Creating New Commands
Creating New Commands

◆ You are not limited to AutoCAD's predefined commands.

◆ **Custom commands** can be
  ◆ created and then
  ◆ added to ribbon panels, menus, tool palette and toolbars.

◆ First, you must create the new command.
Creating New Commands

To create a custom command,

- Pick the **Create a new command button** in the Command List: pane of the Customize User Interface dialog box.
  - This button is to the right of the drop-down list.
- A new command is added to the list in the Command List: pane.
- Also the Button Image and Properties panes are displayed for the new command.
Customizing Interface Elements and Commands

- Pick to create a new command
- New command is added to the list
- Predefined button images
- Command name
- Description
- Command macro
- Command List:
  - Search command list
  - All Commands Only
- Command List:
  - Close Hatch Editor
  - Close Hole
  - Close Test Block Window
  - Collapse Face or Edge
  - Collect Horizontal
  - Collect Vertical
  - Color edges
  - Color faces
  - Color...:
  - Command Aliases Editor...
  - Command Line
  - Command1
  - Cone
  - Constraint Settings, Dimensional
  - Constraint Settings, Geometric
  - Constraint Settings...
General Command Properties

- **Naming:**
  - By default, the new command name is Commandn, where n is a sequential number.
  - Command1, Command2, etc
Give the command a descriptive name,

- Highlight the command in the Command List: pane.
- Pick in the Name property edit box in the Command category of the Properties Pane.
- Type the new name and press [Enter].

  - This property is displayed as the command name in the tooltip.
  - The name should be logical and short, such as Draw Box.
### Command Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Command1</td>
</tr>
<tr>
<td>Description</td>
<td>Description of New Command</td>
</tr>
<tr>
<td>Extended Help File</td>
<td></td>
</tr>
<tr>
<td>Command Display Name</td>
<td></td>
</tr>
<tr>
<td>Macro</td>
<td>^C^C</td>
</tr>
<tr>
<td>Tags</td>
<td></td>
</tr>
</tbody>
</table>

### Advanced Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element ID</td>
<td>MMU_0032</td>
</tr>
</tbody>
</table>

### Images

<table>
<thead>
<tr>
<th>Image Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td></td>
</tr>
</tbody>
</table>
General Command Properties

- The entry in the Command Display Name property
  - is what appears in the command-line section of the tooltip.
### Properties

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Command Display Name</th>
<th>Macro</th>
<th>Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Command1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Description of New Command</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extended Help File</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command Display Name</td>
<td>Command Display Name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Macro</td>
<td>^C^C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tags</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Advanced

<table>
<thead>
<tr>
<th>Element ID</th>
<th>MMU_0032</th>
</tr>
</thead>
</table>

### Images

<table>
<thead>
<tr>
<th>Small image</th>
<th>Large image</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
General Command Properties

◆ The text that appears in the Description property text box
  ◆ appears in the tooltip when the cursor is over the button.

◆ This text is called the help string
  ◆ Should also be logical, but can be longer and more descriptive than the command name.
Customizing Interface Elements and Commands
General Command Properties

- The **Extended Help File property** is used to specify an Extensible Application Markup Language (XAML) **file** to use as extended help.
  - The extended help is displayed in the tooltip when the cursor is paused over a tool for a longer period of time.
  - By default, if you pause the cursor for two seconds, the extended help is displayed (if the tool contains extended help).
### Command

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Command1</td>
</tr>
<tr>
<td>Description</td>
<td>Input help string here</td>
</tr>
<tr>
<td>Command Display Name</td>
<td></td>
</tr>
<tr>
<td>Macro</td>
<td>^C^C</td>
</tr>
<tr>
<td>Tags</td>
<td></td>
</tr>
</tbody>
</table>

### Advanced

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element ID</td>
<td>MMU_0032</td>
</tr>
</tbody>
</table>

### Images

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small image</td>
<td></td>
</tr>
<tr>
<td>Large image</td>
<td></td>
</tr>
</tbody>
</table>
To assign an XAML file,

- Select the property and
- Pick the ellipsis button (…) on the right-hand side of the text box.
- Then, locate and open the file.

For information on creating XAML files, search the Internet for resources.

Many resources can be found on the Microsoft website
Example

Create a command that draws a rectangular border for an ANSI E-size sheet (44" x 34) using a wide polyline, sets the drawing limits, and finishes with ZOOM Extents.
Creating a New Command

◆ Create a new command that draws an E-size border, sets limits and zooms extents.

◆ Create a new command and

◆ Command name
  ◆ “E-Border”

◆ Description property
  ◆ “Draws E –size borders, sets limits, and zooms extents”

◆ Command Display Name property,
  ◆ Enter Draw Borders.
### Command Properties

#### Command
- **Name**: E-Border
- **Description**: Draws E-size border, sets limits, etc.
- **Command Display Name**: Draws Border
- **Macro**: °C

#### Advanced
- **Element ID**: MMU_0032

#### Images
- **Small image**
- **Large image**
**Button Image**

- The Button Image pane in the Customize User Interface dialog box is used to define the image that appears on the command button.

- The image should graphically represent the function of the command.
  - AutoCAD provides several predefined images.
  - One of these can be selected as the button image.
You can also right-click on the list of images and
Select Import Image... from the shortcut menu to import an image.
Button Image

The CUI Editor-Image Manager dialog box can be used to control and store custom images in a CUIx file.
Customizing Interface Elements and Commands
Customizing Interface Elements and Commands
To display this dialog box

- pick the Image Manager... button
- Located to the right of the drop-down list at the top of the Customizations in All Files pane.
- Any image stored in a loaded CUIx file is available in the list of predefined images.
Customizing Interface Elements and Commands
**Button Image**

- **Two types of images**
  - Large button image
  - Small button image
- Can be different or the same image.
- It may be a good idea for a button to have a separate image for each of the two button sizes.

*If you use the same image for both small and large buttons, the image is appropriately scaled as needed*
Button Image

- Pick the appropriate radio button
- Select an image.
  - The name of the image appears in the Images category in the Properties pane.
  - The small image also appears next to the command name in the Command List pane.
However, confusion may arise if your custom command has the same button image as an existing AutoCAD command. It is best to create custom button images for use with your custom commands.
Button Image

- The CUI Editor-Image Manager dialog box makes it easy to manage the button images.

- You can either
  - Modify an existing button image or
  - Create a new image from scratch.

- In either case, a predefined image must be selected from the list of existing images.
Button Image

- Then, pick the Edit... button in the Button Image pane to open the Button Editor dialog box.
Creating a Custom Button Image

- The Button Editor dialog box has
  - Basic "pixel-painting" tools and
  - Several features to simplify the editing process.
Creating a Custom Button Image

- The four tools are shown as buttons at the top of the dialog box.
- The pencil paints individual pixels.
- The line tool allows you to draw a line between two points.
- The circle tool allows you to draw center/radius style ellipses and circles.
- The erase tool clears the color from individual pixels.
Creating a Custom Button Image

- The **current color** is selected from the color palette on the left-hand side of the dialog box and indicated by a depressed color button.
  - Anything you draw appears in the current color.

- A **preview** of the button image
  - appears to the right of the tools.
Customizing Interface Elements and Commands
Creating a Custom Button Image

◆ Draw button image with the grid turned on.
  ◆ The grid provides outlines for each pixel in the graphic.
  ◆ Each square represents one pixel.
  ◆ Picking the Grid check box toggles the state of the grid.
Creating a Custom Button Image

- **Button sizes**
  - **Small**
    - Default
    - A drawing area of 16 pixels x 16 pixels.
  - **Large**
    - If Use large buttons for Toolbars is turned on
      - Then the button image drawing area is 32 pixels x 32 pixels.
Creating a Custom Button Image

- The images in the CUI db are displayed at the current size setting (small or large).
Creating a Custom Button Image

- Other tools in the Button Editor dialog box.
  - Clear.
    - If you want to erase everything and start over, pick the Clear button to clear the drawing area.

- Use this button to clear the existing image and start a button image from scratch.
Creating a Custom Button Image

- Other tools in the Button Editor dialog box.
  - **Undo**.
    - You can undo the last operation by picking this button.
    - Only the last operation can be undone.
    - An operation that has been undone cannot be redone.
Creating a Custom Button Image

- Other tools in the Button Editor dialog box.
  - **Save.**
    - Names the current button image and saves it to the current CUIx file.
  - **Import.**
    - Use this button to open an existing bitmap (BMP) file that does not appear in the Button Image pane of the Customize User Interface dialog box.
    - The image is automatically resized to fit the current button size.
Creating a Custom Button Image

- Other tools in the Button Editor dialog box.
  - Export.
    - This button saves a file using a standard save dialog box.
  - Close.
    - Ends the Button Editor session.
    - A message is displayed if you have unsaved changes.
  - Help.
    - Provides context-sensitive help.
Customizing Interface Elements and Commands
Creating a Custom Button Image

- Other tools in the Button Editor dialog box.
- **More**
  - Opens the standard Select Color dialog box.
  - Allows you to use colors in the button other than those in the default color palette.
Creating a Custom Button Image

Once a button image is saved, it appears in the list of predefined images in the Button Image pane of the CUI db.

All images saved for use as button images must be stored where AutoCAD will find them.
Creating a Custom Button Image

- AutoCAD provides the `\icons` folder within the user's support file search path.
  - This is the default folder when using the Export... button in the Button Editor dialog box.
  - If you choose to use a different folder, it must be added to the support file search path, which is specified in the Files tab of the Options dialog box.
Creating a Custom Button Image

Example

Rather than using an existing button image for the E-Border command, an entirely new button image will be created.

With E-Border highlighted in the Command List: pane, select any one of the images in the Button Image pane and pick the Edit... button.
Creating a Custom Button Image

Example

- The Button Editor dialog box is displayed.
- Select the Clear button to completely remove the existing image.

The next slide

- Shows a 16 x 16 pixel image created for the E-Border button with the Grid option activated.
- Use the pencil and line tools to create this or a similar image.
Created image
Creating a Custom Button Image

- **Example**
  - Using the **Save...** button saves our image with a name of E-border
    - It will be stored in the current CUIx.
  - Pick the **Close** button to return to the CUI db.
  - Your newly created image now appears in the list of existing images in the Button Image Pane as shown in the next slide
    - It is automatically associated with the command.
Associating a Custom Image with a Command

Two ways to associate a new custom button image with a command.

1. Use the **Button Image pane** or
2. The **Properties pane** in the Customize User Interface-dialog box.

Once a button image is associated with a command, the image is used for that command on all ribbon panels, menus, and toolbars where the command is inserted.
To use the Button Image pane to assign an image to a command:

- Make sure the command is selected in the Command List: pane.
- Select the Large image, Small image, or Both radio button in the Button Image pane to determine for which size of button the image will be used.
- Pick the button image in the list of predefined button images.
- Pick the **Apply** button.
You can also use the Properties pane to associate a saved button image file with the command.

- Select the command in the Command List: pane.
- In the Properties pane, expand the Images category to display the Smali image and Large image properties.
Associating a Custom Image with a Command

- If there is an image currently associated with the property, the path to the image is displayed.
- If there is no path displayed, the image is saved in a CUIx file.
- Pick in each property textbox and type the path and file names of the saved image files.
Associating a Custom Image with a Command

- Alternately, you can pick the ellipsis button (...) to display a standard open dialog box and locate the file.
  - This button appears when the property is selected.
- Finally, pick the Apply button to assign the image(s) to the button.
Associating a Custom Image with a Command

- If you designate an image file only for small buttons
  - the button for the command will be blank when you switch to large buttons.

- Be sure to specify an image for both small and large buttons!
**Associating a Custom Image with a Command**

- **RIBBONICONRESIZTE**
  - System variable that controls the resizing of ribbon images to standard sizes of 16 x 16 (small) and 32 x 32 (large) icons.
  - Has an integer value of either 0 (off) or 1 (on).
  - By default this system variable is set to 1.
    - Which means that icons are resized.
    - This system variable is also controlled by the Resize ribbon icons to standard sizes option in the Display tab of the Options dialog box.
Defining a Custom Command
Defining a Custom Command

- Now, you need to define the action that the custom command will perform.
- A text string called a macro defines the action performed by the command.
  - This text string appears in the Macro property text box in the Command category in the Properties pane of the Customize User Interface dialog box.
Defining a Custom Command

- In many cases, this ‘Command’ is actually a macro that invokes more than one command.
- **Start the macro with the text \(^C^C\)**
  - \(^C\) is a cancel command.
    - This is the same as pressing the [Esc] key.
    - The default text, then, represents two cancels.
Whenever a command is not required to operate transparently, it is best to begin the macro with two cancel keystrokes (^C^C) to fully exit any current command and return to the Command: prompt.

Typically, two cancels are required to be sure you begin at the Command: prompt.

One cancel may not completely exit some commands or functions.
For example, when grips are active, one cancel deactivates grips, but a second cancel is required to fully exit the command and return to the Command: prompt.
Defining a Custom Command

- The macro must perfectly match the requirements of the activated commands.
  - For example, if the LINE command is issued, the subsequent prompt expects a coordinate point to be entered.
    - Any other data input is inappropriate and will cause an error in the macro.
    - It is best to manually "walk through the desired macro, writing down each step and the data required by each prompt."
Defining a Custom Command

- The following command sequence creates the rectangular polyline border with a .015 line width.
- Absolute coordinates are used, not relative coordinates.
- Command: PLINE
- Specify start point: 1, 1.
- Current line-width is 0.0000
- Specify next point or [Arc/Halfwidth/Length/Undo/Width]: W
- Specify starting width< 0.0000>: 0.15
- Specify ending width< 0.0150>: 
- Specify next point or [Arc/Halfwidth/Length/Undo/Width]: 42, 1
- Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: 42, 32
- Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: 1, 32
- Specify next point or [Arc/Close/Halfwidth/Length/Undo/Width]: C
- Command:
Defining a Custom Command

- Creating the macro for your custom E-Border command involves duplicating these keystrokes with a couple of differences.
- Some symbols are used in menu macros to represent keystrokes.
Defining a Custom Command

- For example, a cancel (^C) is not entered by pressing [Esc].
- Instead the [Shift]+[6] key combination is used to place the caret symbol, which is used to represent the Ctrl key in combination with the subsequent character (a C in this case)
Defining a Custom Command

- Another keystroke represented by a symbol is the [Enter] key.
- An [Enter] = semicolon
- A space can also be used to designate [Enter]
  - However, the semicolon is more commonly used because it is very easy to count to make sure that the correct number of "enters" is supplied.
Defining a Custom Command

- AutoCAD system variables and control characters can be used in menus.
- They can be included to increase the speed and usefulness of your menu commands.
- Become familiar with these variables so you can make use of them in your menus.
Defining a Custom Command

- ^B. Snap mode toggle.
- ^C. Cancel.
- ^D. Dynamic UCS toggle.
- ^E. Cycles to next isoplane.
- ^G. Grid mode toggle.
- ^H. Issues a backspace.
- ^1. Issues a tab.
- ^M. Issues a return.
- ^O. Ortho mode toggle.
- ^P. MENUECHO system variable toggle.
Defining a Custom Command

- ^Q. Toggles echoing of prompts, status listings, and input to the printer.
- ^R. Toggles command versioning. Allows macros written in previous releases to properly function.
- ^T. Tablet toggle.
- ^V. Switches current viewport.
- ^Z. Suppresses the addition of the automatic [Enter] at the end of a command macro.
- \. Pauses for user input.
Defining a Custom Command

- Macro draws the border
  - `^C^CPLINE ;1,1;W ;.015; ;42,1;42,32;1,32;C;`

- Macro sets the limits
  - `LIMITS;0,0; 44,34;ZOOM;E`

- An "enter" is automatically issued at the end of the macro so it is not necessary to place a semicolon at the end.

- The macro for the custom command is now complete
Defining a Custom Command

To assign the macro to your custom E-Border command

- Select the command in the Command List: pane of the Customize User Interface dialog box.
- Pick Macro property text box in the Properties pane and enter the complete macro shown above.
Defining a Custom Command

- For a long macro such as this one, you can pick the ellipsis button (...) at the end of the text box to display the Long String Editor dialog box.
- Enter the macro in this dialog box and pick the OK button to return to the Customize User Interface dialog box.
- Pick the Apply button to associate the macro with the custom command.
Long String Editor

Type the macro
Overview of the Ribbon
Overview of the Ribbon

- By default the **ribbon** is docked at the top of the drawing area.
  - The **ribbon** can also be
    - Docked to the left or right of the drawing area, or
    - It can be **floating**.
- The **ribbon** contains commands and tools on **panels**.
Overview of the Ribbon

◆ The panels are grouped on tabs that can be individually displayed.
  ◆ Think of the tabs as the containers that hold the ribbon panels.

◆ Together, the panels and tabs make up the ribbon.
Overview of the Ribbon

- You may quickly change the display from the full ribbon view to one of the minimized options by double-clicking any ribbon tab title.

- By default, the ribbon is set to cycle through all views.
Overview of the Ribbon

- To the right of the last tab name is a **button with an arrow icon** that is used to set the appearance of the docked ribbon and its cycling.
Overview of the Ribbon

◆ The three options to set the view of tabs are
  ◆ Minimize to tabs.
  ◆ Minimize to panel titles.
  ◆ Minimize to panel buttons.
Overview of the Ribbon

- When minimized to panel titles or buttons,
  - Hover the cursor over the minimized icon to display the corresponding panel.

- When minimized to tabs
  - Pick the tab name to display the tab.
A ribbon panel may contain rows of:
- Command buttons
- Drop-down lists, or
- Sliders.
Overview of the Ribbon

-you can choose which panels are visible by right-clicking on the ribbon to display a shortcut menu.

- Select either Show Tabs or Show Panels and choose which content to display.

- The currently displayed items are checked in the submenus.
Overview of the Ribbon

Customizing Interface Elements and Commands
Overview of the Ribbon

Customizing Interface Elements and Commands
Overview of the Ribbon

Figure 22-11.
A—The **Show Tabs** submenu is used to choose which tabs are displayed in the ribbon.
B—The **Show Panels** submenu is used to choose which panels are displayed in the current tab.
Overview of the Ribbon

- The items displayed in the Show Panels submenu are based on which tab is current (on top).
  - There are separate panels for each tab.
  - Workspaces are typically used to set which ribbon components are displayed.
Overview of the Ribbon

- Also the ribbon can dynamically change when commands are accessed.
- Contextual tabs may be displayed on the ribbon when a command is active and then hidden when the command is finished.
Tabs and Panels

- The ribbon has three customization branches in the CUI dialog box.
  - Tabs branch
  - Panels branch, and
  - Contextual Tab States branch.
    - These branches are located below the Ribbon branch.
Tabs and Panels

The Tabs contain the panels and
They are displayed first in the tree.
Tabs and Panels

- In the CIAF pane, expand the Tabs branch
  - There are 18 default tabs associated with the main user interface.
  - These appear at the top of the branch.
  - The remaining tabs have ‘contextual” in their name.
    - These are used to reference the contextual tab states.
Some of the main ribbon tab names have the suffix 2D or 3D.
  - These help identify the types of commands contained on the tab.

They are included in the appropriate workspace
  - 2D Drafting & Annotation
  - 3D Modeling
  - 3D Basics
Example: select the Home - 3D branch. Notice that the name of this branch does not match the name displayed on the AutoCAD screen.

With the branch selected look at the Properties pane.

The name displayed on the AutoCAD screen is the value in the Display Text property.
Customizing Interface Elements and Commands
Expand the Home - 3D branch.

It contains 11 branches:

- Eight with a prefix of Home 3D
- Two with a prefix of Home, and
- One with a prefix of View.

These branches correspond to the panels associated with the Home tab in the 3D Modeling workspace.

Notice that there are no branches below these.
Tabs and Panels

Customizing Interface Elements and Commands
Tabs and Panels

- The Tabs branch contains
  - Branches for *tabs* and
  - Branches for *panels*.

- The Panels branch contains
  - Branches for *panels* and
  - Branches for the *commands on each panel*.
Customizing Interface Elements and Commands
Tabs and Panels

- Expand the Panels branch.
  - All of the available panels are shown as branches below the Panels branch.

- Expand the Home 3D – Modeling branch.
  - It consists of the Panel Dialog Box Launcher branch,
  - Two rows and
  - a Slideout branch.
Any row listed below the Slideout branch is only visible when the ribbon panel is expanded.

For the Modeling panel in the ribbon, row 2 is located in the expanded portion of the panel.

You can expand the branches for the rows in a tab.
Tabs and Panels

Customizing Interface Elements and Commands
Next expand the branch for the subpanel (Sub-Panel1).

This subpanel contains two rows.

Expanding the branch for each row displays the commands contained in it.

Notice how the drop-down list and subpanel contained in row 1 of the Home 3D – Modeling branch are fitted together in the Modeling panel in the ribbon.
Tabs and Panels

Expand the branch for row 2 in the Home 3 D – Modeling branch.

- This branch contains commands, *but no drop-down lists or subpanels.*

- Notice that row 2 is below the Slideout branch which is the panel separator.

- This means it is displayed in the expanded panel.
Tabs and Panels

ٍ The final branch in the Ribbon branch is Contextual Tab States.

ٍ When you expand this branch, several branches of AutoCAD commands and features are displayed.

ٍ Some branches have contextual tab panels assigned to them and others do not.
Example:

- Expand the Text Editor in progress branch.
- Below it is the Text Editor Contextual Tab branch.
- This indicates the Text Editor contextual tab will be displayed when text is being created or edited in the drawing.
**Tabs and Panels**

◆ **Example**
  - Launch the MTEXT command to create sample multiline text.
  - While the command is active, the Text Editor tab is displayed in the ribbon and made active.
  - When the command is complete, the tab is automatically removed from the ribbon.
Customizing Interface Elements and Commands
Tabs and Panels

- You can associate tabs for any command or feature listed in the Contextual Tab States branch in the Customize User Interface dialog box.

- To do so,
  - Drag the tab from the Tabs branch and
  - Drop it in the desired branch in the Contextual Tab States branch.
Button Properties

- In the ribbon, **command buttons** have additional properties from the basic command properties.
  - These properties are located in the Appearance section of the Properties pane.
Button Properties

- Buttons may be displayed in one of five default sizes:
  - Large with text (vertical)
  - Large with text (horizontal)
  - Large without text
  - Small with text, and
  - Small without text.
Button Properties

- The size
  - is set in the customize User Interface dialog box.
  - The size setting is actually the maximum display size for the button.
  - When the subpanel is set to do so, AutoCAD adjusts the button size smaller as needed based on the space available for the ribbon.
Button Properties

To set the size of a command button,

- Expand the Ribbon branch and the Panels branch.
- Expand the branches for the panel and row that contain the command.
- Select the command in the Customizations in All Files Pane.
- Set the Button Style property in the Appearance section of the Properties pane.
Button Properties

- Generally, text labels are not shown when small buttons are specified.
- When large buttons are specified
  - The vertical orientation is usually selected so the text is below the button.
  - This helps reduce the width of the panel to preserve space on the ribbon.
To add a command to a panel

- Open the CUI dialog box.
- Expand the Ribbon branch
- Expand the Panels branch.
- Expand the branches for the row to which the command will be added.
- In the Command List: pane, locate the command to add to the panel.
- Drag the command from the Command List: pane and drop it into position in the tree in the Customizations in All Files pane.
Customizing A Panel

To remove a command from a panel,
- right-click on the command in the panel’s branch in the Customizations in All Files pane and
- Select Remove from the shortcut menu.
- You can also select the command and press the [Delete] key.
Customizing A Panel

- Row 1 is the top of the panel
  - The top of a row branch is the left-hand side of the panel.
  - Commands are displayed in this order on the panel
  - Commands can be rearranged.
Customizing A Panel

- The Customizations in All Files pane
  - Select the command to move
  - Drag it to a new location

- Rows can be rearranged
  - Drag them within the tree.
  - After you drag a row to a new location, all rows are automatically renumbered.
Customizing A Panel

- Row at the top of the panel branch
  - Is always row 1
  - All other rows are sequentially numbered

- The panel separator (Slideout branch)
  - Can be dragged to a new location
  - Rows listed after the panel separator are not displayed until the panel is expanded
Customizing A Panel
Customizing A Panel

- A new row can be added to a panel in the Customizations in All Files pane
  - Right-click the row after which you would like the new row added.

- To add a new first row
  - Right-click on the panel branch name.
  - Select New Row from a shortcut menu
    - New row is added
    - All other rows are renumbered
Adding Drop-Down List to Ribbon Panel

- A drop-down list can be added to a row in a ribbon panel.
- Use the CUI dialog box
- Right-click on the row branch
- Select New Drop-down from the shortcut menu.
  - A branch is added to the bottom of the row’s branch
Creating a New Tab or Panel

- Open the CUI dialog box
- In the CIAF pane, right-click on the Panels branch.
- Select New Panel from shortcut menu.
  - A new panel is added
  - Default name: Panels
- Enter a name for new panel.
  - The Dialog Box launcher and
  - Row 1 and
  - Slide out branches are automatically added.
Creating a New Tab or Panel

- A tool palette group can be associated with a tab in the ribbon.
  - The associated tool palette group is displayed in the Tool Palettes window when
  - You right-click on the tab
  - Select Show Related Tool Palette Group.
Creating a New Tab or Panel

- Right-click on the tab in the ribbon.
- Keep the CUI dialog box CLOSED.
- Select Tool Palette group in the Shortcut menu
- Name the group in the submenu
  - Right-click on the tab
  - Select Show Related Tool Palette Group
Overview of Toolbars
Overview of Toolbars

◆ The Toolbar interface can be customized.
◆ Choose CUI dialog box.
◆ All toolbars are listed in Toolbars branch.
◆ Commands are represented by a button on the toolbar.
  ◆ Infrequently used commands can be removed.
  ◆ Toolbar buttons can be rearranged.
  ◆ New toolbars can be created.
Toolbars

◆ Toolbars are NOT displayed except in the AutoCAD Classic workspace.

◆ To display a toolbar
  ◆ Use the Toolbars button on the Windows panel of the View Tab in the ribbon
  ◆ Use the –TOOLBAR command.
Toolbars

-TOOLBAR

- Enter the toolbar name
- The name consists of the menu group and toolbar name.
- Separated by a period.
Toolbars

- Command: -TOOLBAR
  - Enter toolbar name or [ALL]: ACAD.DRAW
- Enter an option[
  Show/Hide/Left/Right/Top/Bottom/Float]<Show>

Customizing Interface Elements and Commands 148
Toolbars

These options are used to hide, show or specify a location for the toolbar.

- **Show**. Makes the toolbar visible.
- **Hide**. Causes the toolbar to be invisible.
- **Left**. Places the toolbar in a docked position at the left side of the AutoCAD window.
- **Right**. Places the toolbar in a docked position at the right side of the AutoCAD window.
- **Top**. Places the toolbar in a docked position at the top of the AutoCAD window.
- **Bottom**. Places the toolbar in a docked position at the bottom of the AutoCAD window.
- **Float**. Places the toolbar as a floating toolbar.
Toolbar Display Options

- Located in the Window Elements of the Display tab of the Options dialog box.
  - Four check boxes
  - And a text box relating to toolbars
Toolbar Display Options

- Show ToolTips
  - Button name is shown next to cursor
- Show shortcut keys in ToolTips check box
  - The shortcut key combination for the command is displayed in the tooltip.
- Show extended ToolTips
  - Determines whether extended tooltips are displayed
- Number of seconds to delay
  - Time elapsed before tooltips are displayed
Creating a New Toolbar

- Open CUI dialog box.
- Right-click Toolbars branch
- Pick New Toolbar
  - New toolbar is added at the bottom of the TB branch
  - Default name is given
  - Change name to something appropriate
Creating a New Toolbar

- New toolbar is highlighted
- Edit Properties in the CUI dialog box.
Adding a Command to a Toolbar

To add a command to a toolbar:
- Expand Toolbars branch in CIAF pane of CUI db.
- Expand the Toolbar branch
- Select a command from Command List:
- Drag it into position.
Overview of Menus
Overview of Menus

- When menu bar is displayed
  - The names of the standard, classic menus appear at the top of the AutoCAD drawing window.
- The menu bar is displayed in the AutoCAD Classic workspace.
Overview of Menus

- To display menu bar in other workspaces
  - Pick the arrow icon at the right-hand end of the Quick Access toolbar.
  - Select Show Menu Bar
  - Or
  - Type MENUBAR command
Overview of Menus

- Menus
  - By Default. 13 menus displayed on the menu bar
  - If no menus are defined in the current CUIx file
    - AutoCAD inserts default:
    - File, Window, Help menus.
Overview of Menus

- Name of menu
  - Make concise as possible

- Menu item names
  - Can be any length.
  - The menu is displayed as wide as its longest menu item name.
  - Each menu can have multiple submenus
  - Menu can have up to 999 items, including submenus.
Overview of Menus

To create an access key (mnemonic) for a menu or menu item,

- place an ampersand (&) before the desired access key character
Shortcut and Access Keys

- Shortcut keys
  - Also called accelerator keys
  - Are key combinations to initiate a command
  - Example: [Ctrl] + [1]
    - Displays or closes the Properties palette.
Shortcut and Access Keys

Access Keys

- Also called mnemonic keys
- Are keys used to access a menu or menu item via the keyboard.

Press the [Alt] key

- Activates the access keys for the menus
- Access keys are shown as underlined letters

- An access key must be unique for a menu or submenu.
Creating a New Menu

- Use the CUI dialog box to create new menu.
- A menu is added to the Menus branch
  - Right-click menus branch to display shortcut menu
  - Pick New Menu from shortcut menu
  - A new menu is added to the bottom of the list of existing menus.
  - Give menu an appropriate name
  - Drag desired commands from the Command List pane and drop them into the new menu.
  - Right-click to add Separator if desired
Creating a New Menu

- New menus are *automatically assigned* an alias of POPn.
  - N is the next available integer.

- The menu is automatically available in all workspaces.
Adding a Submenu (Cascading)

- A **submenu** is a menu contained within another menu.
- Adding a **submenu** to a menu is similar to adding a “main” menu.

- Open the CUI dialog box
- In the CIAF pane, expand branch for menu where the submenu is to be added.
- Pick New Sub-menu in shortcut menu
  - Drag commands from Command List: pane and drop them into new menu.
Adding a Command To a Menu

- Expand the Menus branch in the CIAF pane of the CUI dialog box.
- Expand the branch for the menu to which the command will be added.
- Select a command from the Command List:
- Drag it into position in the tree.
Removing a Menu

- Open the CUI dialog box.
- Expand the tree in the CIAF pane
- Locate menu to be deleted
- Right-click on the menu
- Pick Delete

- This procedure is not recommended. Menu is permanently removed.
- Must be rebuilt to be restored.
- Better way is delete menu from workspace.
Menu Notes

- **Menu Label**
  - Can be as long as needed
  - Should be as brief as possible for easy reading.
  - The menu width is automatically created to fit the width of the longest item.
  - Menus that are longer than the screen display are truncated to fit on the screen.
Menu Notes

◆ Menus are disabled
  ◆ when using the DTEXT command after the rotation angle is entered
  ◆ When creating text with MTEXT
  ◆ When editing text created with MTEXT
  ◆ When using Sketch after the record increment is set.
Sample Custom Commands
Sample Custom Commands

✦ Example #1
   ● 1. This HEXAGON command will start the POLYGON command and draw a six-sided polygon inscribed in a circle.
   ● 2. POLYGON
   ● 6
   ● (select enter)
   ● I
Sample Custom Commands

♦ 3. Name: &Hexagon
♦ Macro: *^C^C polygon;6; \i

♦ The asterisk in front of the ^C^C repeats the command until it is canceled.
♦ The \ in front of i indicates that the macro will wait for user input, in this case the center of the polygon, before continuing.
Sample Custom Commands

Example #2

1. This DOT command draws a solid dot that is 1 unit in diameter.
   - Use the DONUT command.
   - The inside diameter is 0 (zero) and the outside diameter is .1

2. DONUT
   - O
   - .1

3. Name: &Dot
   - Macro: ^C^Cdonut;0;.1
Example #3

1. This X-POINT command sets the PDMODE system variable to 3 and draws an X at the pick point. The command should repeat.

2. PDMODE

3

POINT

(pick the point)

3. Name: &X-Point

Macro: *^C^Cpdmode;3;point
Example #4

1. This command named NOTATION, could be used by a drawing checker or instructor. It allows the user to circle features on a drawing and then add a leader and text.

It first sets the color to red, then draws a circle, snaps a leader to the nearest point that is picked on the circle, and prompts for the text.

User input for text is provided, then a cancel returns the Command: prompt, and the color is set to ByLayer.
Sample Custom Commands

2. -COLOR
   • RED.
   • CIRCLE
     • (pick center point)
     • (pick radius)
   • LEADER
   • NEA
     • (pick a point on the circle)
     • (pick end of leader)
     • (press Enter for automatic shoulder)
   • (enter text) J
     • (press [Enter to cancel)
   • -COLOR
   • BYLAYER
Sample Custom Commands

3. Name: &Notation
Macro:\^C\^C-color;red;circle;\\leader;nea;\\\\:-color;bylayer
Sample Custom Commands

Example #5

1. This is a repeating command named MULTISQUARE that draws one-unit squares oriented at a 0o horizontal angle until the command is canceled.

2. RECTANG
   (pick lower-left corner)
   @1,1

3. Name: &Multisquare
   Macro: *^C^Crectang; \@1,1
Professional Tip

- Some commands such as the COLOR command, display a dialog box.
- Menu macros can provide input to the command line, but cannot control dialog boxes.
Professional Tip

- To access the command-line version of a command, prefix the command name with a hyphen (-)

- Not all commands that display a dialog box have a command-line equivalent.