Exemplo:

| Comando                | Resultado  |
|------------------------|--|
| #cd                    | Retorna ao diretório do usuário atual.                             |
| #cd Desktop            | Entra no diretório “Desktop”.                                      |
| #cd Docs/Textos/Cartas | Entra no diretório “Cartas”.                                       |
| #cd ..                 | Sai do diretório atual e vai para o diretório de nível logo acima. |
| #cd ../../             | Sobe dois níveis da árvore de diretórios.                          |
| #cd -                  | Altera entre o diretório atual e o anteriormente visitado.         |
| #cd ~                  | Vai para o diretório ‘home’ do usuário atual.                      |

## clear

**Limpa a tela** (equivale ao comando **cls** do antigo sistema operacional MS-DOS). O mesmo efeito pode ser obtido com **Ctrl + L**.

Exemplo:

| Comando | Resultado     |
|---------|---------------|
| #clear  | Limpa a tela. |

## date

**Mostra a data e a hora do sistema**, e também permite ajustá-las.

Exemplo:

| Comando | Resultado  |
|---------|--|
| #date   | Mostra data e hora atuais do sistema numa formatação padrão. |

## du

**Exibe o tamanho de arquivos e/ou diretórios.**

## ls

**Lista os arquivos e diretórios da pasta** (equivale ao comando DIR do MS-DOS).

Exemplo:

| Comando             | Resultado   |
|---------------------|---|
| #ls                 | Lista o conteúdo da pasta atual.  |
| #ls Desktop         | Lista o conteúdo da pasta Desktop.  |
| #ls Docs/Textos     | Lista o conteúdo da pasta Textos, localizada na pasta Docs.   |
| #ls -l              | Lista detalhada de arquivos e diretórios da pasta.  |
| #ls -a              | Lista todos os arquivos, inclusive os <b>ocultos</b> .  |
| #ls -a -l ou ls -al | <p><b>Na maioria dos comandos, podemos utilizar 2 ou mais argumentos seguidos</b>, como em “-a” e “-l”.</p> <p>Esse exemplo lista arquivos executáveis e ocultos (-a) em forma de lista detalhada (-l).</p> <p>Ao utilizar o argumento “-l”, veremos os <u>atributos dos arquivos</u>, detalhados a seguir.</p> |

As permissões são mostradas como uma série de **10 travessões e/ou letras** no começo de cada linha.

| 1 <sup>a</sup><br><b>Tipo de Arquivo</b> | Leitura                        |                |                | Escrita                 |                |                | Execução               |                |                 |
|--|--------------------------------|----------------|----------------|-------------------------|----------------|----------------|------------------------|----------------|-----------------|
|  | 2 <sup>a</sup>                 | 3 <sup>a</sup> | 4 <sup>a</sup> | 5 <sup>a</sup>          | 6 <sup>a</sup> | 7 <sup>a</sup> | 8 <sup>a</sup>         | 9 <sup>a</sup> | 10 <sup>a</sup> |
|  | <b>Proprietário do arquivo</b> |                |                | <b>Grupo do arquivo</b> |                |                | <b>Outros usuários</b> |                |                 |

A **1<sup>a</sup> posição** indica o **tipo de arquivo**, que pode ser:

|   |   |
|---|---|
| - | Hífen, indica que se trata de um arquivo          |
| d | Diretório   |
| l | Link simbólico ( <i>como se fosse um atalho</i> ) |
| c | Dispositivos de caracteres                        |
| b | Dispositivos de bloco                             |

As **9 posições restantes** representam as **chaves de permissões**.

Quando uma chave está **acionada** (permissão concedida), uma letra aparece. Quando uma chave está **inativa** (permissão negada), um travessão aparece no lugar da letra.

**As 3 primeiras chaves** (2<sup>a</sup>, 3<sup>a</sup> e 4<sup>a</sup> posições) aplicam-se ao **proprietário** do arquivo.

**As próximas 3 chaves** (5<sup>a</sup>, 6<sup>a</sup> e 7<sup>a</sup> posições) aplicam-se ao **grupo** ao qual pertence o arquivo.

**As 3 últimas chaves** (8<sup>a</sup>, 9<sup>a</sup> e 10<sup>a</sup> posições) aplicam-se aos **outros usuários**.

Cada grupo de 3 chaves contém uma chave de **leitura**, uma de **escrita** e uma de **execução**, nesta ordem. As **chaves de permissão** são:

|   |   |
|---|---|
| r | Permissão de leitura ( <b>read</b> ).   |
| w | Permissão de <b>escrita/gravação (write)</b> .  |
| x | Permissão de <b>execução (executable)</b> .<br>Obs.: Permissão de execução: quando aparece em diretórios, significa permissão de entrar nesse diretório, usando “cd”. |
| - | Nada.   |

**Exemplo 1)** Um arquivo com os atributos – **rwxr -- r --**, pode ser definido assim:

| 1 <sup>a</sup><br>Tipo de<br>Arquivo | Leitura Escrita Execução                             |   |  | Leitura Escrita Execução |                |                | Leitura Escrita Execução |                |                 |
|--------------------------------------|--|---|--|--------------------------|----------------|----------------|--------------------------|----------------|-----------------|
|                                      | 2 <sup>a</sup><br><b>Proprietário do<br/>arquivo</b> | 3 <sup>a</sup><br><b>Grupo do<br/>arquivo</b> | 4 <sup>a</sup><br><b>Outros<br/>usuários</b> | 5 <sup>a</sup>           | 6 <sup>a</sup> | 7 <sup>a</sup> | 8 <sup>a</sup>           | 9 <sup>a</sup> | 10 <sup>a</sup> |
| -                                    | r  | w   | x  | r                        | -              | -              | r                        | -              | -               |

Assim, podemos entender que:

- “–” trata-se de um arquivo;
- rwx** indica que o **proprietário** do arquivo pode lê-lo, alterá-lo e executá-lo;
- r--** indica que o **grupo** do arquivo pode apenas lê-lo;
- r--** indica que os **outros usuários** que não pertencem ao grupo do arquivo podem apenas lê-lo.

**Exemplo 2)** Um diretório com os atributos **d rwx -----**, pode ser definido assim:

|                 | Leitura                        | Escrita              | Execução             |                         | Leitura              | Escrita              | Execução               |   | Leitura              | Escrita              | Execução              |
|-----------------|--------------------------------|----------------------|----------------------|-------------------------|----------------------|----------------------|------------------------|---|----------------------|----------------------|-----------------------|
| 1 <sup>a</sup>  | <b>2<sup>a</sup></b>           | <b>3<sup>a</sup></b> | <b>4<sup>a</sup></b> |                         | <b>5<sup>a</sup></b> | <b>6<sup>a</sup></b> | <b>7<sup>a</sup></b>   |   | <b>8<sup>a</sup></b> | <b>9<sup>a</sup></b> | <b>10<sup>a</sup></b> |
| Tipo de Arquivo | <b>Proprietário do arquivo</b> |                      |                      | <b>Grupo do arquivo</b> |                      |                      | <b>Outros usuários</b> |   |                      |                      |                       |
| d               | r                              | w                    | x                    | -                       | -                    | -                    | -                      | - | -                    | -                    | -                     |

Podemos entender que:

- “d” trata-se de um **diretório** e não de um arquivo;
- rwx** indica que o **proprietário** do diretório pode lê-lo, alterá-lo e executá-lo;
- indica que o **grupo** do arquivo não tem permissões para lidar com este diretório;
- indica que os **outros usuários** que não pertencem ao grupo do arquivo também não têm permissões.

Agora que você entendeu os atributos, saiba que ao utilizar a linha de comando “**ls -l**”, obtemos, além dos atributos do arquivo, outras informações, listadas a seguir. Exemplo:

|                    |                       |                          |       |       |                  |   |                 |           |
|--------------------|-----------------------|--------------------------|-------|-------|------------------|---|-----------------|-----------|
| <b>- rwx -----</b> | 2                     | user1                    | user1 | 24    | Jun              | 30  | 18:00           | carta.txt |
| Tipo de Objeto     | Permissões de Arquivo | Número de Ligações/Links | Dono  | Grupo | Tamanho em Bytes | Data e hora da Criação ou da Última Modificação | Nome do Arquivo |           |

**Exemplo 3)** Foi digitado o seguinte comando:

**# ls -l arquivo**

Saída do comando:

**- rw- r-- r-- 1 root root 30 2004-11-12 16:26 leo.txt**

em que:

**-** = É a identificação de arquivo que pode ser:

**d** => indica que se trata de um diretório

**I** => indica que se trata de um link (como se fosse um atalho)

**-** => hífen, indica que se trata de um arquivo

**c** => indica dispositivo de caractere

**n** => indica dispositivo de bloco

**rw-** = Permissão do Dono

**r--** = Permissão do Grupo

**r--** = Permissão dos outros

**1** = Indicando ser um arquivo único (não possui links em outro lugar)

**root** = Dono do Arquivo

**root** = Grupo do Arquivo

**30** = Tamanho do Arquivo

**2004-11-12 12 16:26** Data do Arquivo

**leo.txt** = Nome do Arquivo

**cat**

**Exibe o texto contido em um arquivo. Concatena (junta) o conteúdo de arquivos. Cria arquivos baseados em caracteres de texto.**

Exemplo:

| Comando                 | Resultado   |
|-------------------------|---|
| #cat Carta              | Exibe o conteúdo do arquivo "Carta".  |
| #cat Carta  more        | Exibe o conteúdo do arquivo "Carta" linha por linha, pausadamente.          |
| #cat Carta.txt Memo.txt | Exibe na tela o conteúdo do arquivo "Carta.txt" e "Memo.txt", em sequência. |
| #cat -n Carta.txt       | Exibe o conteúdo do arquivo "Carta.txt", onde "-n" numera cada linha!       |

| Comando            | Resultado   |
|--------------------|---|
| #cat Carta.txt -n  | Exibe o conteúdo do arquivo “Carta.txt”, onde “-n” numera cada linha!                   |
| #cat > Relatório   | Cria o arquivo “Relatório” e aguarda a digitação do texto. [Ctrl]+[d] para finalizar.   |
| #cat > receita.txt | Cria o arquivo “receita.txt” e aguarda a digitação do texto. [Ctrl]+[d] para finalizar. |
| #cat >> Carta Memo | Acrescenta o conteúdo do arquivo “Memo” ao arquivo “Carta”.                             |
| #cat Carta >> Memo | Acrescenta o conteúdo do arquivo “Carta” ao arquivo “Memo”.                             |

## Chmod

**Altera as permissões de acesso a arquivos.**

Há duas maneiras para setar uma permissão com o comando chmod, **com letras e com números (octal)**.

===== **Com letras** =====

Aplica-se permissão para 03 “pessoas”:

|   |                        |
|---|------------------------|
| u | <b>Usuário (user).</b> |
| g | <b>Grupo (group).</b>  |
| o | <b>Outros (other).</b> |

Aplicam-se 03 tipos de permissões:

|   |  |
|---|--|
| r | Permissão de leitura ( <b>read</b> ).          |
| w | Permissão de <b>escrita/gravação (write)</b> . |
| x | Permissão de <b>execução (executable)</b> .    |

Falando dos sinais, temos:

|   |                           |
|---|---------------------------|
| = | Aplique exatamente assim. |
| + | Adicionar mais essa.      |
| - | Tirar essa.               |

Exemplo:

| Comando                               | Resultado   |
|---------------------------------------|---|
| <b># chmod u=rwx,g=rw,o=r arquivo</b> | Nesse caso, o dono ( <b>u</b> ) que é o usuário dono do arquivo terá permissão total: leitura (r), gravação (w), execução (x). O grupo ( <b>g</b> ), grupo de usuários, terá apenas a permissão de leitura(r) e gravação(w). E todo o resto dos usuários ( <b>o</b> ) apenas leitura (r). |

### ===== **No modo Octal**=====

Nesse modo as permissões serão aplicadas **com uso de números**, sendo que a permissão de leitura (r) equivale ao número 4, a permissão de escrita/gravação (w) corresponde ao número 2 e a permissão de execução (x) corresponde a 1.

|   |          |  |
|---|----------|--|
| 4 | <b>r</b> | Permissão de leitura ( <b>read</b> ).          |
| 2 | <b>w</b> | Permissão de <b>escrita/gravação (write)</b> . |
| 1 | <b>x</b> | Permissão de <b>execução (executable)</b> .    |

Assim, temos:

| Número        | Significado                            |
|---------------|--|
| 0             | Nenhuma permissão.                     |
| 1             | Permissão para executar.               |
| 2             | Permissão para gravar.                 |
| 3             | Permissão para gravar e executar.      |
| 4             | Permissão para ler.                    |
| 5 = 4 + 1     | Permissão para ler e executar.         |
| 6 = 4 + 2     | Permissão para ler e gravar.           |
| 7 = 4 + 2 + 1 | Permissão para ler, gravar e executar. |

Exemplos:

| Comando   | Resultado   |
|---|---|
| #chmod 764 arquivo<br>#chmod u=rwx,g=rw,o=r arquivo | Altera a permissão do arquivo para: proprietário (7xx), com permissão para ler, gravar e executar; grupo (x6x) com permissão para ler e gravar; qualquer outro usuário (xx4), com permissão para ler. |
| #chmod 664 teste.txt                                | Altera a permissão do arquivo teste.txt para: proprietário (6xx), com permissão para ler e gravar; grupo (x6x) com permissão para ler e gravar; qualquer outro usuário (xx4), com permissão para ler. |

**Detalhe importante sobre permissões.** Quando é abordado permissão total (**rwx**), temos o seguinte:

### Falando de diretórios

r – Posso listar o conteúdo do mesmo.

w – Posso criar arquivos dentro do mesmo.

x – Posso entrar nele para criar os arquivos ou listar.

É importante saber que para que o usuário tenha acesso a um diretório o mesmo deverá ter permissão de execução.

### Falando de Arquivos

- r – Posso ler o conteúdo desse arquivo.
- w – Posso alterar o conteúdo desse arquivo.
- x – Posso executar esse arquivo.

### Mas atenção!

O sistema por padrão **não** adota que todo arquivo criado será um shell script (ou seja, um executável), então a opção **x** em arquivo não tem que ser setada por padrão, senão terei vários arquivos executáveis que na verdade são apenas arquivos de texto normal.

**cp**

### Copia arquivos ou diretórios.

Exemplos:

| Comandos                      | Descrição   |
|-------------------------------|---|
| #cp Teste2.txt /root/Arquivos | Copia “Teste2.txt” do diretório atual para o diretório “Arquivos”.  |
| #cp T1.txt T2.txt             | Copia o arquivo “T1.txt” chamando a cópia de “T2.txt”.  |
| #cp Arq Arq2                  | Copia “Arq”, chamando de “Arq2”. Se “Arq2” já existir, será substituído.  |
| #cp -b Arq Arq2               | Copia “Arq”, chamando de “Arq2”. Se “Arq2” já existir, será criado um backup: “Arq2~”.  |
| #cp -b Arq Arq2 -v            | Copia “Arq”, chamando de “Arq2”. Se “Arq2” existir, será criado um backup: Arq2~”. O argumento <b>-v</b> indica “exibição em modo “verbose” (Arq → Arq2). |

### Find

### Busca arquivos e diretórios.

Sintaxe:

**\$find [diretório] [opções/expressão]**

O conteúdo deste livro eletrônico é licenciado para MARIO LUIS DE SOUZA - 41250799864, vedada, por quaisquer meios e a qualquer título, a sua reprodução, cópia, divulgação ou distribuição, sujeitando-se aos infratores à responsabilização civil e criminal.

em que

**-name [expressão]**: procura pelo nome [expressão] nos nomes de arquivos e diretórios processados.

Exemplo:

| Comando             | Resultado  |
|---------------------|--|
| # find / -name grep | Procura no diretório raiz e nos subdiretórios um arquivo/diretório chamado grep. |

## grep

**Permite encontrar informações dentro dos arquivos.**

Em outras palavras, cabe destacar que esse comando **procura strings (palavras ou expressões) em arquivos!**

Exemplo:

| Comando                 | Resultado   |
|-------------------------|---|
| # grep bash /etc/passwd | Permite localizar o texto bash no arquivo /etc/passwd.<br>Saída do comando:<br><i>root:x:0:0:root:/root:/bin/bash<br/>saito:x:1000:1000:saito,,,,:/home/saito:/bin/bash<br/>postgres:x:108:113:PostgreSQL administrator,,,,:/var/lib/postgresql:/bin/bash<br/>jboss:x:1001:1001:JBoss Administrator,,,,:/home/jboss:/bin/bash</i> |

## history

**Mostra os últimos comandos executados pelo usuário.**

## ifconfig

**Mostra as configurações de todos os adaptadores de rede (placas de rede) ativos na máquina.**

Com o argumento **-a**, mostrará o status de todas as interfaces, mesmo as não ativas.

**kill**

**Encerra um ou mais processos em andamento.**

Sintaxe:

**\$kill [sinal] [pid do processo]**

*Encerra os processos sendo que sinal pode ser:*

- 1 – Reinicia o processo;
- 9 – Destroi o processo;
- 15 – Envia uma solicitação de encerramento ao processo.

**killall**

**Permite finalizar processos através do nome.**

Exemplo:

| Comando          | Resultado                    |
|------------------|------------------------------|
| #killall firefox | Finaliza o processo firefox. |

**ln**

**Usado para criar uma ligação (atalho ou *link* simbólico como é mais conhecido) entre arquivos do sistema de arquivos.**

**lpq**

**Mostra o status da fila de impressão.**

Exemplos:

| Comando   | Resultado  |
|-----------|--|
| #lpq      | Exibe os arquivos da fila de impressão da impressora ativa.    |
| #lpq -all | Exibe os arquivos da fila de impressão de qualquer impressora. |

## **lpr**

**Imprime o arquivo indicado.**

Exemplo:

| Comando    | Resultado                  |
|------------|----------------------------|
| #lpr Carta | Imprime o arquivo “Carta”. |

## **login**

**Inicia a sessão pedindo nome de usuário e senha (userid e password).**

Exemplo:

| Comando | Resultado  |
|---------|--|
| #login  | Inicia a sessão pedindo nome de usuário e senha. |

## **logout**

**Finaliza a sessão do usuário atual e pede login de novo usuário e senha (password).**

Exemplo:

| Comando | Resultado                                      |
|---------|--|
| #logout | Finaliza a sessão e pede “login” e “password”. |

## **mkdir**

**Cria diretórios.**

Exemplos:

| Comando                     | Resultado   |
|-----------------------------|---|
| #mkdir Documentos           | Cria o diretório “Documentos”.  |
| #mkdir Fotos1 Fotos2 Fotos3 | Cria os diretórios “Fotos1, Fotos2 e Fotos3” dentro do diretório atual. |
| #mkdir Fotos1/Paisagens     | Cria o diretório “Paisagens” dentro do diretório “Fotos1”.              |

O conteúdo deste livro eletrônico é licenciado para MARIO LUIS DE SOUZA - 41250799864, vedada, por quaisquer meios e a qualquer título, a sua reprodução, cópia, divulgação ou distribuição, sujeitando-se aos infratores à responsabilização civil e criminal.

## **mv [opções] [origem] [destino]**

*origem* é o Arquivo/diretório de origem e *destino* é o local onde será movido ou novo nome do arquivo/diretório.

**Move ou renomeia arquivos e diretórios.** O processo é semelhante ao do comando cp mas o arquivo de origem é apagado após o término da cópia.

*Opções:*

|                   |   |
|-------------------|---|
| -f, --force       | Substitui o arquivo de destino sem perguntar. |
| -i, --interactive | Pergunta antes de substituir. É o padrão.     |
| -v, --verbose     | Mostra os arquivos que estão sendo movidos.   |

*Exemplos:*

| Comando                       | Resultado  |
|-------------------------------|--|
| #mv Carta.txt Texto.txt       | Renomeia o arquivo “Carta.txt”, chamando-o de “Texto.txt”.                   |
| #mv Teste2.txt /root/Arquivos | Move “Teste2.txt” do diretório atual para o diretório “Arquivos”.            |
| #mv Teste2.txt Teste.txt -v   | Renomeia o arquivo Teste2.txt para “Teste.txt”, em modo “verbose6”.          |
| #mv -v Test1 Test2            | Renomeia Test1 para “Test2”, em modo “verbose”: ‘Teste2.txt’ → ‘Teste.txt’). |

## **passwd**

**Permite criar ou modificar a senha de um determinado usuário.**

Atenção: somente o usuário **root** pode alterar as senhas.

*Exemplo:*

| Comando       | Resultado  |
|---------------|--|
| #passwd user1 | Permite <i>criar ou modificar a senha do usuário user1</i> . |

## DIRETO DO CONCURSO

**005.** (CESPE/CGM-PB/2018) Julgue o item, relativo aos sistemas operacionais Linux e Windows e ao editor de texto Microsoft Word 2013.

No Linux, a senha de usuário pode ser alterada via terminal por meio do comando `passwd`, mas o usuário, com exceção do root, não consegue alterar sua própria senha.



O comando `passwd` é utilizado para criar ou modificar a senha de um determinado usuário ou grupo.

Um usuário somente pode alterar a senha de sua conta, após informar a sua senha atual. Já o superusuário (root) pode alterar a senha de qualquer conta de usuário, inclusive a sua própria, sem necessidade de informar a senha atual. Os donos de grupos também podem alterar a senha do grupo com este comando.

Guarde isso: Você deve ser o dono da conta para poder modificar a senha. O usuário root pode modificar/apagar a senha de qualquer usuário.

Exemplo:

| Comando                    | Resultado  |
|----------------------------|--|
| <code>#passwd user1</code> | Permite <i>criar ou modificar a senha do usuário user1</i> . |

**Errado.**

**ps**

**Mostra os processos em execução.**

Exemplo:

| Comando                              | Resultado  |
|--------------------------------------|--|
| <code>#ps</code>                     | Mostra todos os processos do usuário.  |
| <code>#ps -aux</code>                | “a” mostra todos os processos, “u” de todos os usuários. “x” inclusive não gerados pelos terminais.  |
| <code>#ps -aux   grep firefox</code> | “a” mostra todos os processos, “u” de todos os usuários. “x” inclusive não gerados pelos terminais e usa o grep para filtrar pelos processos com nome firefox. |

**pwd**

**Destaca em qual diretório o usuário se encontra.** Mostra o “path” (caminho) do diretório atual.

Exemplo:

| Comando | Resultado                                   |
|---------|---|
| #pwd    | Mostra o diretório em que você se encontra. |

 **DIRETO DO CONCURSO**

**006.** (CESPE/2012/PEFOCE) O comando pwd do Linux possibilita ao usuário efetuar a troca de senha (password).



O comando **pwd** mostra o caminho do diretório corrente. O comando **passwd** é utilizado para a troca de senha. Observe que um usuário somente pode alterar a senha de sua conta, mas o superusuário (**root**) pode alterar a senha de qualquer conta de usuário, inclusive a data de validade da conta etc. Os donos de grupos também podem alterar a senha do grupo com este comando.

**Errado.**

---

**rm**

Remove diretórios, vazios ou não, e arquivos.

Utilize a chave “–r” para remover diretórios, recursivamente, pedindo confirmação para apagar os arquivos e os subdiretórios encontrados. Não pedirá confirmação se for utilizado em conjunto com a chave –f.

Exemplo:

| Comando               | Resultado   |
|-----------------------|---|
| #rm Carta.txt         | Exclui o arquivo “Carta.txt”, que se encontra no diretório atual.       |
| #rm –r MeusDocumentos | Exclui o diretório “MeusDocumentos”: o argumento “–r” indica diretório. |

| Comando                  | Resultado  |
|--------------------------|--|
| #rm -r -f MeusDocumentos | Exclui o diretório “MeusDocumentos” sem pedir confirmação “-f”.  |
| #rm teste                | Remove o arquivo “teste”, que se encontra no diretório atual.  |
| #rm -rf mala             | Remove o diretório sem pedir confirmação. A ação será recursiva.   |
| #rm -rf /                | Apaga todos os arquivos e diretórios, a partir da raiz do sistema, recursivamente e sem pedir confirmação. <b>Isso irá apagar todo o sistema operacional e todos os dados!</b> |

## rmdir

**Remove diretórios “vazios”.** É uma alternativa ao comando rm quando o diretório em questão estiver vazio.

Exemplo:

| Comando          | Resultado  |
|------------------|--|
| #rmdir Imagens   | Exclui o diretório “Imagens”, que se encontra dentro do diretório atual, desde que esteja “vazio”. |
| #rmdir /usr/mala | Remove o diretório “mala”, que se encontra dentro de /usr.   |

## shutdown

**Desliga ou reinicia o computador.**

Exemplo:

| Comando           | Resultado  |
|-------------------|--|
| # shutdown -r now | Reinicia o computador.   |
| #shutdown -h now  | Desliga o computador.  |
| #shutdown -r +10  | Reinicia o micro em 10 min. /Basta substituir “now” pelo tempo requerido: +1, +2...) |
| #shutdown -h +30  | Desliga um computador com o sistema operacional Linux após 30 minutos.               |

**startx**

**Inicia a interface gráfica do Linux (Xwindow).**

Exemplo:

```
#startx
```

**su**

**Troca de usuário.**

Exemplo:

| Comando       | Resultado   |
|---------------|---|
| \$su          | Vai para o usuário root, que é o 'super-usuário'. |
| \$su Patricia | Pede senha para alternar para a usuária Patricia. |

**sudo**

Permite a um usuário em particular executar vários comandos como superusuário sem que possua sua senha.

 **DIRETO DO CONCURSO**

**007.** (CESPE/TRE-RJ/CBNS/2012) No Linux, a sintaxe sudo adduser fulano criará o usuário fulano no grupo /etc/skell, bem como criará o diretório /home/fulano.



O comando **sudo** permite a um usuário em particular executar vários comandos como superusuário sem que possua sua senha, ou seja, sem a senha do root.

O comando **adduser** adiciona um usuário ou grupo no sistema. Por padrão, quando um novo usuário é adicionado, é criado um grupo com o mesmo nome do usuário. Será criado um diretório home com o nome do usuário (a não ser que o novo usuário criado seja um usuário do sistema) e este receberá uma identificação. Já o comando para criação de diretórios no Linux é o **mkdir**. Portanto, a ação desejada na questão não será possível de ser realizada com o comando aqui ilustrado.

**Errado.**

## top

Exibe, em tempo real, os processos que estão sendo executados no sistema. Também permite a manipulação de processos por meio de comandos interativos.

O comando **top** permite aferir o uso da CPU e da memória de uma estação de trabalho instalada com Linux.

Veja uma tela de saída desse comando:

```
Cpu(s): 21.5%us, 5.6%sy, 0.0%ni, 63.7%id, 9.2%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 3058052k total, 2083212k used, 974840k free, 167964k buffers
Swap: 3481788k total, 0k used, 3481788k free, 1090496k cached

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
2398 ricardo 20 0 428m 76m 29m S 15 2.6 2:48.27 compiz
5489 ricardo 20 0 169m 14m 10m S 9 0.5 0:00.28 gnome-screensho
3187 ricardo 20 0 1042m 393m 42m S 8 13.2 27:00.38 firefox
1238 root 20 0 92920 26m 8724 S 7 0.9 4:22.79 Xorg
```

## touch

**Atualiza a última data e hora de acesso ao arquivo. Caso o arquivo não exista, será criado um arquivo vazio como padrão.**

Exemplo:

| Comando                | Resultado  |
|------------------------|--|
| \$ touch Carta.txt     | Se “Carta.txt” existe, a data/hora do último acesso ao arquivo será a data/hora atual. |
| \$ touch Relatorio.txt | Se o arquivo “Relatório.txt” não existe, é criado um arquivo “Relatório.txt” vazio.    |
| \$ touch -m Foto.bmp   | Se “Foto.bmp” existe, será atualizada apenas a “hora” de acesso, para a atual.         |

## uptime

**Mostra há quanto tempo o sistema está ligado.**

## wc

Grande parte dos arquivos de configuração e de dados usa uma linha por registro. A **contagem destas linhas** pode nos fornecer informações muito interessantes com uso do comando wc.

Por exemplo, a saída abaixo:

```
# wc /etc/passwd
```

Pode indicar por exemplo que o arquivo contém 32 linhas, 49 blocos (palavras) e 1528 caracteres.

Caso seja necessário apenas o número de linhas, o comando **wc** pode ser usado com o parâmetro **-l**, como abaixo:

```
# wc -l /etc/passwd
```

Outros parâmetros possíveis são **-w** para blocos (palavras) e **-c** para caracteres.

## WILDCARDS (CARACTERES CURINGA)

Alguns caracteres podem ser utilizados como curingas em alguns comandos, ou seja, podem substituir outros caracteres ou sequências de caracteres. Esses curingas são o asterisco (\*), a interrogação (?) e os colchetes ([]).

No GNU/Linux são usados **três tipos de curingas**, especificados a seguir:

|          |   |
|----------|---|
| *        | O asterisco representa “de zero a infinitos caracteres”.  |
| ?        | A interrogação pode ser interpretada como <b>um caractere qualquer</b> . No entanto, ela substitui <b>apenas</b> um caractere. Em outras palavras, deverá existir um caractere para cada interrogação.  |
| [padrão] | Os colchetes servem para definir possibilidades de recursos. Substituem um conjunto de caracteres especificados dentro dos colchetes. Os caracteres circunflexo (^) e exclamação (!) podem ser usados como negação ( <b>apenas</b> para <b>um</b> caractere). O caractere <b>hífen</b> pode ser utilizado para definir <b>intervalos</b> , seguindo a tabela ASCII. |

Exemplos de sintaxe com colchetes:

|           |  |
|-----------|--|
| [A-Z]     | Qualquer caractere de A a Z, com caixa alta (maiúsculos).                              |
| [ABCde15] | Qualquer um dos caracteres citados, considerando a caixa dos mesmos no caso de letras. |
| [0-9]     | Qualquer algarismo, de 0 a 9.  |
| [!t]      | Qualquer caractere exceto o t.   |

|            |   |
|------------|---|
| [Tt]este   | Qualquer arquivo ou diretório que inicie com o caractere T ou com t e que termine com este. |
| [a-z][0-9] | Usado para trabalhar com caracteres de a até z seguidos de um caractere de 0 até 9.         |
| [a-z,1,0]  | Faz referência do intervalo de caracteres de a até z ou 1 ou 0 naquela posição.             |

Vamos aos exemplos. Supondo que existam 5 arquivos no diretório /home/patricia. Podemos listá-los com o uso do comando **ls**:

```
# ls
```

Saída do comando:

```
arq1.txt arq2.txt arq3.txt arq4.new arq5.new
```

Vamos listar agora todos os arquivos do diretório /home/patricia. Podemos usar o curinga "\*" para visualizar todos os arquivos do diretório:

```
# cd /home/patricia
# ls *
```

Saída do comando:

```
arq1.txt arq2.txt arq3.txt arq4.new arq5.new
```

Para listarmos todos os arquivos do diretório /home/patricia que tenham "new" no nome:

```
# ls *new*
```

Saída do comando:

```
arq4.new arq5.new
```

## CONEXÃO DE COMANDOS

**Conectar comandos** é fazer com que o resultado gerado por um comando seja processado por outro comando, mediante a aplicação do caractere **pipe** (**|**).

Exemplo: **# cat /etc/hosts | grep localhost**

O primeiro comando iria ler o conteúdo do arquivo /etc/hosts. O resultado dessa operação, ao invés de ser exibido na tela, seria enviado para o comando grep localhost, que iria selecionar somente as linhas que contivessem a palavra localhost. Esse último resultado seria exibido na tela.

## REDIRECIONAMENTO DE E/S (ENTRADA/SAÍDA) DE DADOS

O Linux permite o **redirecionamento de entrada e saída de dados**. Nesse caso, quase todos os comandos Linux têm uma entrada e produzem uma saída.

A **entrada** de um comando são os dados que o comando vai processar, e essa entrada pode vir de um arquivo especificado pelo usuário, de um arquivo do sistema, do terminal ou da saída de outro comando. A **saída** de um comando são os dados de entrada processados, e pode ser pressa na tela de um terminal, enviada a um arquivo, ou servir de entrada a um outro comando.

É possível tratar a entrada padrão e a saída padrão usando apenas alguns **caracteres especiais**.

Os principais estão listados a seguir:

|    |   |
|----|---|
| >> | Redireciona a saída padrão para um arquivo sem apagar o conteúdo do arquivo.                  |
| >  | Redireciona a saída padrão <u>para um arquivo</u> , porém o arquivo é apagado caso já exista. |
| <  | Redireciona a entrada padrão usando um arquivo.   |
|    | Conecta a saída padrão na entrada padrão de outro arquivo.                                    |

Todos esses redirecionamentos são muito usados. Apesar de parecer complicado não é tanto assim.

Usando exemplos práticos fica simples:

```
$ cat > arquivo.txt
```

O comando acima envia a saída do comando **cat** para o arquivo **arquivo.txt**.

```
echo TESTE > arquivo.txt
```

Escreve TESTE em um arquivo chamado arquivo.txt porém o arquivo é apagado caso já exista.

```
$ ls /home/patricia
```

O comando acima irá mostrar na tela todos os arquivos do diretório /home/patricia.

```
$ ls /home/patricia > /tmp/ls.txt
```

Neste momento, enviamos a saída do comando **ls** para o arquivo **/tmp/ls.txt**. Fizemos um redirecionamento de saída.

## CONCATENAÇÃO DE COMANDOS

É possível **concatenar comandos** (o mesmo que **sequenciar comandos**) usando o caracter **ponto e vírgula (;**).

A concatenação de comandos faz com que um comando seja executado após o outro.

Seria algo do tipo:

**comando 1 ; comando 2 ; comando 3 ; comando 4**

Assim, podem-se executar dois comandos em uma mesma linha, separando-os com “ponto-e-vírgula”. Exemplo: ls; man ls.

### DIRETO DO CONCURSO

**008.** (CESPE/2016/TRE-PI/NÍVEL SUPERIOR/CONHECIMENTOS BÁSICOS/TODOS OS CARGOS) Assinale a opção que apresenta o comando que um usuário deve utilizar, no ambiente Linux, para visualizar, em um arquivo de texto (nome-arquivo), apenas as linhas que contenham determinada palavra (nome-palavra).

- a) pwd nome-arquivo | locate nome-palavra
- b) find nome-palavra | ls -la nome-arquivo
- c) cat nome-arquivo | grep nome-palavra
- d) lspci nome-arquivo | find nome-palavra
- e) cd nome-arquivo | search nome-palavra



**Conectar comandos** faz com que o resultado gerado por um comando seja processado por outro comando, mediante a aplicação do caractere pipe (`|`). O caractere pipe (`|`) é utilizado para enviar a saída de um comando para a entrada do próximo comando. Os dados enviados são processados pelo próximo comando que mostrará o resultado do processamento.

Veja os 2 comandos utilizados na sequência destacada a seguir:

**cat nome-arquivo | grep nome-palavra**

- **cat**: exibe o conteúdo de um arquivo, sem pausa;
- **grep**: filtra o conteúdo de um arquivo. Exibe somente as linhas que contenham a sequência informada.

Assim, no contexto da questão, o primeiro comando iria ler o conteúdo do arquivo **nome-arquivo**. O resultado dessa operação, ao invés de ser exibido na tela, seria enviado para o comando **grep nome-palavra**, que iria selecionar somente as linhas que contivessem a palavra **nome-palavra**. Esse último resultado seria exibido na tela. Portanto, a letra C é a resposta da questão.

**Letra c.**

## INSTALAÇÃO DE APLICATIVOS

Existem várias **formas de instalar aplicativos no Linux**, dependendo da distribuição.

Em linha de comando, há duas ferramentas principais que instalam **pacotes** gerenciando **dependências**:

| Comandos | Usos   |
|----------|--|
| apt-get  | Em distribuições baseadas em Debian.   |
| yum      | O <b>yum</b> é o gerenciador de pacotes usado por padrão no <b>Fedora</b> , assumindo o papel que no Debian e no Ubuntu é desempenhado pelo <b>apt-get</b> |

## INSTALAR E CONFIGURAR O DNS

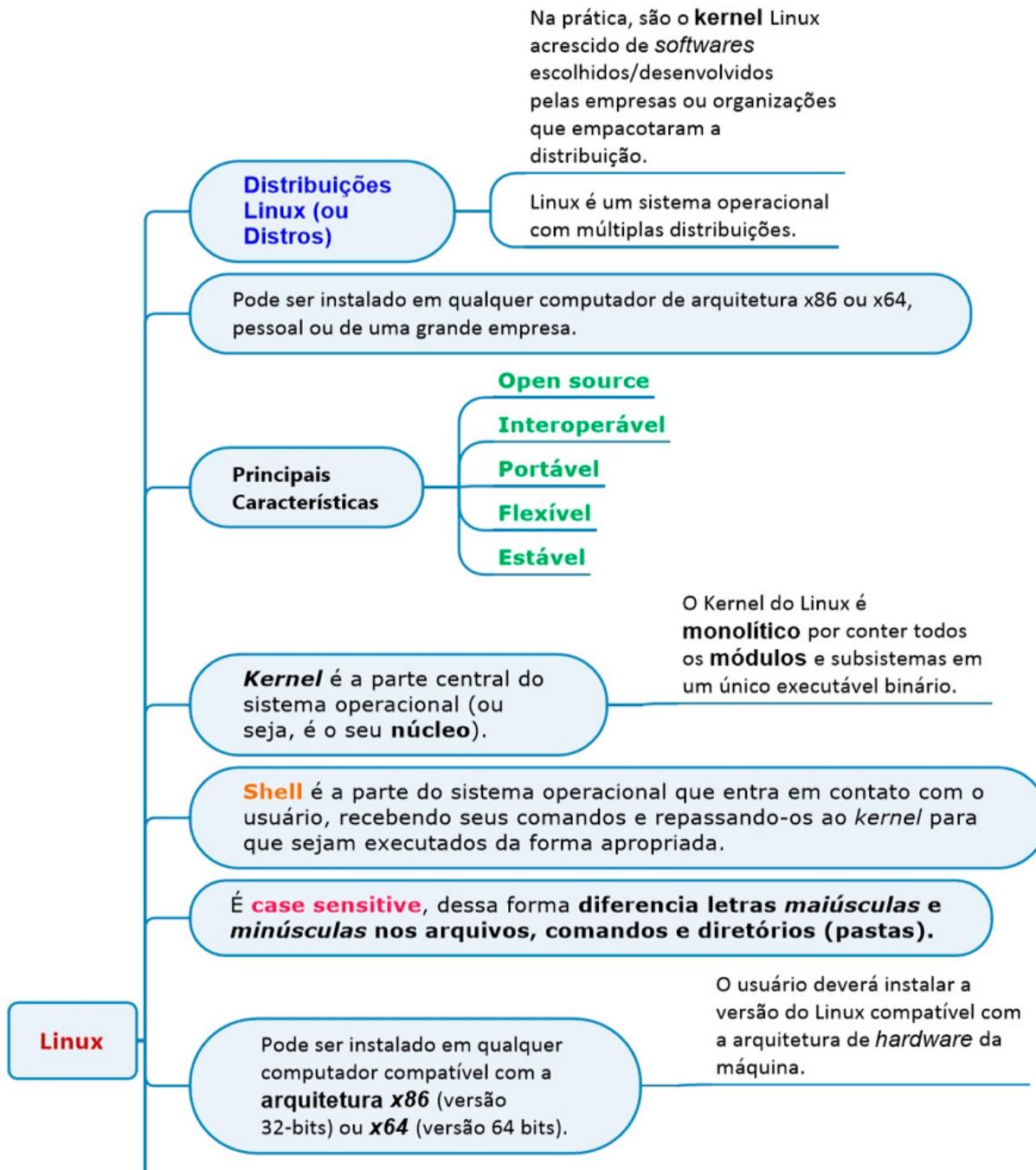
Para instalar e configurar o **DNS**, o software mais utilizado para resolução de nomes em ambientes Linux é conhecido por: **BIND**.

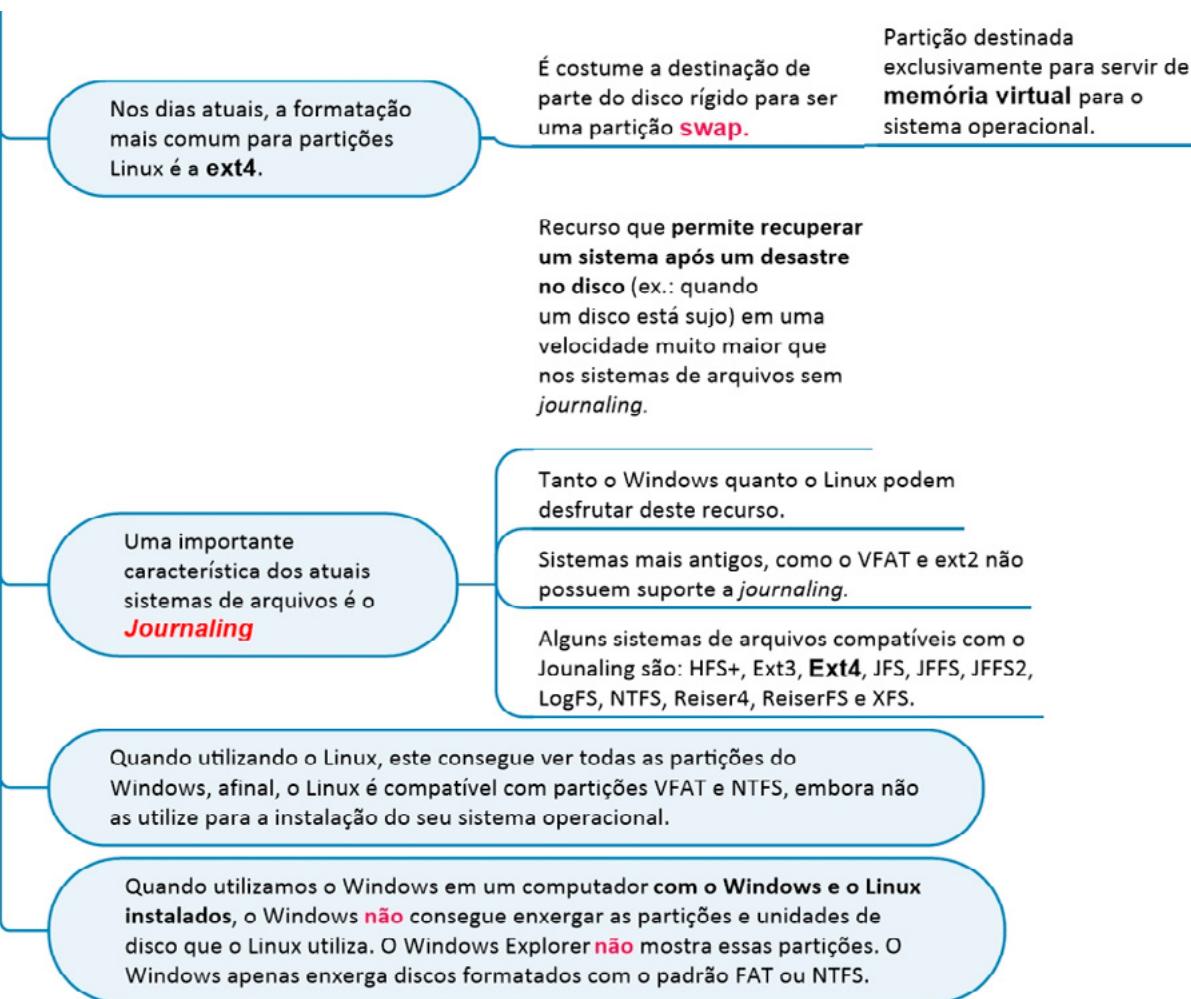
O servidor DNS open source mais utilizado em servidores Linux é o BIND.

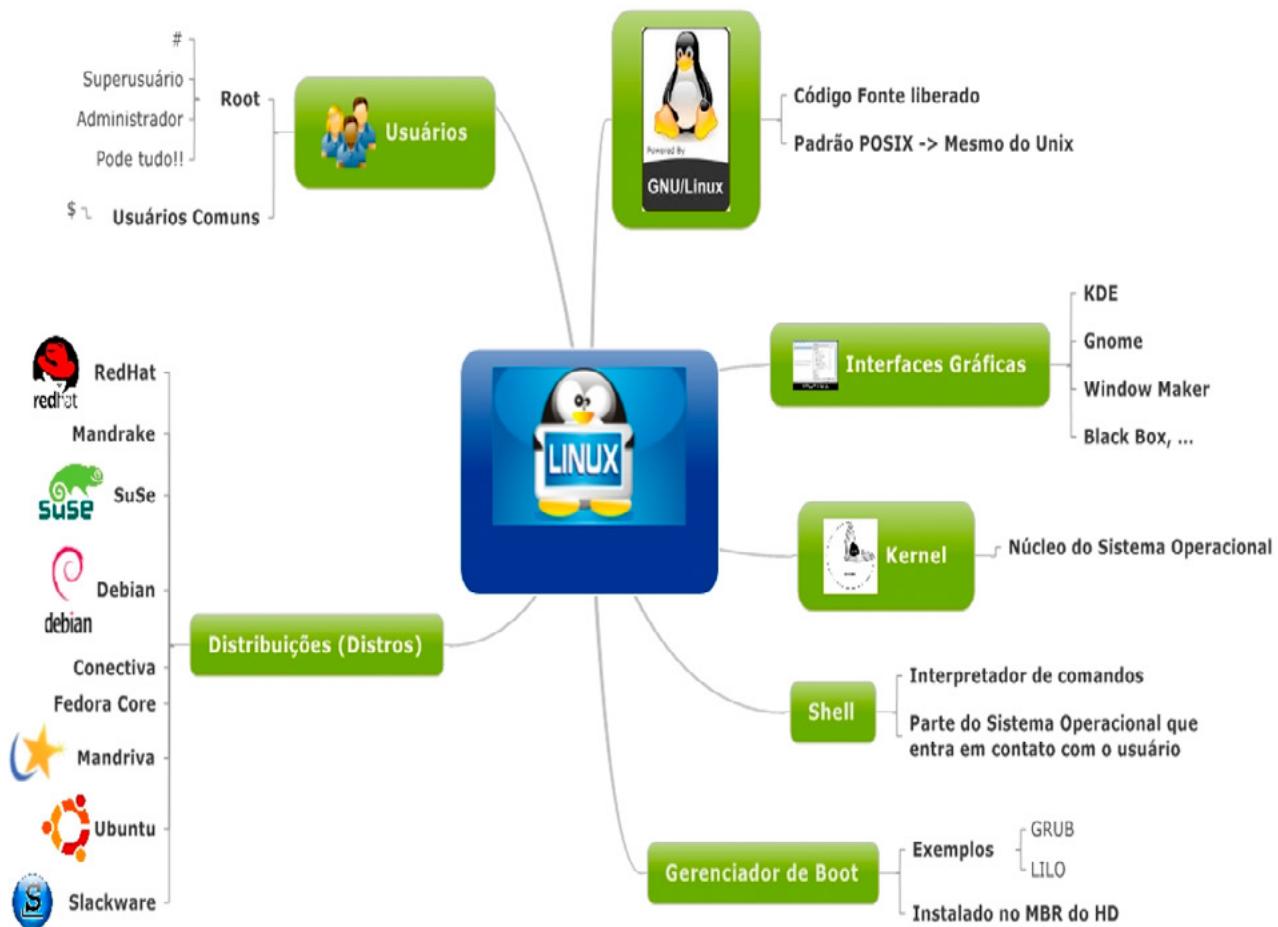
## NAVEGADORES WEB UTILIZADOS NO LINUX

- **Galeon, Ópera, Lynx e Konqueror são exemplos de navegadores web** utilizados no Linux
- **Galeon**: é um navegador de web para o ambiente GNOME, baseado no Mozilla. Praticamente todas as distribuições Linux têm este navegador em seus repositórios.
- **Ópera**: é um navegador da web multi-plataforma muito utilizado.
- **Konqueror**: é um navegador web que pode também funcionar como visualizador de arquivos. Faz parte da suíte KDE. Voltado para plataformas Linux.
- **Lynx**: é um navegador web capaz de exibir apenas texto e que pode ser utilizado em linha de comandos, ideal para sistemas baseados em console ou com poucos recursos gráficos que só mostra texto. Presente em diversas distribuições de sistemas POSIX (Unix, Linux etc.), além de possuir versões para sistemas da Microsoft, como o MS-DOS e as versões do Windows.

## RESUMO







A seguir temos os principais comandos do Linux que são exigidos em provas! **Muita atenção!**



## Mais Comandos

| Comando         | Descrição  |
|-----------------|--|
| <b>cat</b>      | <b>Exibe o conteúdo de um arquivo, sem pausa.</b>                      |
| <b>chmod</b>    | Altera as permissões de arquivos e diretórios.                         |
| <b>chown</b>    | Altera o dono e o grupo dono de um arquivo ou diretório.               |
| <b>clear</b>    | Limpa a tela e posiciona o cursor no canto superior esquerdo do vídeo. |
| <b>cmp</b>      | Compara arquivos.  |
| <b>cp</b>       | Copia arquivos e diretórios.   |
| <b>date</b>     | Exibe ou altera a data do sistema.                                     |
| <b>df</b>       | Exibe informações sobre o espaço dos discos.                           |
| <b>echo</b>     | Exibe texto na tela.   |
| <b>fdisk</b>    | <b>Edita partições de um disco.</b>                                    |
| <b>file</b>     | <b>Exibe o tipo de um arquivo.</b>                                     |
| <b>find</b>     | Procura arquivos.  |
| <b>free</b>     | Exibe o estado da memória RAM e memória virtual.                       |
| <b>grep</b>     | Filtrar o conteúdo de um arquivo.                                      |
| <b>groupadd</b> | Adiciona grupos.   |
| <b>head</b>     | Mostra as linhas iniciais de um arquivo texto.                         |
| <b>history</b>  | Mostra os últimos comandos executados pelo usuário.                    |
| <b>kill</b>     | Envia um sinal a um processo. Utilizado para “matar processos”.        |
| <b>less</b>     | Exibe o conteúdo de um arquivo de texto pausadamente.                  |
| <b>ln</b>       | Cria <i>links</i> para arquivos e diretórios no sistema.               |
| <b>login</b>    | Permite a entrada de um usuário no sistema.                            |
| <b>ls –la</b>   | <b>Lista todos os arquivos (inclusive os ocultos).</b>                 |
| <b>man</b>      | Exibe o manual de um comando.  |
| <b>more</b>     | Exibe o conteúdo de um arquivo.  |

| Comando         | Descrição  |
|-----------------|--|
| <b>mount</b>    | Monta unidades de disco rígido, disquete, CD-ROM.  |
| <b>mv</b>       | Move ou renomeia arquivos e diretórios.  |
| <b>netstat</b>  | Exibe informações sobre as conexões de rede ativas.  |
| <b>passwd</b>   | Altera a senha de usuários.  |
| <b>ps</b>       | Informações sobre processos em execução no sistema.  |
| <b>rpm</b>      | Gerencia pacotes Red Hat.  |
| <b>shutdown</b> | <b>Desliga o sistema de modo seguro.</b>   |
| <b>su</b>       | Troca usuário. Permite trabalhar momentaneamente como outro usuário.   |
| <b>tail</b>     | Exibe o final do conteúdo de um arquivo.   |
| <b>tar</b>      | <b>(Tape ARchive) Trata-se de uma aplicação orientada para backup. Ela agrupa vários arquivos em um só, sem compactar!</b> |
| <b>tree</b>     | Exibe arquivos e diretórios em forma de árvore.  |
| <b>umount</b>   | Desmonta unidades.   |
| <b>uname</b>    | Exibe informações sobre o tipo de UNIX/Linux, kernel, etc.   |
| <b>useradd</b>  | Adiciona usuários.   |
| <b>userdel</b>  | Exclui usuário do sistema.   |
| <b>usermod</b>  | Modifica usuário do sistema.   |
| <b>who</b>      | Exibe os usuários logados no sistema.  |
| <b>who am i</b> | Exibe o nome do usuário logado.  |

| Compactadores/Descompactadores |  |
|--------------------------------|--|
| <b>gzip</b>                    | <p>Usado para gerar uma cópia compactada de um determinado arquivo.</p> <ul style="list-style-type: none"> <li>O que ele <b>não</b> realiza é a união de vários arquivos em um único arquivo. Para isso existe uma aplicação chamada de <b>empacotador</b>. E essa função específica é desempenhada pelo <b>tar</b>.</li> </ul>  |
| <b>gunzip</b>                  | <p>Para <b>descompactar</b> um arquivo com a extensão <b>.gz</b>, retornando o arquivo ao seu estado original.</p> <p>Ex.: <b>gunzip linux.pdf.gz</b><br/>     Pode-se também usar o <b>gzip -d linux.pdf.gz</b>.<br/>     Nos 2 comandos acima, usei como exemplo o arquivo <b>linux.pdf.gz</b>.</p>  |

**Vamos praticar mais agora :-)**

## QUESTÕES COMENTADAS EM AULA

**001.** (CESPE/2018/FUB/CONHECIMENTOS BÁSICOS/CARGOS DE NÍVEL MÉDIO) Acerca do sistema operacional Linux, do gerenciador de arquivos Windows Explorer e do Microsoft Excel 2010, julgue o item subsequente. Há uma diversidade grande de distribuições do sistema Linux, cada uma delas com estrutura de diretórios e kernel diferentes.

**002.** (CESPE/TRE-RJ/CBNS/2012) No Linux, em um mesmo diretório, não podem existir dois subdiretórios com o mesmo nome, contudo, em virtude de os nomes dos diretórios serem case sensitive, é possível criar dois subdiretórios de nomes /usr/TreRJ e /usr/trerj.

**003.** (CESPE/2011/EBC/CARGOS DE NÍVEL SUPERIOR/CONHECIMENTOS BÁSICOS/EXCETO CARGO 4 ADVOCACIA) Na árvore de diretórios do Linux, o /lib contém os programas necessários à inicialização do sistema, e o /home contém os arquivos e diretórios dos usuários.

**004.** (CESPE/2015/TCU/TÉCNICO FEDERAL DE CONTROLE EXTERNO/CONHECIMENTOS BÁSICOS) A respeito dos sistemas operacionais Linux e Windows, do Microsoft PowerPoint 2013 e de redes de computadores, julgue o item a seguir.

No console de uma distribuição qualquer do Linux, como, por exemplo, o Ubuntu, é possível cancelar um comando em execução a partir do uso da combinação das teclas CTRL + C.

**005.** (CESPE/CGM-PB/2018) Julgue o item, relativo aos sistemas operacionais Linux e Windows e ao editor de texto Microsoft Word 2013.

No Linux, a senha de usuário pode ser alterada via terminal por meio do comando passwd, mas o usuário, com exceção do root, não consegue alterar sua própria senha.

**006.** (CESPE/2012/PEFOCE) O comando pwd do Linux possibilita ao usuário efetuar a troca de senha (password).

**007.** (CESPE/TRE-RJ/CBNS/2012) No Linux, a sintaxe sudo adduser fulano criará o usuário fulano no grupo /etc/skell, bem como criará o diretório /home/fulano.

**008.** (CESPE/2016/TRE-PI/NÍVEL SUPERIOR/CONHECIMENTOS BÁSICOS/TODOS OS CARGOS) Assinale a opção que apresenta o comando que um usuário deve utilizar, no ambiente Linux, para visualizar, em um arquivo de texto (nome-arquivo), apenas as linhas que contenham determinada palavra (nome-palavra).

- a) pwd nome-arquivo | locate nome-palavra
- b) find nome-palavra | ls -la nome-arquivo
- c) cat nome-arquivo | grep nome-palavra
- d) lspci nome-arquivo | find nome-palavra
- e) cd nome-arquivo | search nome-palavra

## QUESTÕES DE CONCURSO

**009.** (INÉDITA/2020) No Sistema Operacional Linux, o comando que altera o usuário dono de determinado arquivo ou diretório é stat.



Vamos aos comandos mencionados na questão:

- **chown**: altera o dono de um arquivo ou diretório. Opcionalmente, pode também ser usado para mudar o grupo.
- **stat**: mostra informações detalhadas de um arquivo ou diretório, como: tipo de arquivo, permissão de acesso, ID do usuário e do grupo, data de acesso, modificação e criação do arquivo.

**Errado.**

**010.** (INSTITUTO AOCP/2016/CASAN/ADVOGADO) Um determinado usuário administrativo (superusuário) do sistema operacional Linux disparou o seguinte comando em um X-terminal no diretório conhecido como raiz ou barra: rm A\*. O resultado desse comando, após pressionar a tecla Enter do teclado, foi

- listar os arquivos do diretório que iniciam com a letra A.
- fazer um backup dos arquivos do sistema.
- remover todos os arquivos do sistema.
- apagar os diretórios iniciando com a letra A.
- apagar todos os arquivos iniciando com a letra A.



O comando **rm** é utilizado para remover diretórios, vazios ou não, e arquivos.

Utilize a chave “**-r**” para **remover diretórios**, recursivamente, pedindo confirmação para apagar os arquivos e os subdiretórios encontrados. Não pedirá confirmação se for utilizado em conjunto com a chave **-f**.

Exemplos:

| Comando               | Resultado   |
|-----------------------|---|
| #rm A*                | <b>Exclui todos os arquivos iniciando com a letra A, que se encontram no diretório atual.</b> |
| #rm Carta.txt         | Exclui o arquivo “Carta.txt”, que se encontra no diretório atual.                             |
| #rm MeusDocumentos -r | Exclui o diretório “MeusDocumentos”: o argumento “ <b>-r</b> ” indica diretório.              |

| Comando               | Resultado  |
|-----------------------|--|
| #rm -r MeusDocumentos | Exclui o diretório “MeusDocumentos” sem pedir confirmação “-f”.  |
| #rm teste             | Remove o arquivo “teste”, que se encontra no diretório atual.  |
| #rm -rf mala          | Remove o diretório sem pedir confirmação. A ação será recursiva.   |
| #rm -rf /             | Apaga todos os arquivos e diretórios, a partir da raiz do sistema, recursivamente e sem pedir confirmação. <b>Isso irá apagar todo o sistema operacional e todos os dados!</b> |

**Nota:** Alguns caracteres podem ser utilizados como curingas em alguns comandos, ou seja, podem substituir outros caracteres ou sequências de caracteres. Esses curingas são o asterisco (\*), a interrogação (?) e os colchetes ([ ]).

|          |  |
|----------|--|
| *        | O asterisco representa “de zero a infinitos caracteres”.   |
| ?        | A interrogação pode ser interpretada como <b>um caractere qualquer</b> . No entanto, ela substitui <b>apenas</b> um caractere. Em outras palavras, deverá existir um caractere para cada interrogação.   |
| [padrão] | Os colchetes servem para definir possibilidades de recursos. Substituem um conjunto de caracteres especificados dentro dos colchetes.<br>Os caracteres circunflexo (^) e exclamação (!) podem ser usados como negação ( <b>apenas</b> para <b>um</b> caractere). O caractere <b>hífen</b> pode ser utilizado para definir <b>intervalos</b> , seguindo a tabela ASCII. |

Exemplos de sintaxe com colchetes:

|           |  |
|-----------|--|
| [A-Z]     | Qualquer caractere de A a Z, com caixa alta (maiúsculos).                              |
| [ABCde15] | Qualquer um dos caracteres citados, considerando a caixa dos mesmos no caso de letras. |
| [0-9]     | Qualquer algarismo, de 0 a 9.  |
| [!t]      | Qualquer caractere exceto o t.   |

|            |   |
|------------|---|
| [Tt]este   | Qualquer arquivo ou diretório que inicie com o caractere T ou com t e que termine com este. |
| [a-z][0-9] | Usado para trabalhar com caracteres de a até z seguidos de um caractere de 0 até 9.         |
| [a-z,1,0]  | Faz referência do intervalo de caracteres de a até z ou 1 ou 0 naquela posição.             |

### Letra e.

**011.** (AOCP/2013/PREFEITURA DE PARANAVAÍ-PR/NUTRICIONISTA) Considerando o Sistema Operacional Linux – assinale a alternativa correta.

- a) Windows é uma distribuição do Sistema Operacional Linux.
- b) No Linux o comando “ls” inicia o editor de texto.
- c) No Linux o comando “vi” permite listar os arquivos e diretórios do diretório atual.
- d) No Linux, a partição swap é uma extensão da sua memória RAM.
- e) O Linux é um software pago, desenvolvido por programadores experientes espalhados ao redor do mundo.



- a) Errada. Uma distribuição é uma versão do Linux empacotada por um determinado responsável (pessoa ou empresa), e que compreende um conjunto de programas formado pelo Kernel Linux e por mais alguns softwares distintos (como shells, aplicativos, jogos, utilitários etc.). Principais distribuições: Slackware, RedHat, SuSe, Mandriva, Debian, Ubuntu, Fedora. Windows é um sistema operacional proprietário, da Microsoft.
- b) Errada. A banca trocou as letras B e C. No Linux, o “vi” é um exemplo de comando que inicia um editor de texto.
- c) Errada. No Linux o comando “ls” permite listar os arquivos e diretórios do diretório atual.
- d) Certa. No Linux, a partição SWAP é usada como memória virtual, em apoio à memória RAM.

A **memória virtual** é uma técnica que consiste em reservar parte da memória secundária (no caso, o disco rígido) para ser uma extensão da RAM. Com isso, tem-se a sensação de que a memória é maior do que realmente é.

- e) Errada. O Linux é um software gratuito.

### Letra d.

**012.** (INÉDITA/2021) O Linux, assim como o Windows, diferencia letras maiúsculas e minúsculas nos arquivos, comandos e diretórios (pastas).



Somente o **Linux**, dentre os sistemas mencionados na questão, é **Case Sensitive**, pois **diferencia letras maiúsculas e minúsculas nos arquivos, comandos e diretórios (pastas)**. Assim, pode-se criar os subdiretórios de nomes /usr/ANVISA e /usr/Anvisa, que são distintos para o Linux.

**Errado.**

---

**013.** (CESPE/ANVISA/2016) O sistema operacional Linux, embora seja amplamente difundido, está indisponível para utilização em computadores pessoais, estando o seu uso restrito aos computadores de grandes empresas.



O sistema operacional Linux atualmente já é amplamente utilizado no âmbito doméstico e no corporativo, podendo ser instalado em computadores pessoais (de usuários) e em equipamentos de empresas de todos os tamanhos.



O uso de negação e menosprezo em uma assertiva, como ocorreu nessa questão, **geralmente**, é sinal de erro na maioria das questões do CESPE.

**Errado.**

---

**014.** (CESPE/2016/PC-PE/CONHECIMENTOS BÁSICOS/TODOS OS CARGOS) Para aferir o uso da CPU e da memória de uma estação de trabalho instalada com Linux, deve(m) ser utilizado(s) o(s) comando(s)

- a) top.
- b) system.
- c) proc e mem.
- d) cpu e memory.
- e) fs e du.



| Comando | Descrição   |     |      |       |      |      |     |      |      |          |                 |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |
|---------|---|-----|------|-------|------|------|-----|------|------|----------|-----------------|-------|---------|------|---------|----|---|------|-----|-----|---|----|-----|---------|--------|------|---------|----|---|------|-----|-----|---|---|-----|---------|-----------------|------|---------|----|---|-------|------|-----|---|---|------|----------|---------|------|------|----|---|-------|-----|------|---|---|-----|---------|------|
| top     | <p>Exibe, em tempo real, os processos que estão sendo executados no sistema. Também permite a manipulação de processos por meio de comandos interativos.</p> <p>Veja uma tela de saída desse comando:</p> <pre>Cpu(s): 21.5%us, 5.6%sy, 0.0%ni, 63.7%id, 9.2%wa, 0.0%hi, 0.0%si, 0.0%st Mem: 3058052k total, 2083212k used, 974840k free, 167964k buffers Swap: 3481788k total, 0k used, 3481788k free, 1090496k cached</pre> <table border="1"> <thead> <tr> <th>PID</th><th>USER</th><th>PR</th><th>NI</th><th>VIRT</th><th>RES</th><th>SHR</th><th>S</th><th>%CPU</th><th>%MEM</th><th>TIME+</th><th>COMMAND</th></tr> </thead> <tbody> <tr> <td>2398</td><td>ricardo</td><td>20</td><td>0</td><td>428m</td><td>76m</td><td>29m</td><td>S</td><td>15</td><td>2.6</td><td>2:48.27</td><td>compiz</td></tr> <tr> <td>5489</td><td>ricardo</td><td>20</td><td>0</td><td>169m</td><td>14m</td><td>10m</td><td>S</td><td>9</td><td>0.5</td><td>0:00.28</td><td>gnome-screensho</td></tr> <tr> <td>3187</td><td>ricardo</td><td>20</td><td>0</td><td>1042m</td><td>393m</td><td>42m</td><td>S</td><td>8</td><td>13.2</td><td>27:00.38</td><td>firefox</td></tr> <tr> <td>1238</td><td>root</td><td>20</td><td>0</td><td>92920</td><td>26m</td><td>8724</td><td>S</td><td>7</td><td>0.9</td><td>4:22.79</td><td>Xorg</td></tr> </tbody> </table> <p>Fonte: <a href="http://www.linuxdescomplicado.com.br/2013/12/comandos-linux-dominando-o-comando-top.html">http://www.linuxdescomplicado.com.br/2013/12/comandos-linux-dominando-o-comando-top.html</a></p> <p>Conforme visto, <b>top</b> permite aferir o uso da CPU e da memória de uma estação de trabalho instalada com Linux.</p> | PID | USER | PR    | NI   | VIRT | RES | SHR  | S    | %CPU     | %MEM            | TIME+ | COMMAND | 2398 | ricardo | 20 | 0 | 428m | 76m | 29m | S | 15 | 2.6 | 2:48.27 | compiz | 5489 | ricardo | 20 | 0 | 169m | 14m | 10m | S | 9 | 0.5 | 0:00.28 | gnome-screensho | 3187 | ricardo | 20 | 0 | 1042m | 393m | 42m | S | 8 | 13.2 | 27:00.38 | firefox | 1238 | root | 20 | 0 | 92920 | 26m | 8724 | S | 7 | 0.9 | 4:22.79 | Xorg |
| PID     | USER  | PR  | NI   | VIRT  | RES  | SHR  | S   | %CPU | %MEM | TIME+    | COMMAND         |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |
| 2398    | ricardo   | 20  | 0    | 428m  | 76m  | 29m  | S   | 15   | 2.6  | 2:48.27  | compiz          |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |
| 5489    | ricardo   | 20  | 0    | 169m  | 14m  | 10m  | S   | 9    | 0.5  | 0:00.28  | gnome-screensho |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |
| 3187    | ricardo   | 20  | 0    | 1042m | 393m | 42m  | S   | 8    | 13.2 | 27:00.38 | firefox         |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |
| 1238    | root  | 20  | 0    | 92920 | 26m  | 8724 | S   | 7    | 0.9  | 4:22.79  | Xorg            |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |
| du      | Exibe o tamanho de arquivos e/ou diretórios.  |     |      |       |      |      |     |      |      |          |                 |       |         |      |         |    |   |      |     |     |   |    |     |         |        |      |         |    |   |      |     |     |   |   |     |         |                 |      |         |    |   |       |      |     |   |   |      |          |         |      |      |    |   |       |     |      |   |   |     |         |      |

## Letra a.

**015.** (CESPE/2016/PC-GO/CONHECIMENTOS BÁSICOS) Para o correto funcionamento de determinado ambiente computacional, é necessário que o programa gravado no diretório seja executado simultaneamente aos outros programas do sistema operacional Linux que estejam em execução.

- A respeito dessa situação, é correto afirmar que a execução do programa **xpto**,
- a) pode ser verificada por meio do comando **ls xpto | /sys/proc**.
  - b) não ocorrerá, pois o programa se encontra no diretório **/home**, onde o Linux não permite gravação de arquivos binários.
  - c) pode ser verificada por meio do comando **ps -ef | grep xpto**.
  - d) pode ser verificada por meio do comando **ls /home/fulano/xpto | proc**.
  - e) pode ser verificada por meio do comando **ls process xpto | /sys/proc**.



**Conectar comandos** faz com que o **resultado gerado por um comando seja processado por outro comando**, mediante a aplicação do caractere **pipe** (|).

O caractere pipe (|) é utilizado para enviar a saída de um comando para a entrada do próximo comando. Os dados enviados são processados pelo próximo comando que mostrará o resultado do processamento.

O comando **grep** procura *strings* (palavras ou expressões em arquivos).

O comando **ps -ef** possibilita uma listagem de todos os processos em execução na máquina, levando-se em consideração as opções listadas a seguir:

- “e” exibe as variáveis de ambiente relacionadas aos processos;
- “f” exibe a árvore de execução dos processos.

O resultado de **ps -ef**, ao invés de ser exibido na tela, será enviado para o comando **grep xpto**, que vai selecionar somente as linhas que contenham a palavra **xpto**. Esse último resultado será exibido na tela.

Assim a execução do comando **ps-ef | grep xpto** mostrará uma lista de processos em execução que tenham em sua descrição a sequência de caracteres **xpto**.

**Letra c.**

**016.** (CESPE/ANS/CARGO 3/2013) Com relação aos sistemas operacionais Linux e Windows e às redes de computadores, julgue o item seguinte: Tanto o sistema operacional Linux quanto o Windows possuem gerenciador de arquivos, que permite a organização dos dados em pastas e subpastas, também denominadas, respectivamente, diretórios e subdiretórios.



No Windows temos o Windows Explorer e no Linux, várias opções, como por exemplo o **Nautilus**, o **Konqueror**, o **Dolphin**, que nos permite organizar os dados em pastas (diretórios) e subpastas (subdiretórios).

**Certo.**

**017.** (CESPE/2015/TRE-GO/ANALISTA JUDICIÁRIO) Julgue o item a seguir, a respeito de organização e de gerenciamento de arquivos.

No Linux, todo arquivo executável tem como extensão o sufixo .exe.



Não! O arquivo executável no Linux pode ter qualquer nome e extensão, bastando aplicar as permissões de execução corretas nesse arquivo. Com o comando chmod são destacadas as

pessoas que terão acesso ao arquivo e tipos de permissões que podem ser aplicadas a essas pessoas. No Windows, o sufixo .exe indica um arquivo executável, no entanto, outras extensões também podem ser utilizadas para esse tipo de arquivo, como .com, .msi etc.

**Errado.**

**018.** (CESPE/2015/TRE-GO/ANALISTA JUDICIÁRIO) Julgue o item a seguir, a respeito de noções de sistema operacional.

No Linux, o comando cd /etc/teste cria o diretório teste dentro do diretório /etc.



O comando **cd** (change directory) permite entrar ou sair de um determinado diretório (ou pasta). Exemplo:

| Comando                       | Resultado  |
|-------------------------------|--|
| <b>#cd /etc/teste</b>         | Entra no diretório “teste”.  |
| <b>#cd</b>                    | Retorna ao diretório do usuário atual.                             |
| <b>#cd Desktop</b>            | Entra no diretório “Desktop”.                                      |
| <b>#cd Docs/Textos/Cartas</b> | Entra no diretório “Cartas”.                                       |
| <b>#cd ..</b>                 | Sai do diretório atual e vai para o diretório de nível logo acima. |
| <b>#cd ../../</b>             | Sobe dois níveis da árvore de diretórios.                          |
| <b>#cd -</b>                  | Altera entre o diretório atual e o anteriormente visitado.         |
| <b>#cd ~</b>                  | Vai para o diretório “home” do usuário atual.                      |

**Errado.**

**019.** (CESPE/2015/TRE-GO/TÉCNICO JUDICIÁRIO/ÁREA ADMINISTRATIVA) Julgue o item a seguir, acerca de sistemas operacionais.



No Linux, a execução do comando **ps-aexf | grep arq** mostrará uma lista de processos em execução que tenham em sua descrição a sequência de caracteres arq.

**Conegar comandos** faz com que o **resultado gerado por um comando seja processado por outro comando**, mediante a aplicação do caractere **pipe (|)**.

O caractere pipe (`|`) é utilizado para enviar a saída de um comando para a entrada do próximo comando. Os dados enviados são processados pelo próximo comando que mostrará o resultado do processamento.

O comando **grep** procura *strings* (palavras ou expressões em arquivos).

O comando **ps -aexf** possibilita uma listagem de todos os processos em execução na máquina, levando-se em consideração as opções listadas a seguir:

- “a” mostra todos os processos existentes;
- “e” exibe as variáveis de ambiente relacionadas aos processos;
- “x” lista até os processos que não estão associados a terminais; e
- “f” exibe a árvore de execução dos processos.

O resultado de **ps -aexf**, ao invés de ser exibido na tela, será enviado para o comando **grep arq**, que vai selecionar somente as linhas que contenham a palavra **arq**. Esse último resultado será exibido na tela.

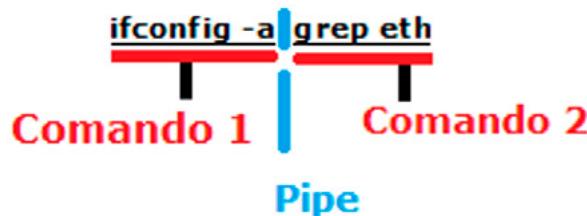
Assim a execução do comando **ps-aexf | grep arq** mostrará uma lista de processos em execução que tenham em sua descrição a sequência de caracteres **arq**.

**Certo.**

**020.** (CESPE/2012/TRE-RJ/TÉCNICO JUDICIÁRIO ADMINISTRATIVA) No Linux, a sintaxe **ifconfig -a|grep eth** permite identificar as interfaces Ethernet configuradas.



Primeiramente é importante destacar sobre a conexão de comandos, utilizada na questão. Conectar comandos é fazer com que o resultado gerado por um comando seja processado por outro comando, mediante a aplicação do caractere pipe (`|`).



O comando **ifconfig** mostra as configurações de todos os adaptadores de rede (placas de rede) ativos na máquina. Com o argumento **-a**, mostrará o status de todas as interfaces, mesmo as não ativas.

Assim, o primeiro comando **ifconfig -a** mostra as configurações de todos os adaptadores de rede (placas de rede) ativos e não ativos na máquina.

O comando **grep** é usado para procurar por linhas em um arquivo que contenham expressões que satisfaçam a um determinado padrão de busca. Em **grep eth** o sistema irá procurar por entradas na saída do primeiro comando que tenham a expressão “**eth**”.

Assim, **ifconfig -a | grep eth** permite identificar as interfaces de rede Ethernet configuradas.

**Certo.**

---

**021.** (CESPE/TRE-RJ/CBNS/2012) No Linux, o diretório /bin contém programas do sistema que são utilizados pelos usuários, não sendo necessário, para que esses programas sejam executados, que eles possuam a extensão .exe.



Os programas executáveis do GNU/Linux, ao contrário dos programas de DOS e Windows, não são executados a partir de extensões .exe, .com ou .bat. O GNU/Linux usa a *permissão de execução* de arquivo para identificar se um arquivo pode ou não ser executado.

**Certo.**

---

**022.** (CESPE/MPE-PI/2012) No sistema Linux, existe um usuário de nome root, que tem poder de superusuário. Esse nome é reservado exclusivamente ao usuário que detém permissão para executar qualquer operação válida em qualquer arquivo ou processo. Há um único superusuário com esse perfil.



Podem ser criados outros usuários no sistema Linux com poderes de superusuários, com privilégios administrativos completos sobre o sistema.

**Errado.**

---

**023.** (CESPE/BRB/ESCRITURÁRIO/2011) Apesar de multiusuário e multiprogramável, o sistema operacional Linux não permite o redirecionamento de entrada e saída de dados.



O sistema operacional Linux é um sistema multiusuário (já que permite que vários usuários estejam logados simultaneamente no sistema) e multiprogramável/multitarefa (permite o gerenciamento de diversos programas ao mesmo tempo).

Também permite o redirecionamento de entrada e saída de dados. Nesse caso, quase todos os comandos Linux têm uma entrada e produzem uma saída. A entrada de um comando são os dados que o comando vai processar, e essa entrada pode vir de um arquivo especificado pelo usuário, de um arquivo do sistema, do terminal ou da saída de outro comando. A saída de um comando são os dados de entrada processados, e pode ser impressa na tela de um terminal, enviada a um arquivo, ou servir de entrada a um outro comando.

**Errado.**

**024.** (CESPE/2011/EBC/CARGOS DE NÍVEL SUPERIOR/CONHECIMENTOS BÁSICOS/ EXCETO CARGO 4 ADVOCACIA) O Windows 7 Professional grava os arquivos em formato nativo ext3 e fat32; o Linux utiliza, por padrão, o formato NTFS, mais seguro que o adotado pelo Windows.



O Linux usa o sistema de arquivos EXT3 e reconhece o HPFS (antigo formato da IBM). O Windows usa NTFS, e reconhece FAT32, FAT16 e FAT12 (FAT). O Linux reconhece todas as partições do ambiente Windows. O Windows não reconhece as partições do ambiente Linux.

**Errado.**

**025.** (CESPE/PRAÇA BOMBEIRO MILITAR OPERACIONAL/QBMG-01/2011) Em algumas das distribuições do Linux disponíveis na Internet, pode-se iniciar o ambiente gráfico por meio do comando startx.



Uma vez tendo iniciado o acesso ao sistema operacional Linux em modo texto, basta executar o comando **startx** para iniciar o modo gráfico.

**Certo.**

**026.** (CESPE/TÉCNICO BANCÁRIO NOVO/NM1/CAIXA ECONÔMICA FEDERAL/2010) O diretório raiz do Linux é o C:\.



O diretório raiz no Linux é o /.

**Errado.**

**027.** (CESPE/TÉCNICO BANCÁRIO/CARREIRA ADMINISTRATIVA- CAIXA-NM1/2010) O acesso à Internet no ambiente Linux é realizado, por padrão, por meio do aplicativo Kernel.



O Kernel é o núcleo do sistema operacional.

**Errado.**

**028.** (CESPE/TÉCNICO BANCÁRIO/CARREIRA ADMINISTRATIVA/CAIXA-NM1/2010) No ambiente Linux, pode haver diversos tipos de interfaces e uso de licenças. Uma dessas interfaces é o KDE, que utiliza a licença GNU.



O KDE é um dos ambientes gráficos disponíveis no Linux e é bastante utilizado.

**Certo.**

**029. (CESPE/TÉCNICO BANCÁRIO/CARREIRA ADMINISTRATIVA/CAIXA-NM1/2010)**

Apache é a denominação de uma comunidade de desenvolvedores de software gratuito para acesso à Web, que foi formada nos Estados Unidos da América, mas hoje está disseminada em vários países, inclusive no Brasil.



O **servidor Apache** (ou Servidor HTTP Apache, em inglês: *Apache HTTP Server*) é o mais bem sucedido servidor Web livre.

**Errado.**

**030. (CESPE/ANALISTA TÉCNICO ADMINISTRATIVO/DPU/2010)** O Linux oferece facilidade de interação entre software de diversas plataformas; no entanto, não permite que sejam criados drivers de configuração para que outros hardware possam rodar no Linux.

O Linux possibilita a criação de *drivers* de configurações específicos para que o sistema operacional reconheça outros hardwares além dos já conhecidos.

**Errado.**

**031. (CESPE/ANALISTA TÉCNICO ADMINISTRATIVO/DPU/2010)** O kernel é um software que se instala dentro do Linux e faz com que o Linux possa ser distribuído gratuitamente.

O Kernel é o núcleo do sistema Linux. É a parte do sistema operacional que “fala” diretamente com o hardware do computador.

**Errado.**

**032. (CESPE/ANALISTA TÉCNICO ADMINISTRATIVO/DPU/2010)** O Linux oferece a opção de que um novo usuário possa abrir uma sessão de uso do ambiente para utilizar seus aplicativos mesmo que outro usuário esteja logado no sistema.

Como o sistema operacional Linux é multiusuário, essa ação será possível de ser realizada no ambiente.

**Certo.**

**033. (CESPE/AGENTE ADMINISTRATIVO/DPU/2010)** Em uma distribuição do sistema Linux, é possível encontrar software destinados a diversas finalidades, como para prover serviço de acesso à Internet. No ambiente Linux, também se dispõe de uma área de trabalho (GUI) para uso do sistema operacional a partir de uma interface gráfica.



Uma *distribuição* é uma versão do Linux empacotada por um determinado responsável (pessoa ou empresa), e que compreende um conjunto de programas formado pelo Kernel Linux e por mais alguns softwares distintos (como jogos, utilitários para prover serviço de acesso à Internet etc.). É possível encontrar no Linux uma área de trabalho (GUI) para uso do sistema a partir de uma interface gráfica. Dentre os ambientes gráficos existentes podemos destacar: o KDE e o GNOME, mais conhecidos, seguidos pelos Xfce, WindowMaker, entre outras diversas opções.

**Certo.**

**034.** (CESPE/AGENTE ADMINISTRATIVO/DPU/2010) Pelo fato de ser um software proprietário, qualquer usuário pode fazer alterações no ambiente e colaborar para a melhoria do sistema Linux.



O sistema operacional Linux não é software proprietário, e sim um software livre.

**Errado.**

**035.** (CESPE/AGENTE ADMINISTRATIVO/DPU/2010) O código-fonte do sistema operacional Linux não pode ser alterado; por essa razão ele não é distribuído sob a licença GPL ou GNU, que é pública e permite modificações no código.



O código-fonte do Linux pode ser alterado e é regido pela licença GNU/GPL.

**Errado.**

**036.** (CESPE/AGENTE ADMINISTRATIVO/DPU/2010) KDE Control Center é a área de trabalho do Linux pela qual se faz acesso a aplicativos instalados no computador, como o BrOffice e outros.



O *KDE Control Center* é o gerente de configurações centralizadas para o ambiente de desktop KDE. Em outras palavras, é o local em que se configura diversos aspectos do funcionamento do ambiente gráfico KDE, de forma similar ao Painel de Controle no Windows.

**Errado.**

**037.** (CESPE/AGENTE ADMINISTRATIVO/DPU/2010) O Linux não permite que sejam instalados outros sistemas operacionais na mesma máquina, pois isso afetaria o desempenho do computador, tornando-o lento.



É perfeitamente possível realizar a instalação de mais de um sistema operacional no mesmo equipamento em uma configuração de *dual boot*, por exemplo. Nessa situação, o usuário deverá escolher qual sistema operacional será utilizado, e, após essa escolha, o programa se encarregará de iniciar o sistema operacional desejado.

**Errado.**

**038.** (CESPE/ANALISTA TÉCNICO-ADMINISTRATIVO/MINISTÉRIO DA SAÚDE/2010) No sistema operacional Linux típico, o subdiretório /dev do diretório raiz contém os arquivos executáveis (binários) de comandos essenciais pertencentes ao sistema, e que são usados com frequência pelas aplicações.



O subdiretório /dev do diretório raiz guarda os arquivos de dispositivo e os arquivos binários (também chamados de executáveis) são guardados no /bin e no /sbin.

**Errado.**

**039.** (CESPE/ANALISTA TÉCNICO-ADMINISTRATIVO/MINISTÉRIO DA SAÚDE/2010) No sistema operacional Linux, um ponto (.) no início do nome identifica os arquivos ocultos.



**Os nomes dos arquivos ocultos do Linux começam com um (.) ponto.** Dessa forma, esses arquivos não irão aparecer nas visualizações normais de arquivos.

**Certo.**

**040.** (CESPE/TÉCNICO/TRE-BA/2010) O Linux é um sistema operacional que pode ser usado apenas em servidores, não sendo adequado para a utilização em estações de trabalho do tipo PC. No entanto, é um sistema cujo código-fonte fica disponível para alterações, permitindo que os usuários contribuam para a sua melhoria.



O Linux pode ser utilizado tanto em estações de trabalho do tipo PC, quanto em servidores!

**Errado.**

**041.** (CESPE/ANALISTA JUDICIÁRIO/TRE-GO/2009) Acerca do Internet Explorer e do sistema operacional Linux, assinale a opção correta.

- a)** Para conectar à Internet um microcomputador que tenha instalado o sistema operacional Linux, é necessária a utilização de uma placa de rede específica.
- b)** A conexão, à Internet, de um microcomputador que possui o sistema operacional Linux instalado é mais lenta quando comparada com um que tenha instalado o Windows XP.

- c) Se um e-mail for criado a partir de algum aplicativo do sistema operacional Linux, ele não poderá ser lido por destinatário que usa o Windows XP.
- d) Com o Linux é possível acessar a Internet usando uma rede sem fio (wireless).



Item A. Não é necessária uma placa específica. O item A é **FALSO**.

Item B. Não podemos afirmar isso, sem conhecer mais detalhes sobre a configuração do hardware. O item B é **FALSO**.

Item C. O e-mail criado a partir de algum programa cliente de correio eletrônico (como o Mozilla Thunderbird) que porventura esteja instalado no Linux poderá, com certeza, ser lido em qualquer outro programa cliente de correio eletrônico instalado sob o sistema operacional Windows. O item C é **FALSO**.

Item D. O acesso à rede sem fio poderá ser feito de um equipamento com o sistema operacional Linux, portanto, a assertiva D está **CORRETA**.

**Letra d.**

---

**042.** (CESPE/TÉCNICO JUDICIÁRIO/TRE-GO/2009) A respeito do sistema operacional Linux, assinale a opção correta.

- a) Kernel é a interface gráfica do Linux, que tem visual muito similar à interface do sistema operacional Windows XP.
- b) O Linux funciona em dezenas de plataformas, desde mainframes até relógios de pulso, passando por várias arquiteturas e dispositivos.
- c) O KDE é o navegador nativo do Linux que permite acesso à Internet e envio de e-mail.
- d) O Linux adota a GPL, uma licença que permite aos interessados usá-lo, mas sem a possibilidade de redistribuí-lo.



Item A. Kernel é a parte central do sistema operacional (ou seja, é o seu núcleo). É a parte do sistema Operacional que “fala” diretamente com o hardware do computador. Gerencia a memória; gerencia dispositivos de hardware; diz que sistema de arquivos o sistema operacional usa, como deve usar, e como deve se comportar. Para um sistema funcionar, só se precisa do kernel, todo o resto é complemento. O item A é **FALSO**.

Item B. O Linux hoje funciona em dezenas de plataformas, desde mainframes até um relógio de pulso, passando por várias arquiteturas: x86 (Intel, AMD), x86-64 (Intel EM64T, AMD64), Alpha, SPARC, entre outros. Vide texto original em <http://pt.wikipedia.org/wiki/Linux> (O item B está **CORRETO**).

Item C. O Mozilla Firefox é o navegador nativo do Linux. O KDE não é navegador, é um ambiente gráfico (um programa que apresenta uma interface gráfica amigável para o usuário). (O item C é **FALSO**).

Item D. Como o Linux é livre (**GPL** – Licença pública Geral), ele pode ser adquirido e modificado por qualquer um, que pode distribuí-lo novamente.(A afirmativa D é **FALSA**).

**Letra b.**

**043.** (CESPE/AUXILIAR JUDICIÁRIO PROGRAMADOR/TJ-PA/2006) Os principais elementos estruturais do Linux são os arquivos e os diretórios. Os primeiros guardam informações, e os segundos são compartimentos que guardam arquivos e (ou) outros diretórios. Considerando a estruturação de diretórios e outras características do Linux, é correto afirmar que o comando **pwd** permite a troca da senha (**password**) do usuário corrente.



O comando **pwd** mostra o nome e caminho do diretório atual. O comando adequado para a troca de senhas é o **passwd**.

**Errado.**

**044.** (FCC/2017/ARTESP/AGENTE DE FISCALIZAÇÃO À REGULAÇÃO DE TRANSPORTE/TECNOLOGIA DE INFORMAÇÃO) Existem várias formas de instalar aplicativos no Linux, dependendo da distribuição. Em linha de comando, há duas ferramentas principais que instalam pacotes gerenciando dependências: uma usada em distribuições baseadas em Debian e outra usada em distribuições baseadas em Fedora. Estas ferramentas são acionadas, respectivamente, pelos comandos

- a) rpmdrake e yum.
- b) itargz e rpm.
- c) rpmi e kyum.
- d) apt-get e synaptic.
- e) apt-get e yum.



Há duas ferramentas principais que instalam **pacotes** gerenciando dependências:

| Comandos       | Usos   |
|----------------|--|
| <b>apt-get</b> | Em distribuições baseadas em Debian.   |
| <b>yum</b>     | O <b>yum</b> é o gerenciador de pacotes usado por padrão no <b>Fedora</b> , assumindo o papel que no Debian e no Ubuntu é desempenhado pelo <b>apt-get</b> |

**Letra e.**

**045.** (FCC/2019/BANRISUL/ESCRITURÁRIO) No Linux e no prompt de comandos do Windows, para mostrar a lista de arquivos e diretórios presentes na unidade de armazenamento atual, por exemplo, um pen drive, utilizam-se, respectivamente, os comandos

- a) ls e dir.
- b) list e mkdir.
- c) cat e rmdir.

d) ps e dir.

e) ls e files.



O comando utilizado para listar o conteúdo de um determinado diretório (ou pasta) no Linux é o **ls**. Exemplos:

| Comando         | Resultado   |
|-----------------|---|
| #ls             | Lista o conteúdo da pasta atual.                            |
| #ls Desktop     | Lista o conteúdo da pasta Desktop.                          |
| #ls Docs/Textos | Lista o conteúdo da pasta Textos, localizada na pasta Docs. |
| #ls -l          | Lista detalhada de arquivos e diretórios da pasta.          |
| #ls -a          | Lista todos os arquivos, inclusive os ocultos.              |

No Windows, o comando similar ao **ls** do Linux é o **dir**.

Veja mais:

| Comando | Descrição   |
|---------|---|
| mkdir   | Cria diretórios.  |
| cat     | Exibe o texto contido em um arquivo. Concatena (junta) o conteúdo de arquivos. Cria arquivos baseados em caracteres de texto. |
| ps      | Lista os programas em execução.   |
| list    | -   |

**Letra a.**

## GABARITO

- |       |       |       |
|-------|-------|-------|
| 1. E  | 16. C | 31. E |
| 2. C  | 17. E | 32. C |
| 3. E  | 18. E | 33. C |
| 4. C  | 19. C | 34. E |
| 5. E  | 20. C | 35. E |
| 6. E  | 21. C | 36. E |
| 7. E  | 22. E | 37. E |
| 8. c  | 23. E | 38. E |
| 9. E  | 24. E | 39. C |
| 10. e | 25. C | 40. E |
| 11. d | 26. E | 41. d |
| 12. E | 27. E | 42. b |
| 13. E | 28. C | 43. E |
| 14. a | 29. E | 44. e |
| 15. c | 30. E | 45. a |

## REFERÊNCIAS

QUINTÃO, P. L. **Informática-FCC-Questões Comentadas e Organizadas por Assunto.** 3<sup>a</sup>. edição. Ed. Gen/Método. 2014.

QUINTÃO, P. L. **1001 Questões Comentadas de Informá5tica -Cespe,** 2<sup>a</sup>. Edição. Ed. Gen/ Método, 2017.

<http://www.grupogen.com.br/1001-questoes-comentadas-de-informatica-cespe.html>

QUINTÃO, P. L. **Notas de aula.** 2021.

## Patrícia Quintão



Mestre em Engenharia de Sistemas e computação pela COPPE/UFRJ, Especialista em Gerência de Informática e Bacharel em Informática pela UFV. Atualmente é professora no Gran Cursos Online; Analista Legislativo (Área de Governança de TI), na Assembleia Legislativa de MG; Escritora e Personal & Professional Coach.

Atua como professora de Cursinhos e Faculdades, na área de Tecnologia da Informação, desde 2008. É membro: da Sociedade Brasileira de Coaching, do PMI, da ISACA, da Comissão de Estudo de Técnicas de Segurança (CE-21:027.00) da ABNT, responsável pela elaboração das normas brasileiras sobre gestão da Segurança da Informação.

Autora dos livros: Informática FCC - Questões comentadas e organizadas por assunto, 3<sup>a</sup>. edição e 1001 questões comentadas de informática (Cespe/UnB), 2<sup>a</sup>. edição, pela Editora Gen/Método.

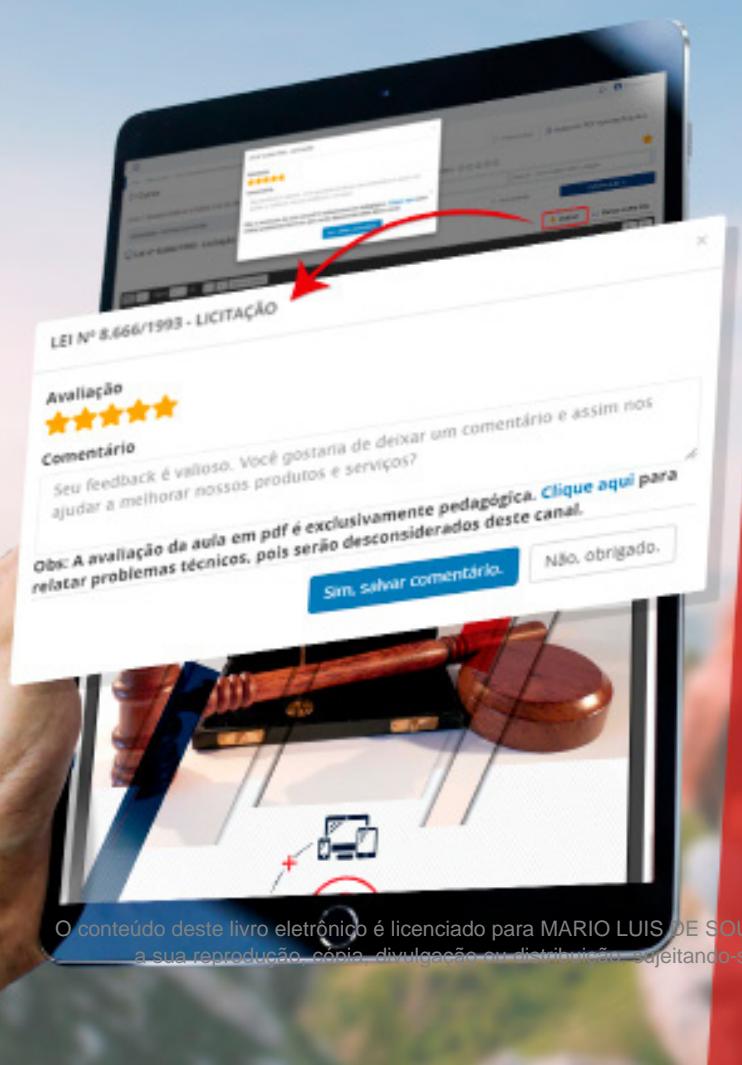
Foi aprovada nos seguintes concursos: Analista Legislativo, na especialidade de Administração de Rede, na Assembleia Legislativa do Estado de MG; Professora titular do Departamento de Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia; Professora substituta do DCC da UFJF; Analista de TI/Suporte, PRODABEL; Analista do Ministério Público MG; Analista de Sistemas, DATAPREV, Segurança da Informação; Analista de Sistemas, INFRAERO; Analista - TIC, PRODEMGE; Analista de Sistemas, Prefeitura de Juiz de Fora; Analista de Sistemas, SERPRO; Analista Judiciário (Informática), TRF 2<sup>a</sup> Região RJ/ES, etc.

@coachpatriciaquintao

/profapatriciaquintao

@plquintao

t.me/coachpatriciaquintao



# NÃO SE ESQUEÇA DE AVALIAR ESTA AULA!

SUA OPINIÃO É MUITO IMPORTANTE  
PARA MELHORARMOS AINDA MAIS  
NOSSOS MATERIAIS.

ESPERAMOS QUE TENHA GOSTADO  
DESTA AULA!

PARA AVALIAR, BASTA CLICAR EM LER  
A AULA E, DEPOIS, EM AVALIAR AULA.

**AVALIAR**