Scheme of Evaluation
Internal Assessment Test 3 – December 2024

Sub:	User interface design						Code:	21IS733
Date:	16/12/2024	Duration:	90 mins	Max Marks:	50	Sem:	VII	Branch:	ISE

Note: Answer Any five full questions.

Question #		Description	Marks Distribution		Max Marks
1	a)	A toy company wants to design a portal for its customers. Analyze and discuss the types of windows to be used in designing the portal which makes easier access to the users according to their requirement. Any 5 types of windows with description.	2M*5	10M	**10M**
2		List and explain the components of a window with example and diagrams. Any 5 components of windows with details and example	2M*5	10M	**10M**
3	a) b)	How to organize window presentation styles? Draw the styles and explain in detail with neat diagram. Any 2 styles of presentation.	2M 4M*2	2M 8M	**10M**
4	a) b)	What are selection controls? List them (any 4). Discuss any four of them in detail with suitable example. Any four selection controls.	2M 2M*4	2M 8M	**10M**
5	a) b)	Explain the purpose of the prototypes. Discuss any two kinds of prototypes with their importance to the system developer Any 2 prototypes	2M 4M*2	2M 8M	**10M**

6		Explain the following kinds of tests.		10M	**10M**
		a. Think Aloud Evaluation	3M		
		b. Usability Test	3M		
		c. Heuristic Evaluation (or) Cognitive Walkthroughs	4M		

Scheme of Evaluation
Internal Assessment Test 3 – December 2024

Sub:	User Interface Design							**Code:**	21IS733
Date:	16/12/2024	Duration :	90mins	Max Marks:	50	**Sem:**	VII	**Branch:**	ISE

Note: <u>Answer Any full five questions</u>

Q. 1 A toy company wants to design a portal for its customers. Analyze and discuss the types of windows to be used in designing the portal which makes easier access to the users according to their requirement.

Types of Windows

1. Primary windows

Figure 5.7 Microsoft Windows primary window.

■ Proper usage:

— Should represent an independent function or application.

— Use to present constantly used window components and controls.

• Menu bar items that are:

— Used frequently.

— Used by most, or all, primary or secondary windows.

• Controls used by dependent windows.

— Use for presenting information that is continually updated.

• For example, date and time.

— Use for providing context for dependent windows to be created.

— Do not:

• Divide an independent function into two or more primary windows.

• Present unrelated functions in one primary window.
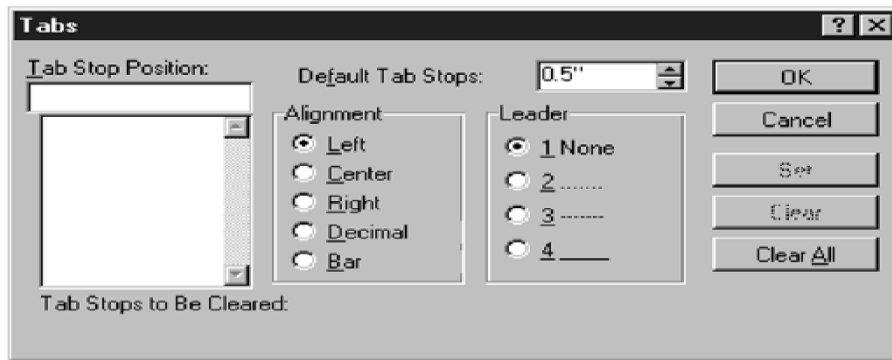
2. Secondary window

Figure 5.8 Microsoft Windows secondary window.

■ Proper usage:

— For performing subordinate, supplemental, or ancillary actions that are:

• Extended or more complex in nature.

• Related to objects in the primary window.

— For presenting frequently or occasionally used window components.

■ Important guidelines:

— Should typically not appear as an entry on the taskbar.

— A secondary window should not be larger than 263 dialog units x 263 dialog units.

Secondary windows are supplemental windows. Secondary windows may be dependent upon a primary window or displayed independently of the primary window. They structurally resemble a primary window, possessing some of the same action controls (Close button) and possibly a What's This? button.

A *dependent* secondary window is one common type. It can only be displayed from a command on the interface of its primary window. It is typically associated with a single data object, and appears on top of the active window when requested. It is movable, and scrollable. If necessary, it uses the primary window's menu bar. Most systems permit the use of multiple secondary windows to complete a task. In general, dependent secondary windows are closed when the primary window closes, and hidden when their primary window is hidden or minimized.

An *independent* secondary window can be opened independently of a primary window—for example, a property sheet displayed when the user clicks the Properties command on the menu of a desktop icon. An independent secondary window can typically be closed without regard to the state of any primary window unless there is an obvious relationship to the primary window.

Model-prompt and modeless-word search

■ Modal:

Use when interaction with any other window must not be permitted.

Use for:

Presenting information.

— For example, messages (sometimes called a message box).

Receiving user input.

For example, data or information (sometimes called a prompt box).

Asking questions.

For example, data, information, or directions (sometimes called a question box).

Use carefully because it constrains what the user can do.

■ Modeless:

Use when interaction with other windows must be permitted.

— Use when interaction with other windows must be repeated.

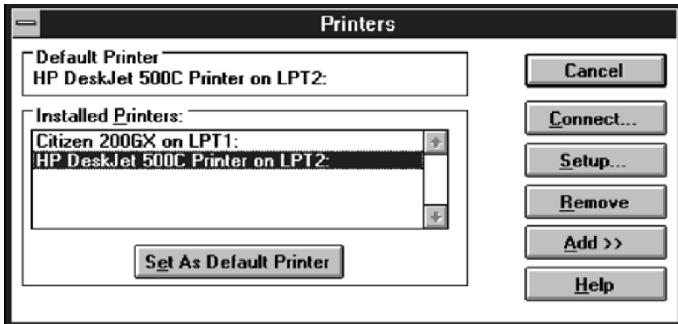Cascading and unfolding*Cascading and Unfolding*

Figure 5.9 Printers secondary window with Connect... cascade button.
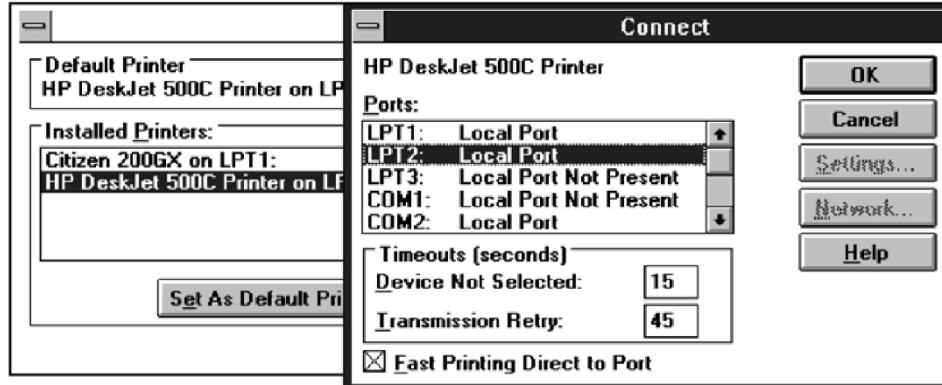

Figure 5.10 Cascading Connect secondary window.

■ Cascading:

Purpose:

To provide advanced options at a lower level in a complex dialog.

Guidelines:

Provide a command button leading to the next dialog box with a "To a Window" indicator, an ellipsis (. . .).

- Present the additional dialog box in cascaded form.
- Provide no more than two cascades in a given path.
- Do not cover previous critical information.

Title Bar.

Relevant displayed information.

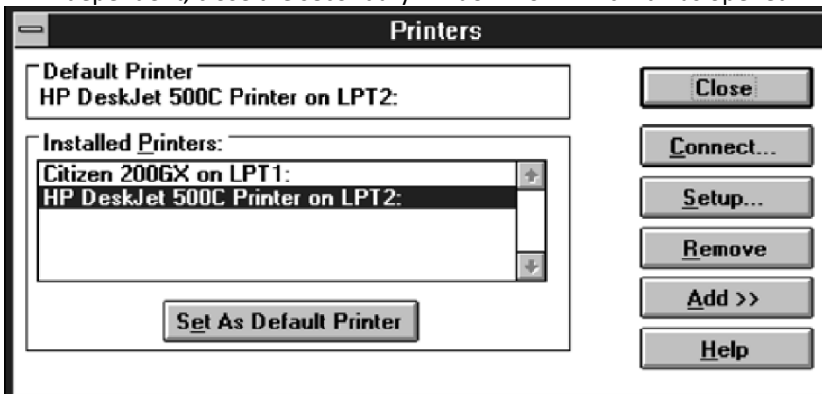• If independent, close the secondary window from which it was opened.

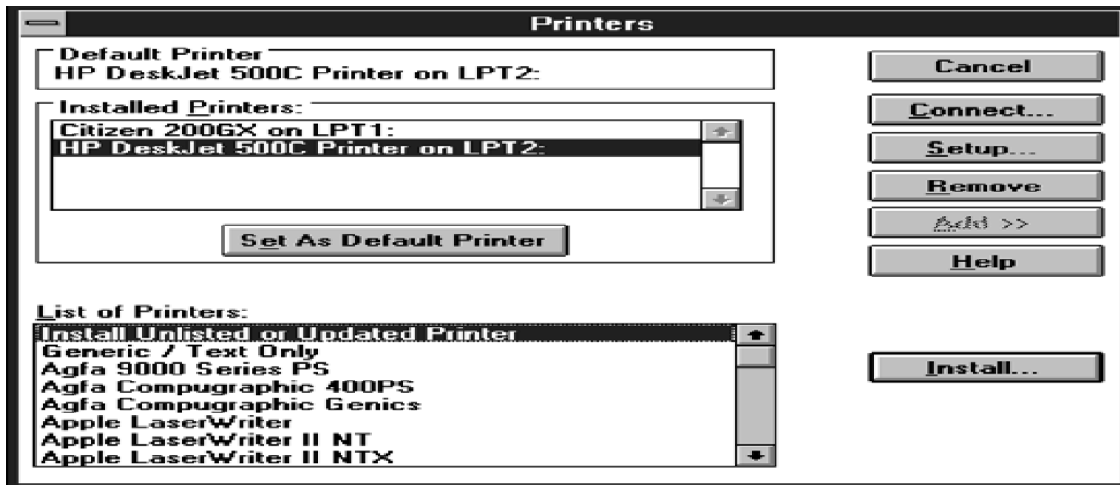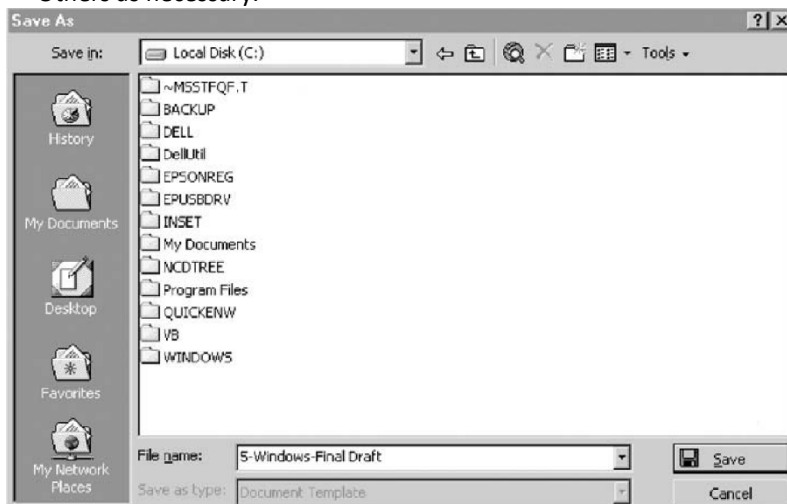Figure 5.11 Printers secondary window with *Add >>* unfolding button.

Figure 5.12 Unfolded Printers secondary window.

3. Dialog box
■ Use for presenting brief messages.
■ Use for requesting specific, transient actions.
■ Use for performing actions that:
Take a short time to complete.
Are not frequently changed.
■ Command buttons to include:
OK.
Cancel.
— Others as necessary.

Property sheet■ Use for presenting the complete set of properties for an object.
■ Categorize and group within property pages, as necessary.
Use tabbed property pages for grouping peer-related property sets.
The recommended sizes for property sheets are:
• 252 DLUs wide x 218 DLUs high
• 227 DLUs wide x 215 DLUs high
• 212 DLUs wide x 188 DLUs high
         Command buttons to include:
• OK.
• Cancel.
• Apply.
• Reset.
• Others as necessary.
For single property sheets, place the commands on the sheet.
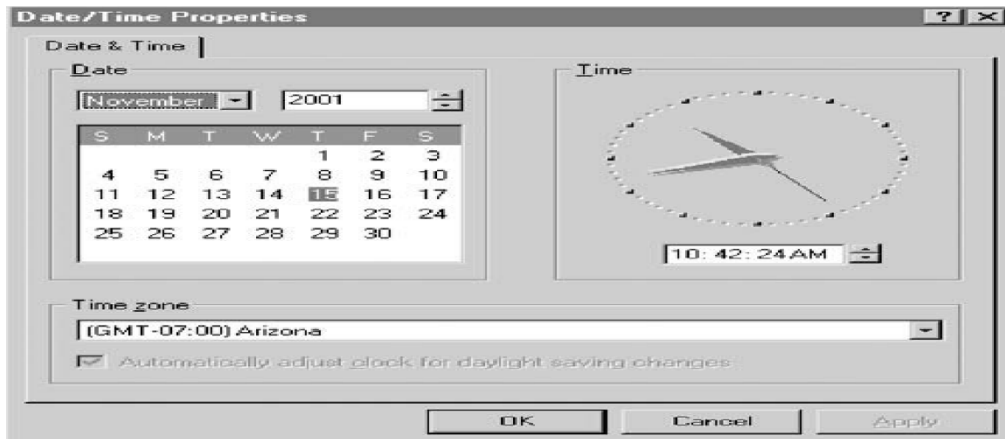— For tabbed property pages, place the commands outside the tabbed pages.

Figure 5.14 Microsoft Windows property sheet.

Figure 5.15 Microsoft Windows property sheet tabbed pages.

Property Inspectors
■ Use for displaying only the most common or frequently accessed object properties.
■ Make changes dynamically.

Figure 5.16 Microsoft Windows property inspector.

4. Message Boxes
■ Use for displaying a message about a particular situation or condition.
■ Command buttons to include:
OK.
Cancel.
Help.
Yes and No.
Stop.
Buttons to correct the action that caused the message box to be displayed.
■ Enable the title bar close box only if the message includes a cancel button.
■ Designate the most frequent or least destructive option as the default command button.
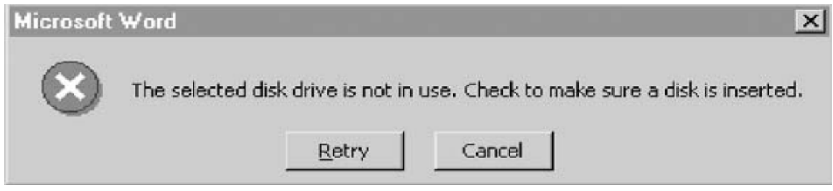
Palette Windows
■ Use to present a set of controls.
■ Design as resizable.
Alternately, design them as fixed in size.

Figure 5.18 Microsoft Windows palette window.

5. Pop-up Windows
■ Use pop-up windows to display:
Additional information when an abbreviated form of the information is the main presentation.
Textual labels for graphical controls.
Context-sensitive Help information.

Figure 5.19 Microsoft Windows pop-up window**.**

Q. 2 List and explain the components of a window with example and diagrams.

Components of a Window

A typical window may be composed of up to a dozen or so elements. Some appear on all windows; others only on certain kinds of windows, or under certain conditions. For consistency purposes, these elements should always be located in the same position within a window.

Figure 5.1 Microsoft Windows primary window.

Window components are

1. **Frame**

A window will have a frame or border, usually rectangular in shape, to define its boundaries

and distinguish it from other windows. While a border need not be rectangular, this shape is a preferred shape for most people. Also, textual materials, which are usually read from left to right, fit most efficiently within this structure. The border comprises a line of variable thickness and color.

2. **Title Bar**

The title bar is the top edge of the window, inside its border and extending its entire width. This title bar is also referred to by some platforms as the *caption, caption bar,* or *title area*. The title bar contains a descriptive title identifying the purpose or content of the window.

3. Title Bar Icon

Located at the left corner of the title bar in a primary window, this button is used in Windows to retrieve a pull-down menu of commands that apply to the object in the window. It is 16×16 version of the icon of the object being viewed. When clicked with the secondary mouse button, the commands applying to the object are presented. Microsoft suggests that:

- If the window contains a tool or utility (that is, an application that does not create,

  load, and save its own data files), a small version of the application's icon should be placed there instead.

- If the application creates, loads, and saves documents or data files and the window

  represents the view of one of its files, a small version of the icon that represents its document or data file type should be placed there.

- Even if the user has not yet saved the file, display the data file icon rather than the

  application icon, and again display the data file icon after the user saves the file.

4. Window Sizing Buttons

A window's title bar must have equivalent commands on the pop-up or shortcut menu for that window. When these buttons are displayed, use the following guidelines:

When a window does not support a command, do not display its command button.

The *Close* button always appears as the rightmost button. Leave a gap between it and any other buttons. The *Minimize* button always precedes the *Maximize* button. The *Restore* button always replaces the *Maximize*

button or the *Minimize* button when that command is carried out.

5. What's This? Button

The *What's This?* Button, which appears on secondary windows and dialog boxes, is used to invoke the What's This? Windows command to provide contextual Help about objects displayed within a secondary window.

Figure 5.2 What's This? button.

6. Menu Bar

A menu bar is used to organize and provide access to actions. It is located horizontally

at the top of the window, just below the title bar. A menu bar contains a list of topics or

items that, when selected, are displayed on a pull-down menu beneath the choice. A

system will typically provide a default set of menu actions that can be augmented by

an application. In the past, some platforms have called the menu bar an *action bar*.

7. Status Bar

Information of use to the user can be displayed in a designated screen area or areas. They

may be located at the top of the screen in some platforms and called a *status area*, or at

the screen's bottom. Microsoft recommends the bottom location and refers to this area

as the *status bar.* It is also referred to by other platforms as a *message area* or *message bar*.

8. Scroll Bars

When all display information cannot be presented in a window, the additional information

must be found and made visible. This is accomplished by scrolling the display's contents through use of a scroll bar. A scroll bar is an elongated rectangular container consisting of a scroll area or shaft, a slider box or elevator, and arrows or anchors at each end. For vertical scrolling, the scroll bar is positioned at the far right side of the work area, extending its entire length. Horizontal scrolling is accomplished through a scroll bar located at the bottom of the work area.

9. Split Box

A window can be split into two or more pieces or panes by manipulating a *split box* located

above a vertical scroll bar or to the left of a horizontal scroll bar. A split box is sometimes

referred to as a *split bar*. A window can be split into two or more separate viewing areas that are called *panes*. Splitting a window permits multiple views of an object. A split window allows the user to: Examine two parts of a document at the same time.

Display different, yet simultaneous, views of the same information.

10. Toolbar

They are sometimes called *command bars*. Toolbars are designed to provide quick access to specific commands or options. Specialized toolbars are sometimes referred to as *ribbons, toolboxes, rulers,* or *palettes*. Each toolbar band includes a single-grip handle to enable the user to resize or rearrange the toolbars. When the user moves the pointer over the grip, it changes to a two-headed arrow. When the user drags the grip, the pointer changes to a split move pointer.
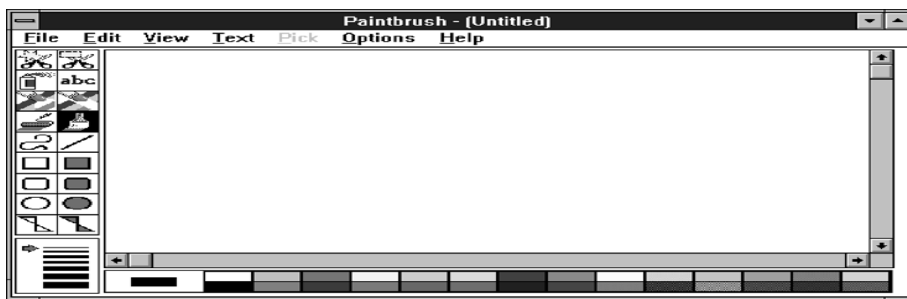
11. Command Area

In situations where it is useful for a command to be typed into a screen, a command area

can be provided. The desired location of the command area is at the bottom of the window.

If a horizontal scroll bar is included in the window, position the command area just below it. If a message area is included on the screen, locate the command area just above it.

12. Size Grip

A size grip is a Microsoft Windows special handle included in a window to permit it to

be resized. When the grip is dragged the window resizes, following the same conventions

as the sizing border. Three angled parallel lines in the lower-right corner of a window

designate the size grip. If the window possesses a status bar, the grip is positioned

at the bar's right end. Otherwise, it is located at the bottom of a vertical

scroll bar, the

right side of a horizontal scroll bar, or the junction point of the two bars.

13. Work Area

The work area is the portion of the screen where the user performs tasks. It is the open

area inside the window's border and contains relevant peripheral screen components

such as the menu bar, scroll bars, or message bars.

Q. 3 How to organize window presentation styles?

Draw the styles and explain in detail with neat diagram.

Window Presentation Styles

The presentation style of a window refers to its spatial relationship to other windows. There are two basic styles, commonly called tiled or overlapping.

**I) Tiled Windows**

Figure 5.4   Tiled windows.

Tiled windows, the first and oldest kind of window, are felt to have these **advantages**:
- The system usually allocates and positions windows for the user, eliminating the necessity to make positioning decisions.
- Open windows are always visible, eliminating the possibility of them being lost and forgotten.
- Every window is always completely visible, eliminating the possibility of information being hidden.
- They are perceived as less complex than overlapping windows, possibly because there are fewer management operations or they seem less "magical."
- They are easier, according to studies, for novice or inexperienced people to learn and use.
- They yield better user performance for tasks where the data requires little window manipulation to complete the task.

Perceived **disadvantages** include the following:
- Only a limited number can be displayed in the screen area available.
- As windows are opened or closed, existing windows change in size. This can be annoying.
- As windows change in size or position, the movement can be disconcerting.
- As the number of displayed windows increases, each window can get very tiny.
- The changes in sizes and locations made by the system are difficult to predict.
- The configuration of windows provided by the system may not meet the user's needs.
- They are perceived as crowded and more visually complex because window borders are flush against one another, and they fill up the whole screen. Crowding is accentuated if borders contain scroll bars or control icons. Viewer attention may be drawn to the border, not the data.
- They permit less user control because the system actively manages the windows.

II) Overlapping Windows
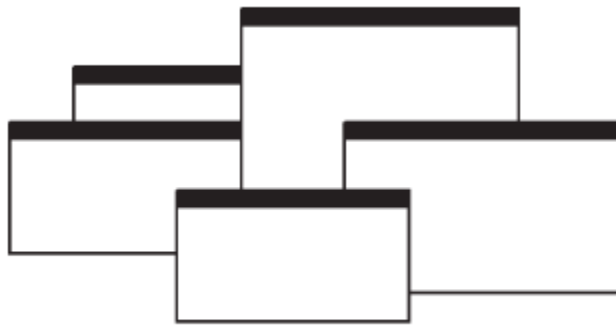
Figure 5.5 Overlapping windows.

Overlapping windows, illustrated in Figure 5.5, may be placed on top of one another like papers on a desk. They possess a three-dimensional quality, appearing to lie on different planes.
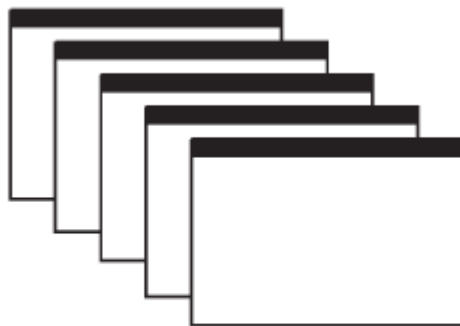They have the following **advantages**:
- Visually, their look is three-dimensional, resembling the desktop that is familiar to the user.
- Greater control allows the user to organize the windows to meet his or her needs.
- Windows can maintain larger sizes.
- Windows can maintain consistent sizes.
- Windows can maintain consistent positions.
- Screen space conservation is not a problem, because windows can be placed on top of one another.
- There is less pressure to close or delete windows no longer needed.
- The possibility exists for less visual crowding and complexity. Larger borders can be maintained around window information, and the window is more clearly set off against its background. Windows can also be expanded to fill the entire display.
- They yield better user performance for tasks where the data requires much window manipulation to complete the task.

Disadvantages include the following:
- They are operationally much more complex than tiled windows. More control functions require greater user attention and manipulation.
- Information in windows can be obscured behind other windows.
- Windows themselves can be lost behind other windows and be presumed not to exist.
- That overlapping windows represent a three-dimensional space is not always realized by the user.
- Control freedom increases the possibility for greater visual complexity and crowding. Too many windows, or improper offsetting, can be visually overwhelming.

III) Cascading Windows

Figure 5.6 Cascading windows.

A special type of overlapping window has the windows automatically arranged in a regular progression.
Advantages of this approach include the following:
- No window is ever completely hidden.
- Bringing any window to the front is easier.
- It provides simplicity in visual presentation and cleanness.

IV) Picking a Presentation Style
■ Use tiled windows for:
— Single-task activities.

— Data that needs to be seen simultaneously.
— Tasks requiring little window manipulation.
— Novice or inexperienced users.
■ Use overlapping windows for:
— Switching between tasks.
— Tasks necessitating a greater amount of window manipulation.
— Expert or experienced users.
— Unpredictable display contents.

Tiled windows. Tiled windows seem to be better for single-task activities and data that must be seen simultaneously. A study found that tasks requiring little window manipulation were carried out faster using tiled windows. They also found that novice users performed better with tiled windows, regardless of the task.

Overlapping windows. Overlapping windows seem to be better for situations that necessitate switching between tasks. A research study concluded that tasks requiring much window manipulation could be performed faster with overlapping windows but only if user window expertise existed. For novice users, tasks requiring much window manipulation were carried out faster with tiled windows. Therefore, the advantage to overlapping windows comes only after a certain level of expertise is achieved. Overlapping windows are the preferred presentation scheme.

Window Management

Microsoft Windows also provides several window management schemes,
1. A *single document interface*-A single primary window with a set of secondary windows.
2. *A multiple-document interface*-— A technique for managing a set of windows where documents are opened into windows.
— Contains:
• A single primary window, called the parent.
• A set of related document or child windows, each also essentially a primary window.
The purpose of this scheme of windows is to provide multiple views of the same object, to permit comparisons
among related objects, and to present multiple parts of an application
3. *Workbooks*-Tabs are used as a navigational interface to move between different sections.
4. *projects*.-A technique that consists of a container: a project window holding a set of objects.

Window Organization
■ Organize windows to support user tasks.
■ Support the most common tasks in the most efficient sequence of steps.
■ Use primary windows to:
— Begin an interaction and provide a top-level context for dependent windows.
— Perform a major interaction.
■ Use secondary windows to:
— Extend the interaction.
— Obtain or display supplemental information related to the primary window.
■ Use dialog boxes for:
— Infrequently used or needed information.
— "Nice-to-know" information.

Number of Windows
■ Minimize the number of windows needed to accomplish an objective.

Q.4 What are selection controls? List them.
Discuss any four of them in detail with suitable example.

Selection Controls

A selection control presents on the screen all the possible alternatives, conditions, or choices that may exist for an entity, property, or value. The relevant item or items are selected from those displayed. Some selection controls present all the alternatives together, visibly on a screen; others may require an action to retrieve the entire listing and/ or scrolling to view all the alternatives. Selection controls

include
1. radio buttons
2. check boxes
3. list boxes
4. drop-down / pop-up list boxes
5. palettes

1. Radio Buttons
■ Description:
— A two-part control consisting of the following:
• Small circles, diamonds, or rectangles.
• Choice descriptions.
— When a choice is selected:
• The option is highlighted.
• Any existing choice is automatically unhighlighted and deselected.
■ Purpose:
— To set one item from a small set of mutually exclusive options (2 to 8).
■ Advantages:
— Easy-to-access choices.
— Easy-to-compare choices.
— Preferred by users.
■ Disadvantages:
— Consume screen space.
— Limited number of choices.
■ Proper usage:
— For setting attributes, properties, or values.
— For mutually exclusive choices (that is, only one can be selected).
— Where adequate screen space is available.
— Most useful for data and choices that are:
• Discrete.
• Small and fixed in number.
• Not easily remembered.
• In need of a textual description to meaningfully describe the alternatives.
• Most easily understood when the alternatives can be seen together and compared
to one another.
• Never changed in content.
— Do not use:
• For commands.
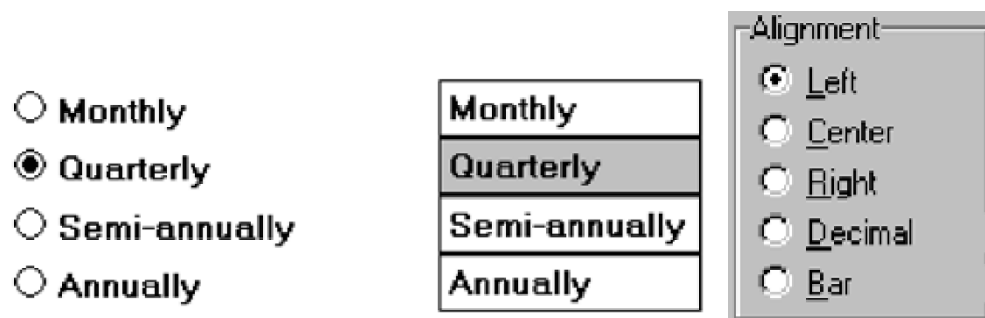• Singly to indicate the presence or absence of a state.

Fig : Radio Buttons

Choice Descriptions
■ Provide meaningful, fully spelled-out choice descriptions clearly describing the values or effects set
by the radio buttons.
■ Display in a single line of text.
■ Display using mixed-case letters, using the sentence style.
■ Position descriptions to the right of the button. Separate them by at least one space from the button.
■ When a choice is conditionally unavailable for selection, display the choice description grayed out
or dimmed.
■ Include a None choice if it adds clarity.

Size
■ Show a minimum of two choices, a maximum of eight.

Defaults
■ When the control possesses a state or affect that has been predetermined to have a higher probability of selection than the others, designate it as the default and display its button filled in.
■ When the control includes choices whose states cannot be predetermined, display all the buttons without setting a dot, or in the *indeterminate* state.
■ When a multiple selection includes choices whose states vary, display the buttons in another unique manner, or in the *mixed value* state.

Structure
■ A columnar orientation is the preferred manner of presentation.
■ Left-align the buttons and choice descriptions.

○ Red

◉ Yellow

○ Green

○ Blue

■ If vertical space on the screen is limited, orient the buttons horizontally.
■ Provide adequate separation between choices so that the buttons are associated with
the proper description.
— A distance equal to three spaces is usually sufficient.

◉ Green   ○ Blue   ○ Yellow   ○ Red

■ Enclose the buttons in a border to visually strengthen the relationship they possess.
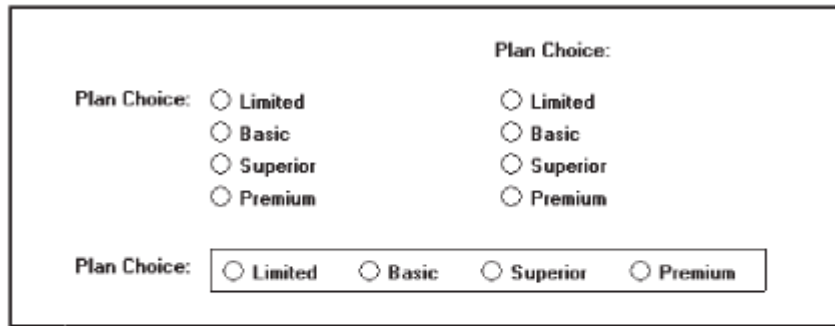
○ Red
○ Yellow
○ Green
○ Blue

○ Green   ○ Blue   ○ Yellow   ○ Red

	○	○	○	○
Plan Choice:	Limited	Basic	Superior	Premium

Plan Choice:	Limited	Basic	Superior	Premium
	○	○	○	○

Plan Choice: ○ Limited ○ Basic ○ Superior ○ Premium

Plan Choice: Limited ○   Basic ○   Superior ○   Premium ○
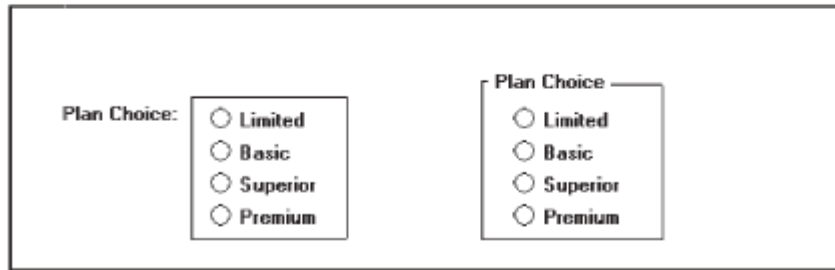
Poor

Plan Choice: ○ Limited   ○ Basic   ○ Superior   ○ Premium
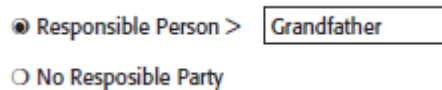
Better

Still Better

Best

Fig : Ways to, and not to, present radio buttons.

Organization
■ Arrange selections in expected order or follow other patterns such as frequency of occurrence, sequence of use, or importance.
— For selections arrayed top to bottom, begin ordering at the top.
— For selections arrayed left to right, begin ordering at the left.
■ If, under certain conditions, a choice is not available, display it subdued or less brightly than the available choices.

Related Control
■ Position any control related to a radio button immediately to the right of the choice description.
■ If the radio button choice description also acts as the label for the control that follows it, end the label with an arrow (>).

Captions
■ Structure:
— Provide a caption for each radio button control.
• Exception: In screens containing only one radio button control, the screen title may serve as the caption.
■ Display:
— Fully spelled out.
— In mixed-case letters, capitalizing the first letter of all significant words.
■ Columnar orientation:
— With a control border, position the caption:
• Upper-left-justified within the border.
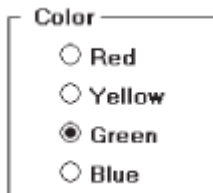
• Alternately, the caption may be located to the left of the topmost choice description.

Color:
> ◉ Red
> ○ Yellow
> ○ Green
> ○ Blue

— Without an enclosing control border, position the caption:
• Left-justified above the choice descriptions, separated by one space line.

Color:

◉ Red
○ Yellow
○ Green
○ Blue

• Alternately, the caption may be located to the left of the topmost choice description.

Color: ○ Red
        ○ Yellow
        ◉ Green
        ○ Blue

■ Horizontal orientation:
— Position the caption to the left of the choice descriptions.

Color: ○ Green   ○ Blue   ○ Yellow   ○ Red

• Alternately, with an enclosing control border, left-justified within the border.

┌ Color ─────────────────────────────┐
│ ○ Green   ○ Blue   ○ Yellow   ◉ Red │
└─────────────────────────────────────┘

Keyboard Equivalents
■ Assign a keyboard mnemonic to each choice description.
■ Designate the mnemonic by underlining the applicable letter in the choice description.

◉ R̲ed

Assign unique keyboard mnemonics for each alternative in the standard way, choosing the first letter (or another) and designating it by character underlining.

Selection Method and Indication
■ Pointing:
— The selection target area should be as large as possible.
• Include the button and the choice description text.
— Highlight the selection choice in some visually distinctive way when the cursor's resting on it and the choice is available for selection.
• This cursor should be as long as the longest choice description plus one space at each end. Do not place the cursor over the small button.

○ Red
○ Yellow
○ Green
○ ⌐Blue ⌐

■ Activation:
— When a choice is selected, distinguish it visually from the unselected choices.
• A radio button should be filled in with a solid dark dot or made to look depressed or higher through use of a shadow.
— When a choice is selected, any other selected choice must be deselected.
■ Defaults:
— If a radio button control is displayed that contains a choice previously selected or

a default choice, display the selected choice as set in the control.

2. Check Boxes
■ Description:
— A two-part control consisting of a square box and choice description.
— Each option acts as a switch and can be either "on" or "off."
• When an option is selected (on), a mark such as an "X" or "check" appears
within the square box, or the box is highlighted in some other manner.
• Otherwise the square box is unselected or empty (off).
— Each box can be:
• Switched on or off independently.
• Used alone or grouped in sets.
■ Purpose:
— To set one or more options as either on or off.
■ Advantages
— Easy-to-access choices.
— Easy-to-compare choices.
— Preferred by users.
■ Disadvantages:
— Consume screen space.
— Limited number of choices.
— Single check boxes difficult to align with other screen controls.
■ Proper usage:
— For setting attributes, properties, or values.
— For nonexclusive choices (that is, more than one can be selected).
— Where adequate screen space is available.
— Most useful for data and choices that are:
• Discrete.
• Small and fixed in number.
• Not easily remembered.
• In need of a textual description to describe meaningfully.
• Most easily understood when the alternatives can be seen together and compared
to one another.
• Never changed in content.
— Can be used to affect other controls.
— Use only when both states of a choice are clearly opposite and unambiguous.

Figure 7.39 Check boxes.

Figure 7.40 Check boxes.

Choice Descriptions
■ Provide meaningful, fully spelled-out choice descriptions clearly describing the values
or effects set by the check boxes.
■ Display them in a single line of text.
■ Display them using mixed-case letters in sentence style.
■ Position descriptions to the right of the check box. Separate by at least one space from the box.
■ When a choice is unavailable for selection under a certain condition, display the choice description
visually dimmed.
Size
■ Show a minimum of one choice, a maximum of eight.
Defaults

■ When the control possesses a state or affect that has been preset, designate it as the default and display its check box marked.

■ When a multiple selection includes choices whose states vary, display the buttons in another unique manner, or the *mixed value* state.

Structure

■ Provide groupings of related check boxes.

■ A columnar orientation is the preferred manner of presentation for multiple related check boxes.

■ Left-align the check boxes and choice descriptions.

☒ Bold

☐ Italic

☐ Underline

■ If vertical space on the screen is limited, orient the boxes horizontally.

■ Provide adequate separation between boxes so that the buttons are associated with the proper description.

— A distance equal to three spaces is usually sufficient.

☒ Bold   ☐ Italic   ☐ Underline

■ Enclose the boxes in a border to visually strengthen the relationship they possess.


Poor

Better

Still Better

Best

Fig : Ways to, and not to, present check boxes.

Organization
■ Arrange selections in logical order or follow other patterns such as frequency of occurrence, sequence of use, or importance.
— For selections arrayed top to bottom, begin ordering at the top.
— For selections arrayed left to right, begin ordering at the left.
■ If, under certain conditions, a choice is not available, display it subdued or less brightly than the available choices.

Related Control
■ Position any control related to a check box immediately to the right of the choice description.
— If a the check box choice description also acts as the label for the control that follows it , end the label with an arrow (>).
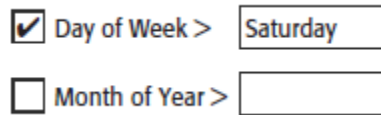
Captions
■ Structure:
— Provide a caption for each grouping of related check boxes.
• Exception: In screens containing only one check box grouping, the screen title may serve as the caption.
— Display:
• Fully spelled out.
• In mixed-case letters capitalizing the first letter of all significant words.
■ Columnar orientation:
— With a control border, position the caption:
• Upper-left-justified within the border.
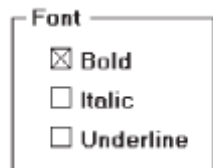
• Alternately, the caption may be located to the left of the topmost choice description.
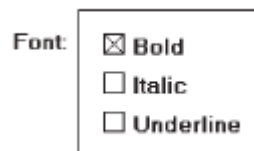
— Without an enclosing control border, position the caption:
• Left-justified above the choice descriptions separated by one space line.

Font:

☒ Bold
☒ Italic
☐ Underline

• Alternately, the caption may be located to the left of the topmost choice description.

Font: ☒ Bold
       ☐ Italic
       ☒ Underline

■ Horizontal orientation
— Position the caption to the left of the choice descriptions.

Font: ☒ Bold   ☐ Italic   ☐ Underline

Font: | ☒ Bold   ☐ Italic   ☐ Underline |

• Alternately, with an enclosing control border, it should be left-justified within the border.

┌─ Font ─────────────────────────
│ ☒ Bold   ☐ Italic   ☐ Underline │

— Be consistent in caption style and orientation within a screen.

Keyboard Equivalents
■ Assign a keyboard mnemonic to each check box.
■ Designate the mnemonic by underlining the applicable letter in the choice description.

☐ Underline

Selection Method and Indication
■ Pointing:
— The selection target area should be as large as possible.
• Include the check box and the choice description text.
— Highlight the selection choice in some visually distinctive way when the cursor's resting on it and the choice is available for selection.
• This cursor should be as long as the longest choice description plus one space at each end. Do not place the cursor over the check box.

☐ Bold

☐ Italic
☒ Underline

■ Activation:
— When a choice is selected, distinguish it visually from the non-selected choices.
• A check box should be filled in or made to look depressed or higher through use of a shadow.
■ Defaults:
— If a check box is displayed that contains a choice previously selected or default choice, display the selected choice as set in the control.
■ Select/deselect all:
— Do not use *Select All* and *Deselect All* check boxes.
■ Mixed-value state:
— When a check box represents a value, and a multiple selection encompasses multiple value occurrences set in both the on and off state, display the check box in a *mixed value* state.
• Fill the check box with another easily differentiable symbol or pattern.

☐ Italic

☐ Underline

— Toggle the check box as follows:
• Selection 1: Set the associated value for all elements. Fill the check box with an "X" or "check."
• Selection 2: Unset the value for all associated elements. Blank-out the check box.
• Selection 3: Return all elements to their original state. Fill the check box with
the mixed value symbol or pattern.

3. List Boxes
■ Description:
— A permanently displayed box-shaped control containing a list of attributes or objects from which:
• A single selection is made (mutually exclusive), or
• Multiple selections are made (non-mutually-exclusive).
— The choice may be text, pictorial representations, or graphics.
— Selections are made by using a mouse to point and click.
— Capable of being scrolled to view large lists of choices.
— No text entry field exists in which to type text.
— A list box may be may be associated with a *summary list box* control, which allows the selected
choice to be displayed or an item added to the list.
■ Purpose:
— To display a collection of items containing:
• Mutually exclusive options.
• Non-mutually-exclusive options.
■ Advantages:
— Unlimited number of choices.
— Reminds users of available options.
— Box always visible.
■ Disadvantages:
— Consumes screen space.
— Often requires an action (scrolling) to see all list choices.
— The list content may change, making it hard to find items.
— The list may be ordered in an unpredictable way, making it hard to find items.
■ Proper usage:
— For selecting values or setting attributes.
— For choices that are:
• Mutually exclusive (only one can be selected).
• Non-mutually-exclusive (one or more may be selected).
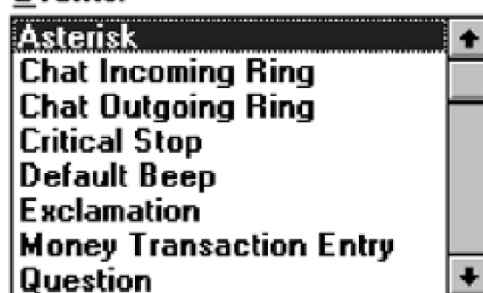— Where screen space is available.
— For data and choices that are:
• Best represented textually.
• Not frequently selected.
• Not well known, easily learned, or remembered.
• Ordered in an unpredictable fashion.
• Frequently changed.
• Large in number.
• Fixed or variable in list length.
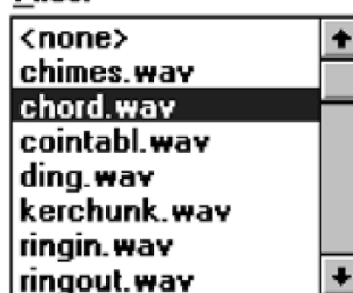— When screen space or layout considerations make radio buttons or check boxes impractical.

Fig : List Box

List Box General Guidelines
First, general list box guidelines will be presented. Then, specific guidelines for single and multiple-selection lists will be reviewed.
Selection Descriptions
■ Clearly and meaningfully describe the choices available. Spell them out as fully as possible.
— Graphical representations must clearly represent the options.
■ Present in mixed case, using the sentence style structure.
■ Left-align into columns.
List Size
■ Not actual limit in size.
■ Present all available alternatives.
■ Require no more than 40 page-downs to search a list.
— If more are required, provide a method for using search criteria or scoping the options.
Box Size
■ Must be long enough to display 6 to 8 choices without requiring scrolling.
— Exceptions:
• If screen space constraints exist, the box may be reduced in size to display at least three items.
• If it is the major control within a window, the box may be larger.
— If more items are available than are visible in the box, provide vertical scrolling to display all items.
■ Must be wide enough to display the longest possible choice.

— When box cannot be made wide enough to display the longest entry:
• Make it wide enough to permit entries to be distinguishable, or,
• Break the long entries with an ellipsis (...) in the middle, or,
• Provide horizontal scrolling.
Organization
■ Order in a logical and meaningful way to permit easy browsing.
— Consider using separate controls to enable the user to change the sort order or filter items displayed in the list.
■ If a particular choice is not available in the current context, omit it from the list.
— Exception: If it is important that the existence and unavailability of a particular list item be communicated, display the choice dimmed or grayed out instead of deleting it.

Layout and Separation
■ Enclose the choices in a box with a solid border.
— The border should be the same color as the choice descriptions.
■ Leave one blank character position between the choice descriptions and the left border.
■ Leave one blank character position between the longest choice description in the list and the right border, if possible.

Captions
■ Use mixed-case letters.
■ The preferred position of the control caption is above the upper-left corner of the list box.

Destination:

Alternately, the caption may be located to the left of the topmost choice description.

• Be consistent in caption style and orientation within a screen, and related screens.

Disabling
■ When a list box is disabled, display its caption and show its entries as grayed out or dimmed.
Selection Method and Indication
■ Pointing:
— Highlight the selection choice in some visually distinctive way when the pointer or cursor is resting on it and the choice is available for selection.
■ Selection:
— Use a reverse video or reverse color bar to surround the choice description when it is selected.
— The cursor should be as wide as the box itself.

— Mark the selected choice in a distinguishing way.
■ Activation:
— Require the pressing of a command button when an item, or items, is selected.

Single-Selection List Boxes
■ Purpose:
— To permit selection of only one item from a large listing.
■ Design guidelines:
— Related text box
• If presented with an associated text box control:

— Position the list box below and as close as possible to the text box.
— The list box caption should be worded similarly to the text box caption.

Destination: [ ]

Destination:
| Australia |
| Canada |
| **England** |
| France |
| Germany |
| New Zealand |
| Netherlands |

— If the related text box and the list box are very close in proximity, the caption may be omitted from the list box.

Destination: [ ]

| Australia |
| Canada |
| **England** |
| France |
| Germany |
| New Zealand |
| Netherlands |

— Use the same background color for the text box as is used in the list box.
— Defaults:
• When the list box is first displayed:
— Present the currently active choice highlighted or marked with a circle or diamond to the left of the entry.
— If a choice has not been previously selected, provide a default choice and display it in the same manner that is used in selecting it.
— If the list represents mixed values for a multiple selection, do not highlight an entry.
— Other:
• Follow other relevant list box guidelines.

Extended and Multiple-Selection List Boxes
■ Purpose:
— To permit selection of more than one item in a long listing.
• Extended list box: Optimized for individual item or range selection.
• Multiple-selection list box: Optimized for independent item selection.
■ Design guidelines:
— Selection indication:
• Mark the selected choice with an X or check mark to the left of the entry.
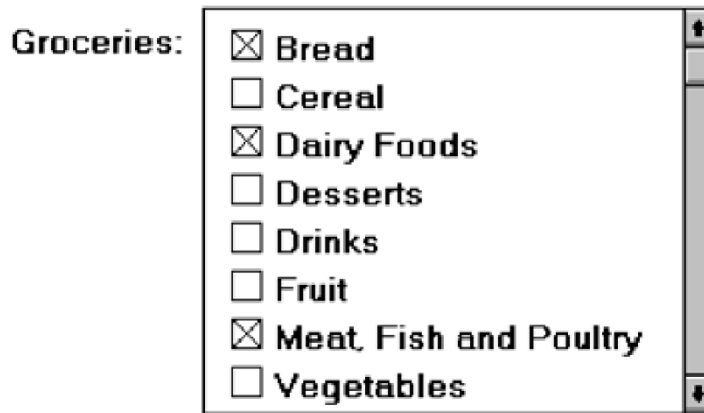
• Consider providing a *summary list box*.
— Position it to the right of the list box.
— Use the same colors for the summary list box as are used in the list box.
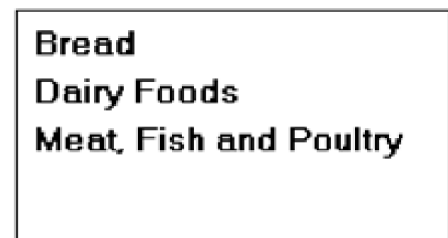
— Provide command buttons to *Add* (one item) or *Add All* (items) to the summary list box, and *Remove* (one item) or *Remove All* (items) from the summary list box.
• Consider providing a display-only text control indicating how many choices
have been selected.
— Position it justified upper-right above the list box.

— Select All and Deselect All buttons
• Provide command buttons to accomplish fast *Select All* and *Deselect All* actions,
when these actions must be frequently or quickly performed.
— Defaults:
• When the list box is first displayed:
— Display the currently active choices highlighted.
— Mark with an X or check mark to the left of the entry.
— If the list represents mixed values for a multiple selection, do not highlight
an entry.
— Other:
• Follow other relevant list box guidelines.

List View Controls

■ Description:

— A special extended-selection list box that displays a collection of items, consisting of an icon and a label.

— The contents can be displayed in four different views:

• Large Icon: Items appear as a full-sized icon with a label below.

• Small Icon: Items appear as a small icon with label to the right.

• List: Items appear as a small icon with label to the right.

— Arrayed in a columnar, sorted layout.

• Report: Items appear as a line in a multicolumn format.

— Leftmost column includes icon and its label.

— Subsequent columns include application-specific information.

■ Purpose and usage:

— Where the representation of objects as icons is appropriate.

— To represent items with multiple columns of information.

4. Drop-down/Pop-up List Boxes

■ Description

— A single rectangular control that shows one item with a small button to the right side.

• The button provides a visual cue that an associated selection box is available but hidden.

— When the button is selected, a larger associated box appears, containing a list of choices from which one may be selected.

— Selections are made by using the mouse to point and click.

— Text may not be typed into the control.

■ Purpose:

— To select one item from a large list of mutually exclusive options when screen space is limited.

■ Advantages:

— Unlimited number of choices.

— Reminds users of available options.

— Conserves screen space.

■ Disadvantages:

— Requires an extra action to display the list of choices.

— When displayed, all choices may not always be visible, requiring scrolling.

— The list may be ordered in an unpredictable way, making it hard to find items.

■ Proper usage:

— For selecting values or setting attributes.

— For choices that are mutually exclusive (only one can be selected).

— Where screen space is limited.

— For data and choices that are:

• Best represented textually.

• Infrequently selected.

• Not well known, easily learned, or remembered.

• Ordered in a unpredictable fashion.

• Large in number.

• Variable or fixed in list length.

— Use drop-down/pop-up lists when:

• Screen space or layout considerations make radio buttons or single-selection list boxes impractical.

• The first, or displayed, item will be selected most of the time.

— Do not use a drop-down list if it important that all options be seen together.

Country:	United States	±
Language:	English (American)	±
Keyboard Layout:	US	±
Measurement:	English	±

Fig: Drop-down list boxes.
opened for Country.

Fig : Drop-down list box
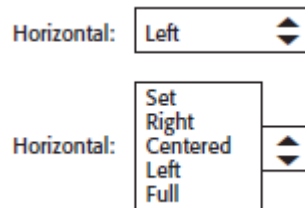

Fig : Pop-up list box, closed and opened.

Prompt Button
■ Provide a visual cue that a box is hidden by including a downward pointing arrow, or other meaningful image, to the right side of the selection field.
— Position the button directly against, or within, the selection field.

Selection Descriptions
■ Clearly and meaningfully describe the choices available. Spell them out as fully as possible.
— Graphical representations must clearly represent the options.
— Left-align them in columns.
— Display the descriptions using mixed-case letters.
List Size
■ Not limited in size.
■ Present all available alternatives.
Box Size
■ Long enough to display 6 to 8 choices without scrolling.
— If more than eight choices are available, provide vertical scrolling to display all items.
■ Wide enough to display the longest possible choice.
■ When a box cannot be made wide enough to display the longest entry:
— Make it wide enough to permit entries to be distinguishable, or,
— Break long entries with ellipses (...) in the middle, or,
— Provide horizontal scrolling.
Organization
■ Order in a logical and meaningful way to permit easy browsing.
■ If a particular choice is not available in the current context, omit it from the list.
— Exception: If it is important that the existence and unavailability of a particular list item be communicated, display the choice dimmed or grayed out instead of deleting it.
Layout and Separation
■ Enclose the choices in a box composed of a solid line border.
— The border should be the same color as the choice descriptions.
— Leave one blank character position between the choices and the left border.
— Leave one blank character position between the longest choice description in the list and the right border, if possible.
Captions
■ Display using mixed-case letters.
■ Position the caption to the left of the box.
— Alternately, it may be positioned left-justified above the box.
Defaults
■ When the drop-down/pop-up listing is first presented, display the currently set value.
■ If a choice has not been previously selected, provide a default choice.

Disabling
■ When a drop-down/pop-up list box is disabled, display its caption and entries as disabled or dimmed.
Selection Method and Indication
■ Pointing:
— Highlight the selection choice in some visually distinctive way when the pointer or cursor is resting on it and the choice is available for selection.
■ Activation:
— Close the drop-down/pop-up list box when an item is selected.

5. Palettes

■ Description:
— A control consisting of a series of graphical alternatives. The choices themselves are descriptive, being composed of colors, patterns, or images.
— In addition to being a standard screen control, a palette may also be presented on a pull-down or pop-up menu or a toolbar.
■ Purpose:
— To set one of a series of mutually exclusive options presented graphically or pictorially.
■ Advantages:
— Pictures aid comprehension.
— Easy-to-compare choices.
— Usually consume less screen space than textual equivalents.
■ Disadvantages:
— A limited number of choices can be displayed.
— Difficult to organize for scanning efficiency.
— Requires skill and time to design meaningful and attractive graphical representations.
■ Proper usage:
— For setting attributes, properties, or values.
— For mutually exclusive choices (that is, only one can be selected).
— Where adequate screen space is available.
— Most useful for data and choices that are:
• Discrete.
• Frequently selected.
• Limited in number.
• Variable in number.
• Not easily remembered.
• Most easily understood when the alternatives may be seen together and compared to one another.
• Most meaningfully represented pictorially or by example.
• Can be clearly represented pictorially.
• Rarely changed in content.
— Do not use:
• Where the alternatives cannot be meaningfully and clearly represented pictorially.
• Where words are clearer than images.
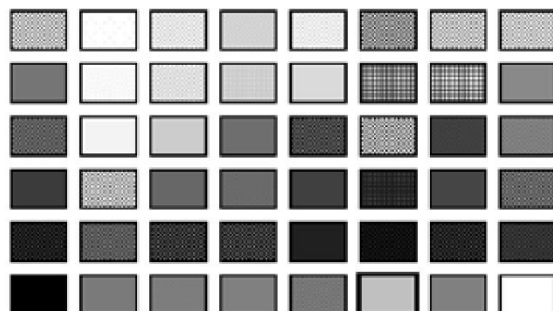• Where the choices are going to change.

Fig : Palette

Graphical Representations
■ Provide meaningful, accurate, and clear illustrations or representations of choices.

■ Create images large enough to:
— Clearly illustrate the available alternatives.
— Permit ease in pointing and selecting.
■ Create images of equal size.
■ Always test illustrations before implementing them.
Size
■ Present all available alternatives within the limits imposed by:
— The size of the graphical representations.
— The screen display's capabilities.
Layout
■ Create boxes large enough to:
— Effectively illustrate the available alternatives.
— Permit ease in pointing and selecting.
■ Create boxes of equal size.
■ Position the boxes adjacent to, or butted up against, one another.
■ A columnar orientation is the preferred manner.
■ If vertical space on the screen is limited, orient the choices horizontally.
Organization
■ Arrange palettes in expected or normal order.
— For palettes arrayed top to bottom, begin ordering at the top.
— For palettes arrayed left to right, begin ordering at the left.
■ If an expected or normal order does not exist, arrange choices by frequency of occurrence,
sequence of use, importance, or alphabetically (if textual).
■ If, under certain conditions, a choice is not available, display it subdued or less
brightly than the other choices.
Captions
■ Provide a caption for each palette.
— On screens containing only one palette, the screen title may serve as the caption.
■ Display the caption fully spelled out using mixed-case letters.
■ Columnar orientation:
— The field caption may be positioned left-aligned above the palette.

Shade:

— Alternately, the caption may be positioned to the left of the topmost alternative.

Shade:

■ Horizontal orientation:
— The field caption may be positioned above the palette.

Shade:

— Alternately, the caption may be positioned to the left of the alternatives.

Shade:

Selection Method and Indication
■ Pointing:
— Highlight the choice in some visually distinctive way when the pointer or cursor is resting on it and
the choice is available for selection.
■ Activation:
— When a choice is selected, distinguish it visually from the unselected choices by highlighting it in

a manner different from when it is pointed at, or by placing a bold border around it.

■ Defaults:

— If a palette is displayed with a choice previously selected or a default choice, display the currently active choice in the manner used when it was selected.

Pointing. The selection target should be as large as possible in order to make it easy to move to. Highlight the selection choice in some visually distinctive way when the pointer or cursor is resting on it and the choice is available for selection.

Activation. When a choice is selected, distinguish it visually from the unselected choices by highlighting it in a manner different from when it is pointed at, or by placing a bolder border around it.

Defaults. If a palette is displayed with a choice previously selected or a default choice, display the currently active choice in the manner used when it was selected.

Q. 5 Explain the purpose of the prototypes.

Discuss any two kinds of prototypes with their importance to the system developer

PROTOTYPE:

A prototype is primarily a vehicle for exploration, communication, and evaluation. Its purpose is to obtain user input in design, and to provide feedback to designers. Its major function is the communicative role it plays, not accuracy or thoroughness. A prototype enables a design to be better visualized and provides insights into how the software will look and work. It also aids in defining tasks, their flow, the interface itself, and its screens.

A prototype is a simulation of an actual system that can be quickly created. A prototype may be a rough approximation, such as a simple hand-drawn sketch, or it may be interactive, allowing the user to key or select data using controls, navigate through menus, retrieve displays of data, and perform basic system functions. A prototype need not be perfectly realistic, but it must be reasonably accurate and legible. A prototype also need not be functionally complete, possessing actual files or processing data. Today, many software support tools for prototyping are available that permit the prototype to be integrated directly into the application code.

Hand Sketches and Scenarios

Description:

— Screen sketches created by hand.

— Focus is on the design, not the interface mechanics.

— A low-fidelity prototype.

■ Advantages:

— Can be used very early in the development process.

— Suited for use by entire design team.

— No large investment of time and cost.

— No programming skill needed.

— Easily portable.

— Fast to modify and iterate.

— A rough approximation often yields more substantive critical comments.

— Easier to comprehend than functional specifications.

— Can be used to define requirements.

■ Disadvantages:

— Only a rough approximation.

— Limited in providing an understanding of navigation and flow.

— A demonstration, not an exercise.

— Driven by a facilitator, not the user.

*Sketch Creation Process**

Sketch (storyboard) the screens while determining:

— The source of the screen's information.

— The content and structure of individual screens.

— The overall order of screens and windows.

Use an erasable medium.

■ Sketch the screens needed to complete each workflow task.

■ Try out selected metaphors and change them as necessary.

■ First, storyboard common/critical/frequent scenarios.

— Follow them from beginning to end.

— Then, go back and build in exceptions.

■ Don't get too detailed; exact control positioning is not important, just overall order

and flow.

■ Storyboard as a team, including at least one user.

■ Only develop online prototypes when everyone agrees that a complete set of screens

has been satisfactorily sketched

Interactive Paper Prototypes

Description:

— Interface components (menus, windows, and screens) constructed of common

paper technologies (Post-It notes, transparencies, and so on).

— The components are manually manipulated to reflect the dynamics of the software.

— A low-fidelity prototype.

■ Advantages:

— More illustrative of program dynamics than sketches.

— Can be used to demonstrate the interaction.

— Otherwise, generally the same as for hand-drawn sketches and scenarios.

■ Disadvantages:

— Only a rough approximation.

— A demonstration, not an exercise.

— Driven by a facilitator, not the user.

— Limited usefulness for usability testing.

Programmed Facades

■ Description:

— Examples of finished dialogs and screens for some important aspects of the system.

— Created by prototyping tools.

— Medium-fidelity to high-fidelity prototypes.

■ Advantages:

— Provide a good detailed specification for writing code.

— A vehicle for data collection.

■ Disadvantages:

— May solidify the design too soon.

— May create the false expectation that the "real thing" is only a short time away.

— More expensive to develop.

— More time-consuming to create.

— Not effective for requirements gathering.

— Not all of the functions demonstrated may be used because of cost, schedule limitations,

or lack of user interest.

— Not practical for investigating more than two or three approaches.

Prototype-Oriented Languages

■ Description:

— An example of finished dialogs and screens for some important aspects of the

system.

— Created through programming languages that support the actual programming

process.

— A high-fidelity prototype.

■ Advantages:

— May include the final code.

— Otherwise, generally the same as those of programmed facades.

■ Disadvantages:

— Generally the same as for programmed facades.

Q. 6 Explain the following kinds of tests.

a.   **Think Aloud Evaluation b) Usability Test c) Heuristic Evaluation (or) Cognitive Walkthroughs**

A test is a tool that is used to measure something. The "something" may be:

- Conformance with a requirement.
- Conformance with guidelines for good design.
- Identification of design problems.
- Ease of system learning.
- Retention of learning over time.
- Speed of task completion.
- Speed of need fulfillment.

- Error rates.
- Subjective user satisfaction.

A test is usually formal; it is created and applied intentionally and with a purpose. It is usually based upon some kind of criteria, an understanding of what a good result would be. Several testing techniques, at varying levels of sophistication and cost, are available to exercise the system.

Think-Aloud Evaluations

Description:
— Users perform specific tasks while thinking out load.
— Comments are recorded and analyzed.
■ Advantages:
— Utilizes actual representative tasks.
— Provides insights into the user's reasoning.
■ Disadvantages:
— May be difficult to get users to think out loud.
■ Guidelines:
— Develop:
• Several core or representative tasks.
• Tasks of particular concern.
— Limit session to 60 to 90 minutes.

Usability Test

■ Description:
— An interface evaluation under real-world or controlled conditions.
— Measures of performance are derived for specific tasks.
— Problems are identified.
■ Advantages:
— Utilizes an actual work environment.
— Identifies serious or recurring problems.
■ Disadvantages:
— High cost for establishing facility.
— Requires a test conductor with user interface expertise.
— Emphasizes first-time system usage.
— Poorly suited for detecting inconsistency problems.

Heuristic Evaluation Effectiveness

Severity Ratings in Heuristic Evaluation
0 = I don't agree that this is a usability problem at all.
1 = A cosmetic problem only. Need not be fixed unless extra time is available.
2 = A minor usability problem. Fixing should be given a low priority.
3 = A major usability problem. Important to fix and should be given a high priority.
4 = A usability catastrophe. Imperative to fix before the product can be released.

Research-Based Set of Heuristics
1. Automate unwanted workload.
• Free cognitive resources for high-level tasks.
• Eliminate mental calculations, estimations, comparisons, and unnecessary thinking.
2. Reduce uncertainty.
• Display data in a manner that is clear and obvious.
3. Fuse data.
• Reduce cognitive load by bringing together lower-level data into a higher-level

summation.

4. Present new information with meaningful aids to interpretation.
• Use a familiar framework, making it easier to absorb.
• Use everyday terms, metaphors, and so on.

5. Use names that are conceptually related to functions.
• Context-dependent.
• Attempt to improve recall and recognition.

6. Group data in consistently meaningful ways to decrease search time.

7. Limit data-driven tasks.
• Reduce the time needed to assimilate raw data.
• Make appropriate use of color and graphics.

8. Include in the displays only that information needed by a user at a given time.
• Allow users to remain focused on critical data.
• Exclude extraneous information that is not relevant to current tasks.

9. Provide multiple coding of data where appropriate.

10. Practice judicious redundancy.
• To resolve the conflict between heuristics 6 and 8.

Possible Web Page Heuristics
1. Speak the user's language.
• Use familiar words, phrases, and concepts.
• Present information in a logical and natural order.

2. Be consistent.
• Indicate similar concepts through identical terminology and graphics.
• Adhere to uniform conventions for layout, formatting, typefaces, labeling, and so on.

3. Minimize the user's memory load.
• Take advantage of recognition rather than recall.
• Do not force users to remember key information across documents.

4. Build flexible and efficient systems.
• Accommodate a range of user sophistication and diverse user goals.
• Provide instructions where useful.
• Lay out screens so that frequently accessed information is easily found.
'

Cognitive Walkthroughs

Description:
— Reviews of the interface in the context of tasks users perform.
■ Advantages:
— Allow a clear evaluation of the task flow early in the design process.
— Do not require a functioning prototype.
— Low cost.
— Can be used to evaluate alternate solutions.
— Can be performed by developers.
— More structured than a heuristic evaluation.
— Useful for assessing "exploratory learning."
■ Disadvantages:
— Tedious to perform.
— May miss inconsistencies and general and recurring problems.
■ Guidelines:
— Needed to conduct the walkthrough are:
• A general description of proposed system users and what relevant knowledge they possess.
• A specific description of one or more core or representative tasks to be performed.
• A list of the correct actions required to complete each of the tasks.

— Review:
• Several core or representative tasks across a range of functions.
• Proposed tasks of particular concern.
— Developers must be assigned roles of:
• Scribe to record results of the action.
• Facilitator to keep the evaluation moving.
— Start with simple tasks.
— Don't get bogged down demanding solutions.
— Limit session to 60 to 90 minutes.

Faculty Signature   CCI Signature   HOD Signature