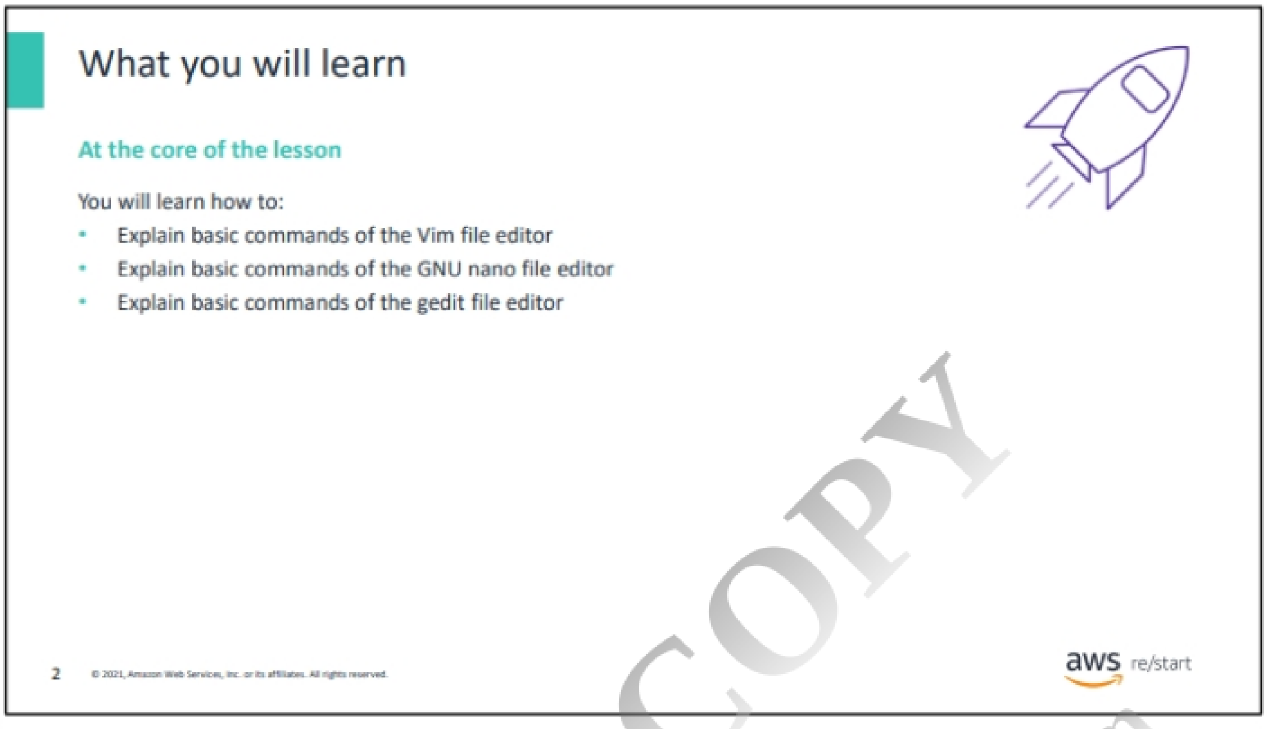
aws re/start

Editing Files

Linux Fundamentals

Welcome to Editing Files.



What you will learn

At the core of the lesson

You will learn how to:

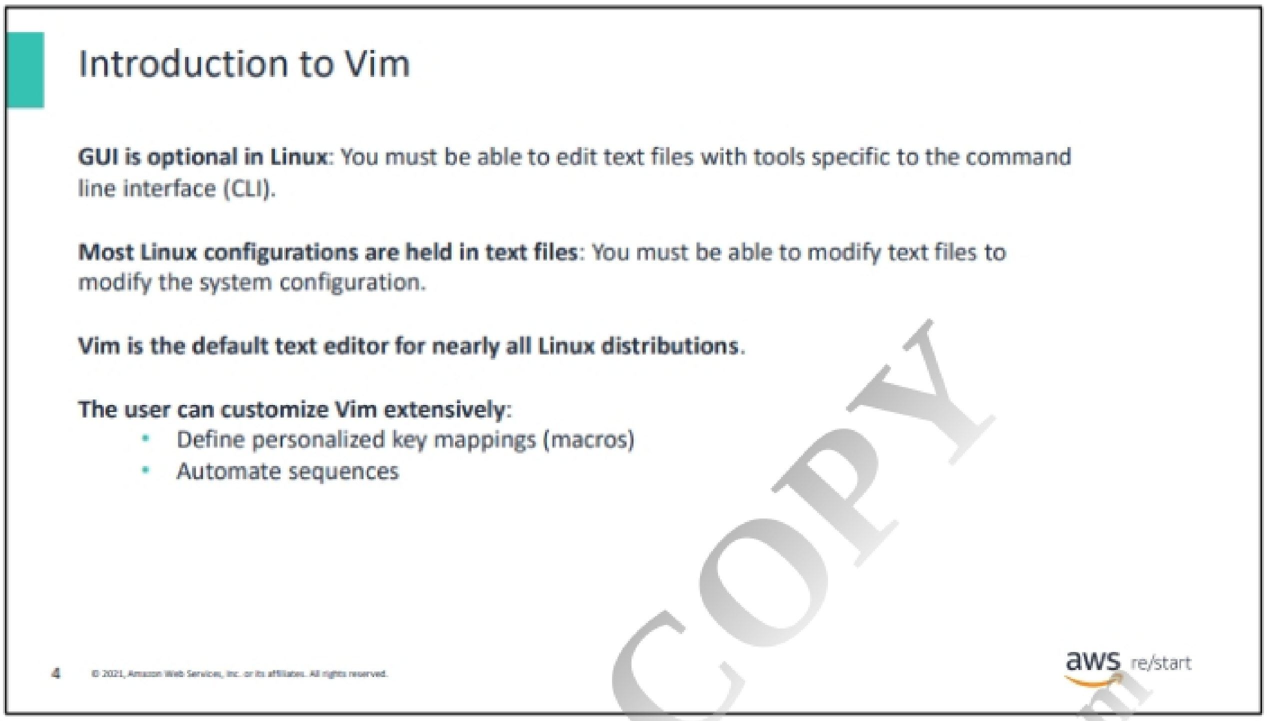
- Explain basic commands of the Vim file editor
- Explain basic commands of the GNU nano file editor
- Explain basic commands of the gedit file editor

In this lesson, you will learn how to:

- Explain basic commands of the Vim file editor
- Explain basic commands of the GNU nano file editor
- Explain basic commands of the gedit file editor

Vim text editor

This section introduces Vim and demonstrates some of the most widely used commands.



Introduction to Vim

GUI is optional in Linux: You must be able to edit text files with tools specific to the command line interface (CLI).

Most Linux configurations are held in text files: You must be able to modify text files to modify the system configuration.

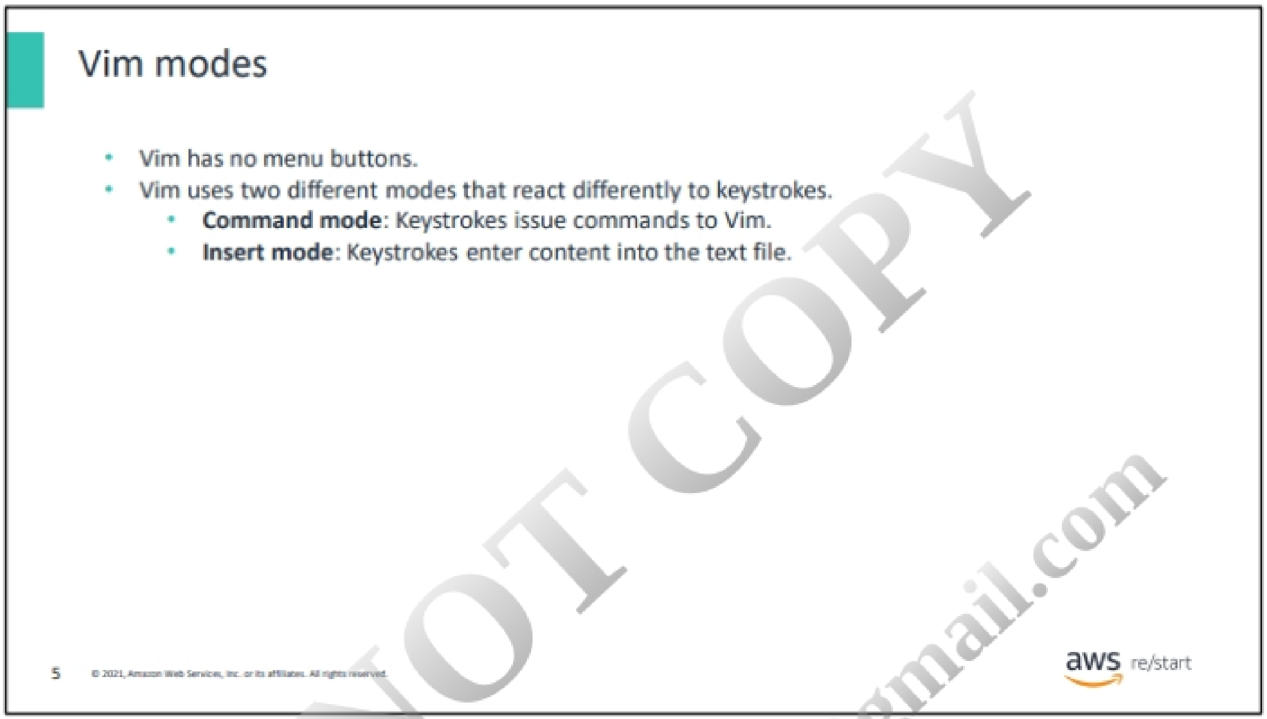
Vim is the default text editor for nearly all Linux distributions.

The user can customize Vim extensively:

- Define personalized key mappings (macros)
- Automate sequences

aws re/start

Vim is an implementation of Vi. Depending on the Linux distribution, you might find Vi or Vim. A basic understanding if this tool is essential.

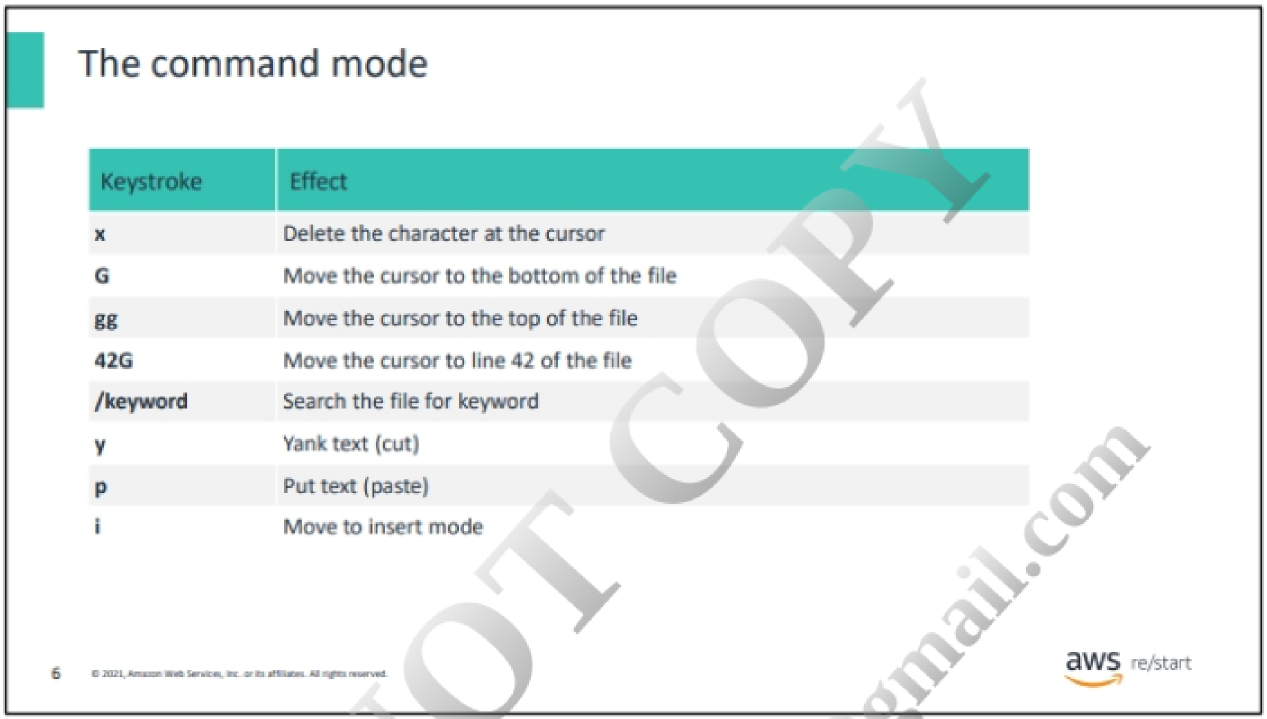


Vim modes

- Vim has no menu buttons.
- Vim uses two different modes that react differently to keystrokes.
  - **Command mode**: Keystrokes issue commands to Vim.
  - **Insert mode**: Keystrokes enter content into the text file.

You can switch among the three modes as needed.

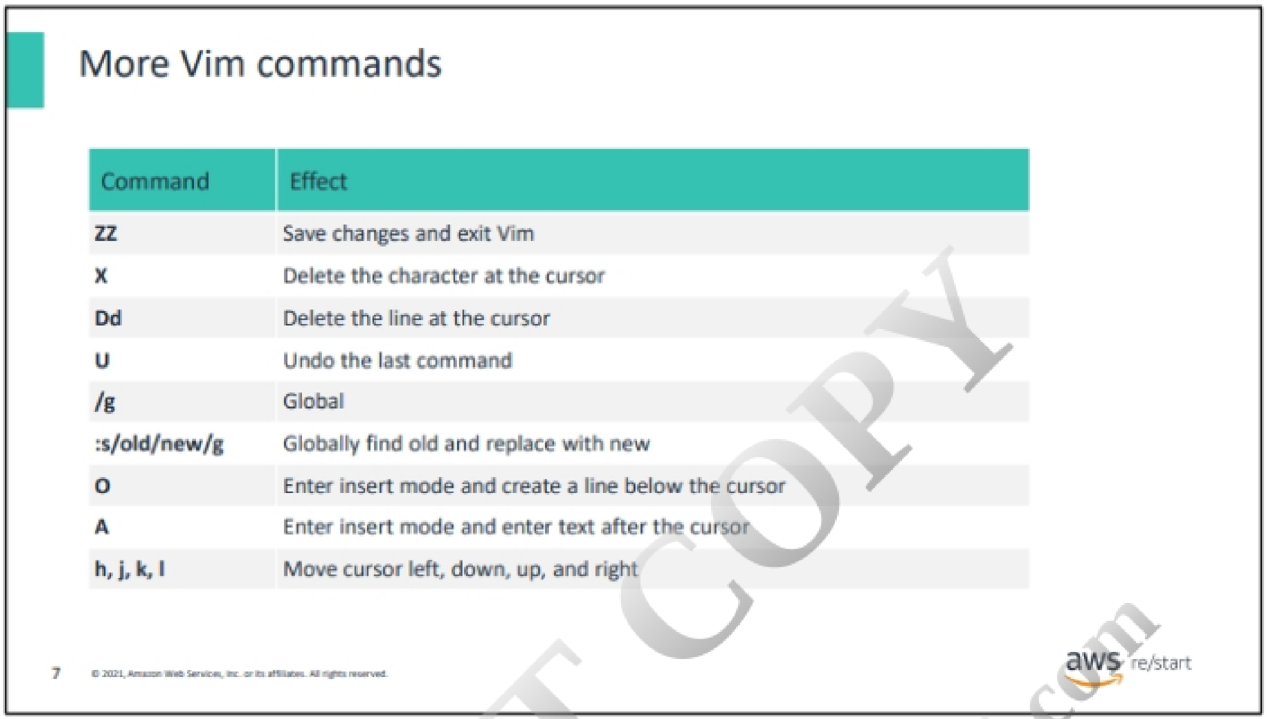
The next few slides will demonstrate some of the common commands and key strokes.



The command mode

Keystroke	Effect
x	Delete the character at the cursor
G	Move the cursor to the bottom of the file
gg	Move the cursor to the top of the file
42G	Move the cursor to line 42 of the file
/keyword	Search the file for keyword
y	Yank text (cut)
p	Put text (paste)
i	Move to insert mode

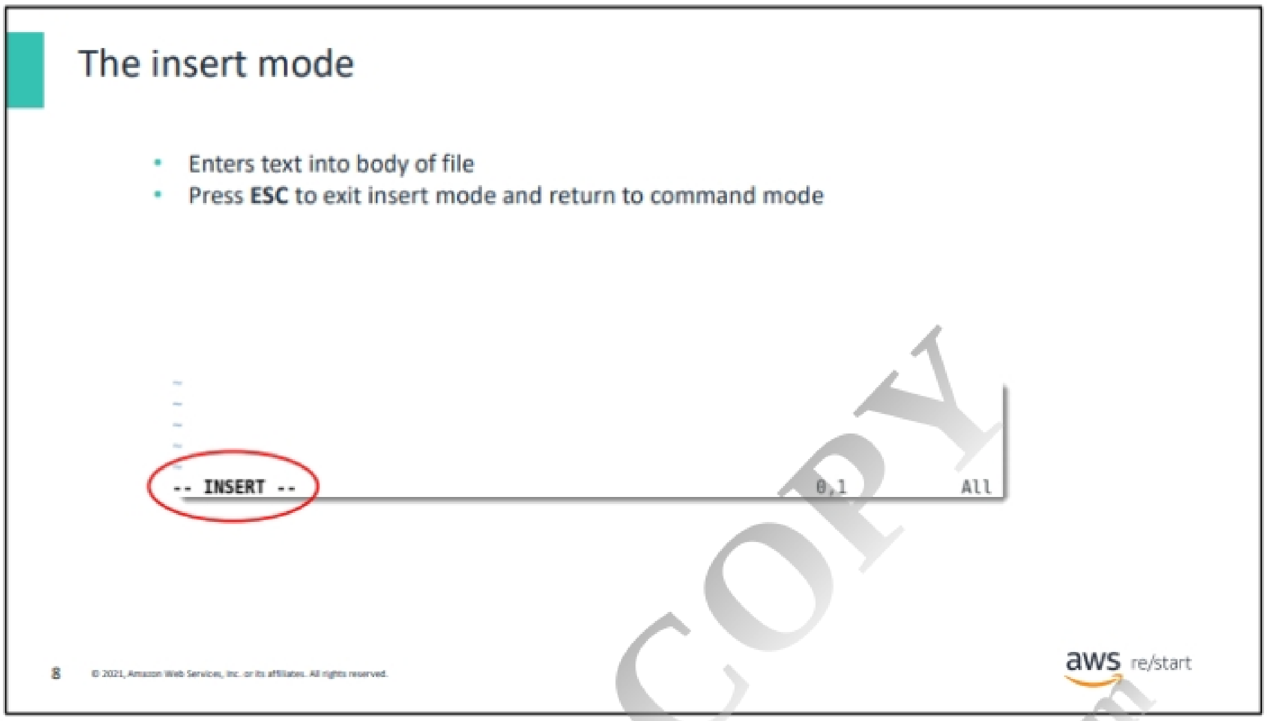
This list is not comprehensive.



More Vim commands

Command	Effect
ZZ	Save changes and exit Vim
X	Delete the character at the cursor
Dd	Delete the line at the cursor
U	Undo the last command
/g	Global
:s/old/new/g	Globally find old and replace with new
O	Enter insert mode and create a line below the cursor
A	Enter insert mode and enter text after the cursor
h, j, k, l	Move cursor left, down, up, and right

This list is not comprehensive.



The insert mode

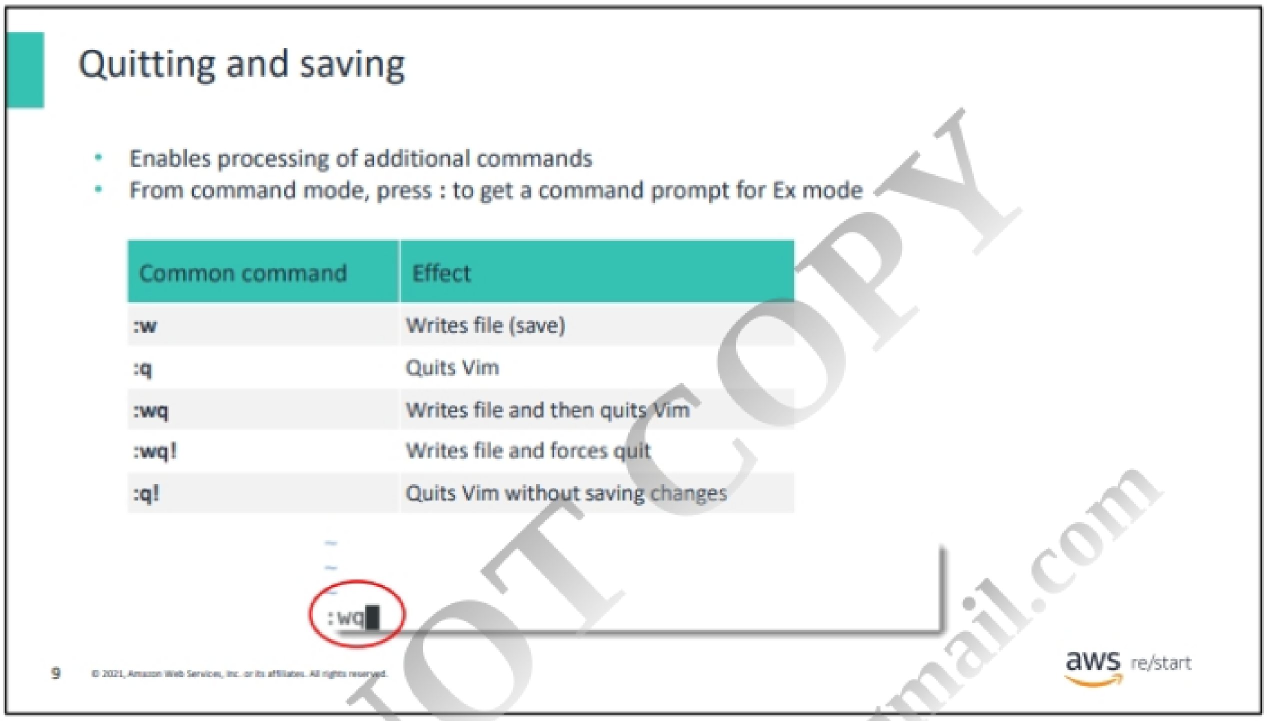
- Enters text into body of file
- Press **ESC** to exit insert mode and return to command mode

```
~
~
~
~
~
-- INSERT --                                                0,1           All
```

Enter *i.*

Enter your text.

Press ESC to exit the insert mode.



Quitting and saving

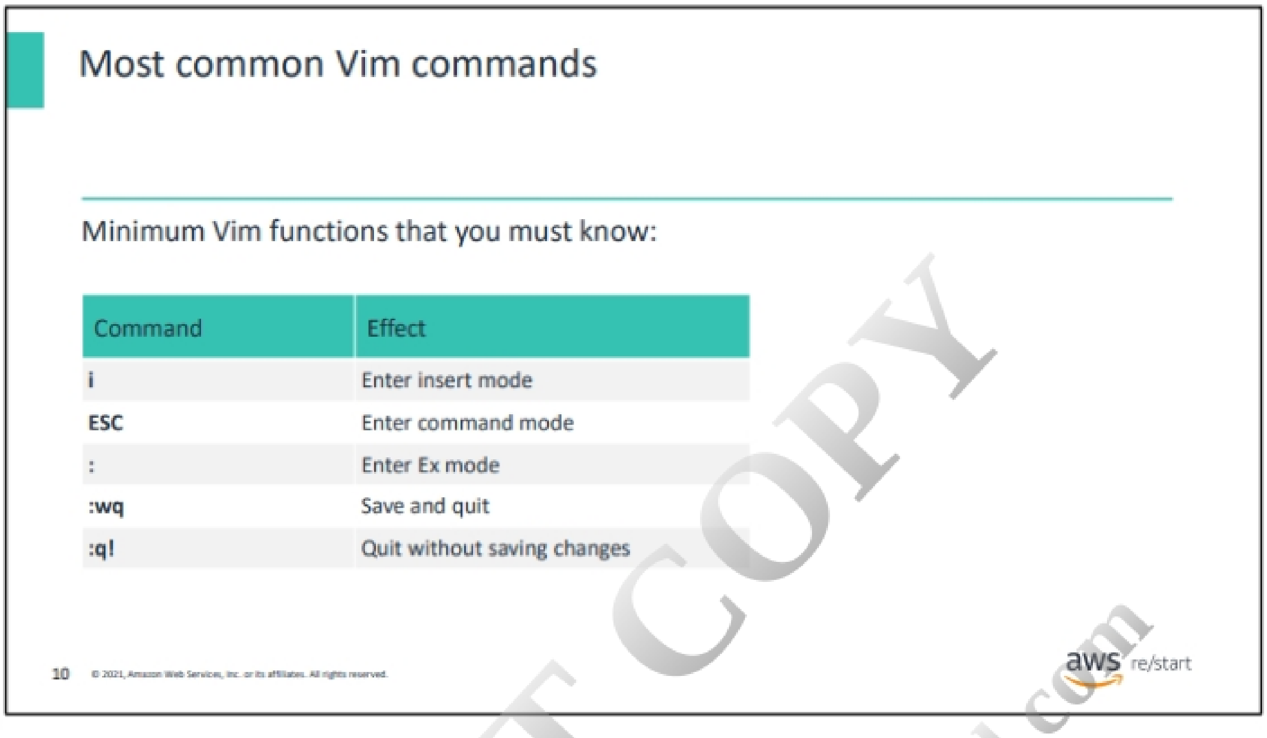
- Enables processing of additional commands
- From command mode, press : to get a command prompt for Ex mode

Common command	Effect
:w	Writes file (save)
:q	Quits Vim
:wq	Writes file and then quits Vim
:wq!	Writes file and forces quit
:q!	Quits Vim without saving changes

```
~
~
:wq
```

To save the file and exit VIM:

- Enter :
- Enter *wq*



Most common Vim commands

Minimum Vim functions that you must know:

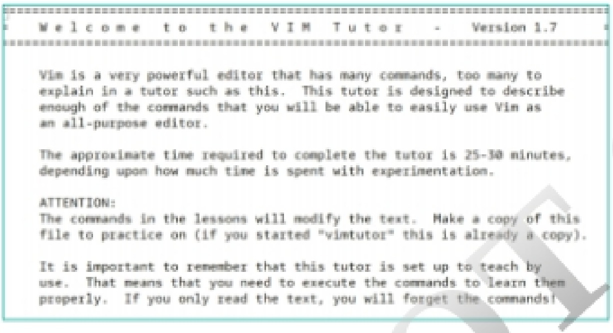
Command	Effect
i	Enter insert mode
ESC	Enter command mode
:	Enter Ex mode
:wq	Save and quit
:q!	Quit without saving changes

aws re/start

Be sure that you are comfortable with these commands, which you need to know to use Vim.

Get help in Vim

- `vimtutor`: A tutorial of common Vim tasks
- `:help`: Enter help
- `:help <keyword>`: Enter help for that keyword
- K: Open the man page for the word at the cursor



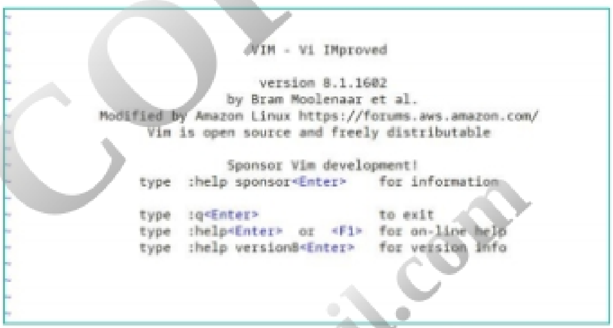
```
===============================================================================
=    W e l c o m e   t o   t h e   V I M   T u t o r    -    Version 1.7      =
===============================================================================

     Vim is a very powerful editor that has many commands, too many to
     explain in a tutor such as this.  This tutor is designed to describe
     enough of the commands that you will be able to easily use Vim as
     an all-purpose editor.

     The approximate time required to complete the tutor is 25-30 minutes,
     depending upon how much time is spent with experimentation.

     ATTENTION:
     The commands in the lessons will modify the text.  Make a copy of this
     file to practice on (if you started "vimtutor" this is already a copy).

     It is important to remember that this tutor is set up to teach by
     use.  That means that you need to execute the commands to learn them
     properly.  If you only read the text, you will forget the commands!
```



```
~
~
~                              VIM - Vi IMproved
~
~                               version 8.1.1602
~                           by Bram Moolenaar et al.
~           Modified by Amazon Linux https://forums.aws.amazon.com/
~                 Vim is open source and freely distributable
~
~                           Sponsor Vim development!
~                type  :help sponsor<Enter>    for information
~
~                type  :q<Enter>               to exit
~                type  :help<Enter>  or  <F1>  for on-line help
~                type  :help version8<Enter>   for version info
~
~
```

Vimtutor is a command to enter in the shell that opens a Vim documentation.

Other commands must be entered inside Vim, such as the following:

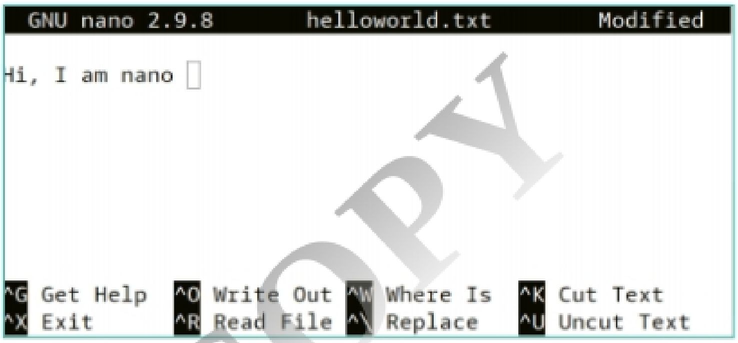
- Press ESC and enter *:help* to get general help, and then enter *:q* to exit the help page.
- Press ESC and enter *:help 'textwidth'* to go directly to the part of the documentation that mentions the word *textwidth*. Enter *:q* to exit the documentation.
- Enter *useradd*, press ESC, and enter *K* to get help about the `useradd` command. Then enter *q* to exit the help page.

GNU nano text editor

Nano is another lightweight text editor that works directly from the shell.

The GNU nano text editor

- Common text editor in Linux
- Not necessarily installed on every distribution



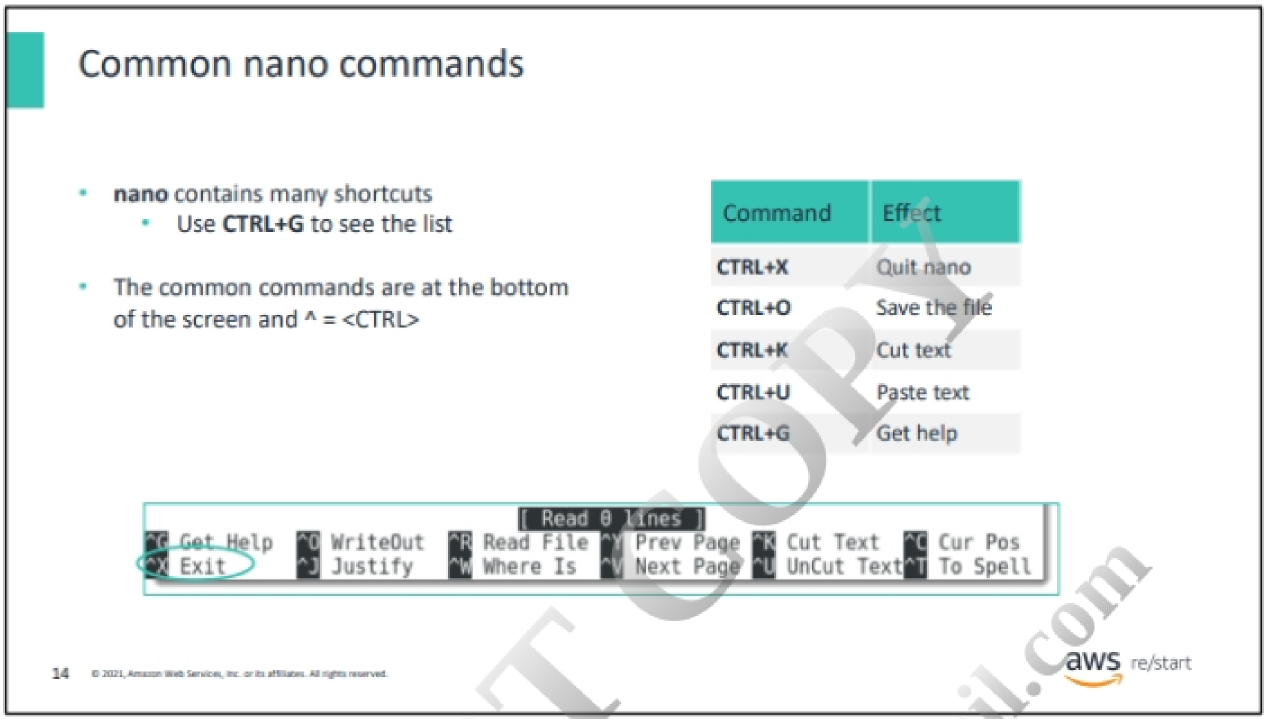
```
 GNU nano 2.9.8          helloworld.txt          Modified

Hi, I am nano

^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text
```

On a CentOS distribution, you can install nano by using the following command: `yum install nano`

On a Debian or Ubuntu distribution, you can use the following command: `sudo apt-get install nano`



Common nano commands

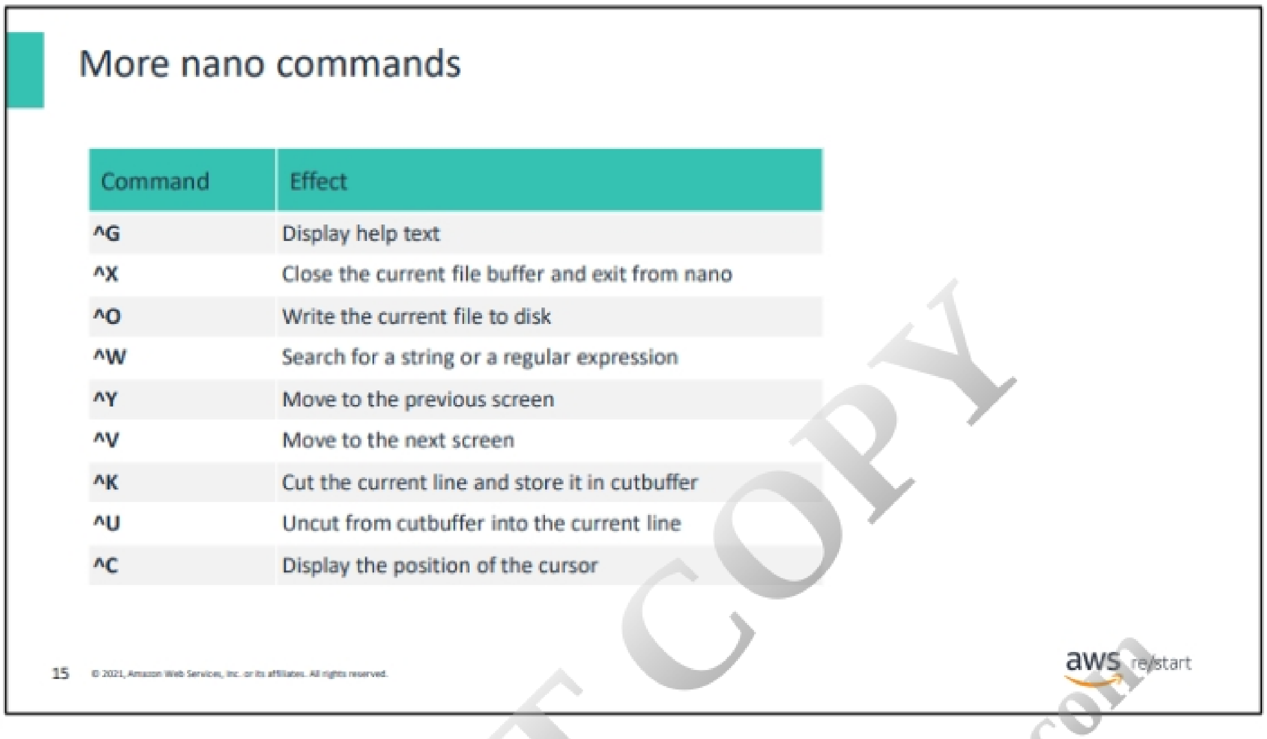
- **nano** contains many shortcuts
  - Use **CTRL+G** to see the list
- The common commands are at the bottom of the screen and ^ = <CTRL>

Command	Effect
CTRL+X	Quit nano
CTRL+O	Save the file
CTRL+K	Cut text
CTRL+U	Paste text
CTRL+G	Get help

```
                        [ Read 0 lines ]
^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

aws re/start

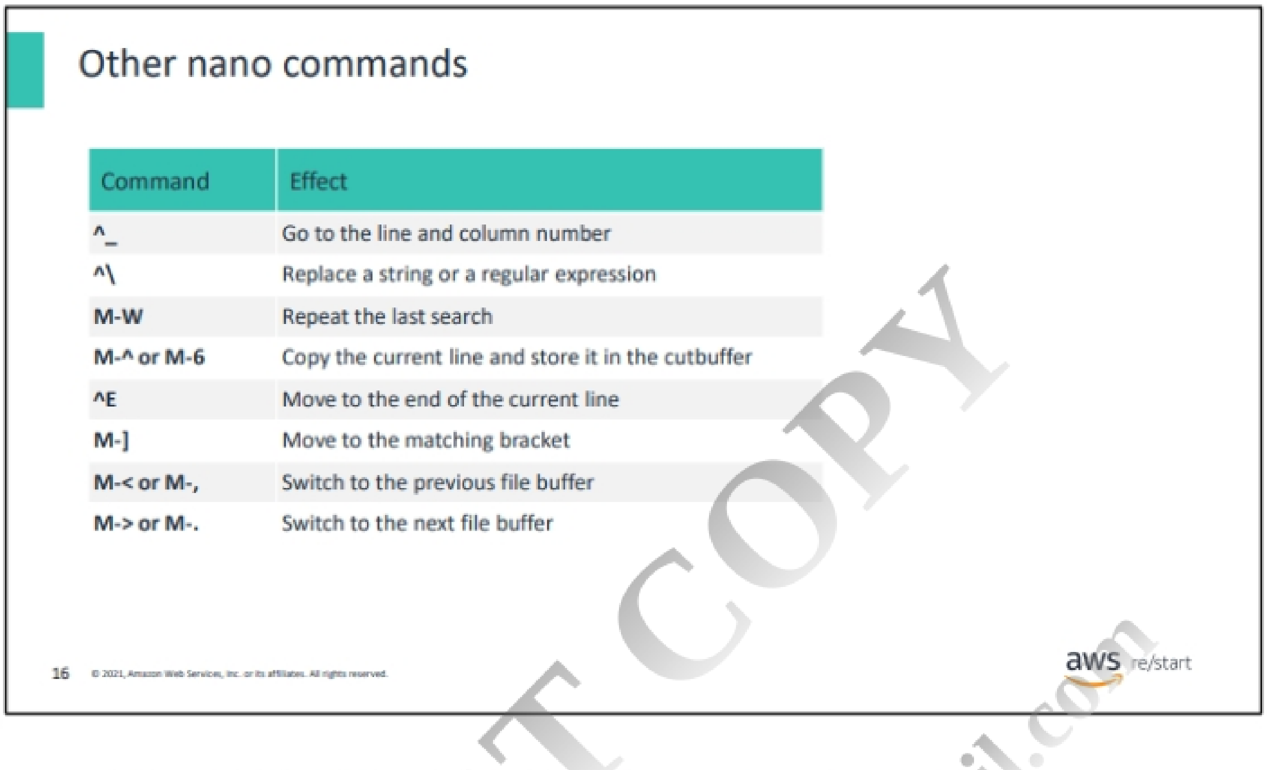
The next few slides demonstrate some common commands and key strokes.



More nano commands

Command	Effect
^G	Display help text
^X	Close the current file buffer and exit from nano
^O	Write the current file to disk
^W	Search for a string or a regular expression
^Y	Move to the previous screen
^V	Move to the next screen
^K	Cut the current line and store it in cutbuffer
^U	Uncut from cutbuffer into the current line
^C	Display the position of the cursor

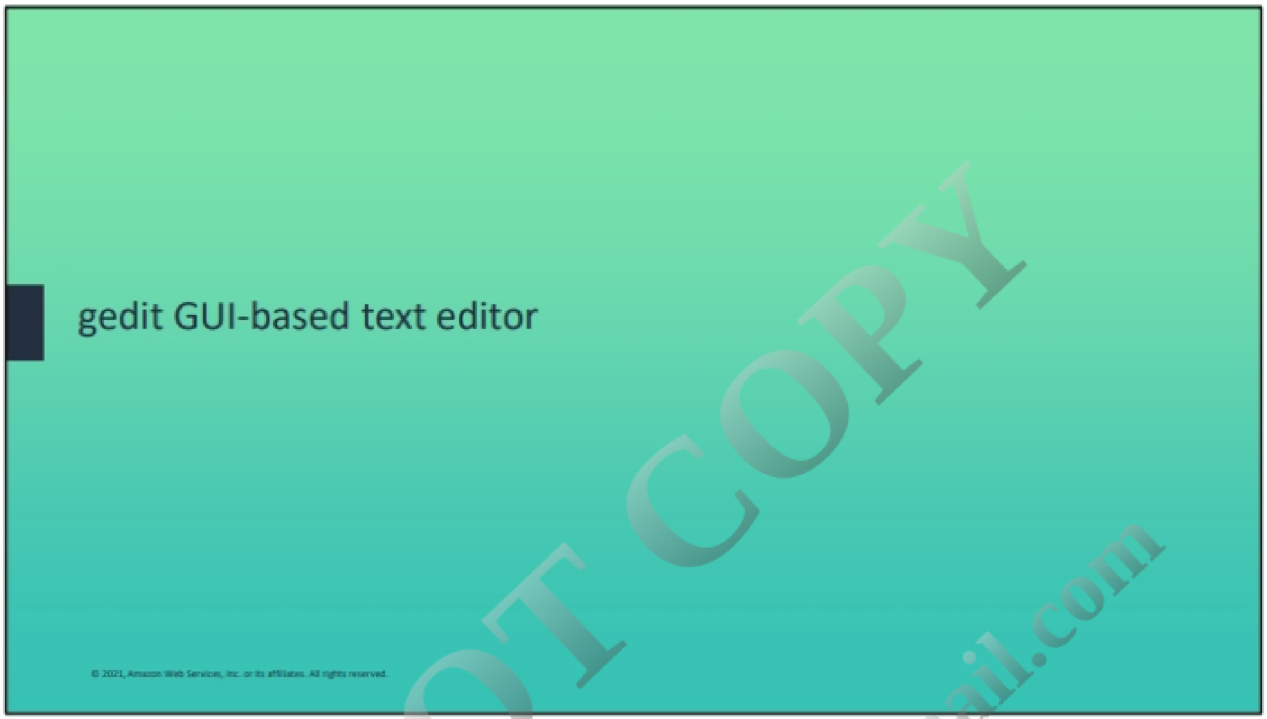
This list is not comprehensive.



Other nano commands

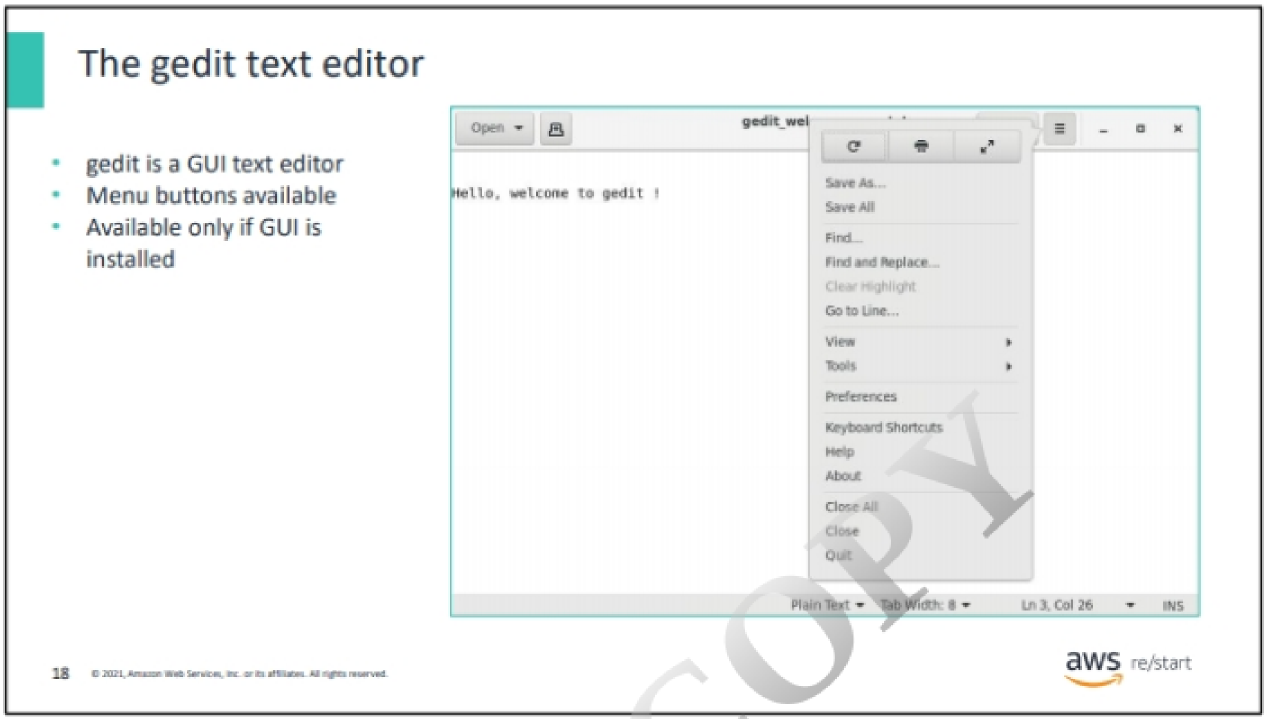
Command	Effect
^_	Go to the line and column number
^\	Replace a string or a regular expression
M-W	Repeat the last search
M-^ or M-6	Copy the current line and store it in the cutbuffer
^E	Move to the end of the current line
M-]	Move to the matching bracket
M-< or M-,	Switch to the previous file buffer
M-> or M-.	Switch to the next file buffer

This list is not comprehensive. Familiarize yourself with these commands.



gedit GUI-based text editor

Gedit is a graphic-based text editor. It requires a graphical user interface such as GNOME, Xfce, or K Desktop Environment (KDE) to be installed on the Linux distribution. It is optional.



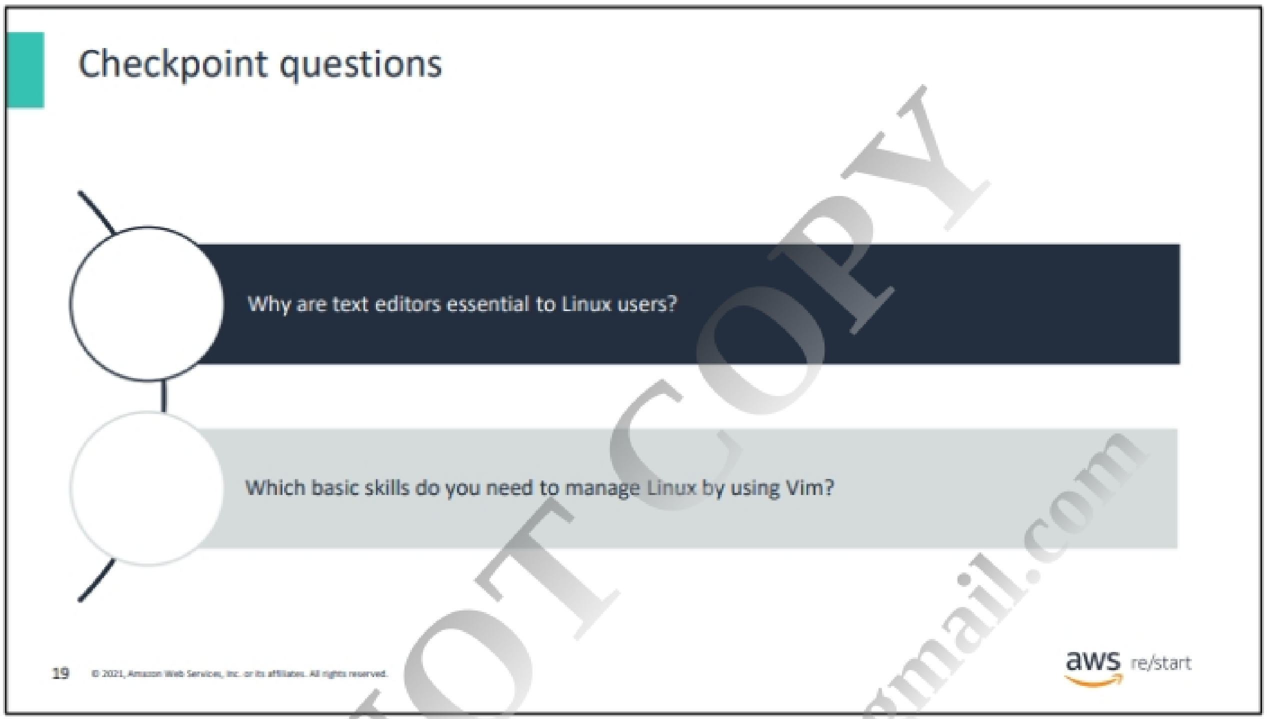
The gedit text editor

- gedit is a GUI text editor
- Menu buttons available
- Available only if GUI is installed

```
Hello, welcome to gedit !
```

AWS offers instructions that guide you through the installation of a graphical user interface (GUI) on an Amazon Elastic Compute Cloud (Amazon EC2) Linux 2 instance.

For more information about how to install a GUI on an EC2 instance running Amazon Linux 2, see https://aws.amazon.com/premiumsupport/knowledge-center/ec2-linux-2-install-gui/.



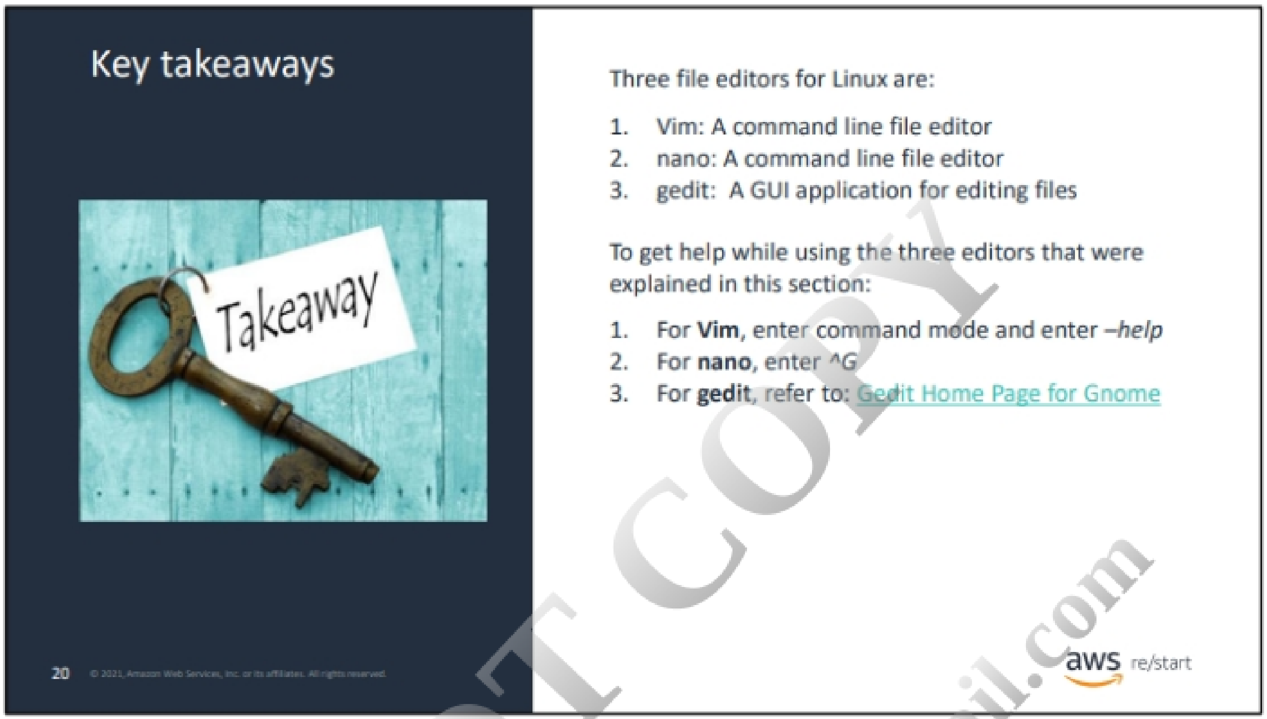
Checkpoint questions

Why are text editors essential to Linux users?

Which basic skills do you need to manage Linux by using Vim?

aws re/start

- Because the entire Linux file system is made up of files, being able to create and update files is essential.
- The basic skills are:
  a. Opening a file for editing (enter *vi <filename>*)
  b. Entering insert mode (press ESC and then enter *i*)
  c. Entering command mode (press ESC)
  d. Saving a file (enter command mode, enter *:w*, and then press Enter)
  e. Exiting vi (enter command mode, enter *:q*, and then press Enter)



Key takeaways

Three file editors for Linux are:

1. Vim: A command line file editor
2. nano: A command line file editor
3. gedit: A GUI application for editing files

To get help while using the three editors that were explained in this section:

1. For **Vim**, enter command mode and enter *–help*
2. For **nano**, enter *^G*
3. For **gedit**, refer to: [Gedit Home Page for Gnome]

20 © 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws re/start

Some key takeaways from this lesson include the following:

The following are the three file editors for Linux:

1. **Vim**: A command line (CLI) file editor
2. **nano**: A command line file editor
3. **gedit**: A GUI application for editing files

Help and further instructions are available for each.

For more information about gedit, see https://help.gnome.org/users/gedit/stable/.

Thank you