

Interaction Design

Form Fill-In

P.VISWANATHAN

pviswanathan@vit.ac.in

04-DEC-2002

Key references/literature:

W.O. Galitz (1989) Handbook of screen format design. QED Information Sciences Inc.

chapter 5: data entry screens.

chapter 6: inquiry screens.

D.J. Mayhew (1992) Principles and guidelines in software user interface design. Prentice Hall.

chapter 5: dialog styles - fill-in forms.

ISO/FDIS 9241 (1997) Ergonomic requirements for office work with visual display terminals (VDTs).

Part 12: presentation of information.

Fill-In Forms: example

STUDENT REGISTRATION				
STUDENT				
NAME: -----		* MAJOR: ---	YR.: ----	
ID: --- --		TRACK: -	STA.: ---	
COURSES				
NO.	* TITLE	SEC.	SEQ.	
---	---	---	---	
---	---	---	---	
---	---	---	---	
---	---	---	---	

LOAN APPLICATION	
NAME: ■ -----	(Last, First, Middle)
ADDRESS: -----	(Street, City, State, Zip)
DATE: ----- (MM/DD/YY)	AMOUNT: ----- (XXXX.XX)
RATE: --,-- (XX.XX)	# MONTHS: --- (XXX)
<i>Press ARROWS to move cursor</i> <i>Press ENTER to accept input</i>	
MicroHelp: ex.: Mayhew, Deborah J. ---	

Fill-In Forms: when to use (1)

- User is presented with a display resembling a paper form and enters data into fields

Advantages:

- Simplifies data entry, reduces need for manuals
- Requires modest training
- makes both semantics and syntax explicit

Issues:

- movement around the form
- input validation and correction

Fill-In Forms: when to use (2)

Advantages:

- requires little memory
 - recognition versus recall
- efficient use of screen ‘real estate’
- accommodates parameters with many possible values
- provides context of use
- enhancements are visible

Fill-In Forms: when to use (3)

Disadvantages:

- assumes knowledge of valid inputs (semantic knowledge)
- assumes typing skill and knowledge of special keys (e.g. TAB, RETURN, BACKSPACE)
- required type-in creates opportunities for user error
- inflexible

Fill-In Forms: when to use (4)

Most appropriate for:

- user characteristics:
 - negative or neutral attitude
 - low or moderate motivation
- knowledge and experience
 - moderate to high typing skills
 - little to moderate system experience
 - moderate to high task experience
 - low to moderate application experience
 - moderate to frequent use of other systems
 - moderate to high computer literacy
- job and task characteristics
 - moderate to high frequency of use
 - little or no training
 - discretionary use
 - low to moderate turnover rate
 - other systems with paper forms
 - moderate task importance
 - high task structure

Fill-In Forms: navigation (1)

Empirical comparison of two methods for navigation
between fields

‘manual tabbing’ versus ‘auto skip’

results:

“manual tabbing resulted in faster performance and fewer
keying errors”

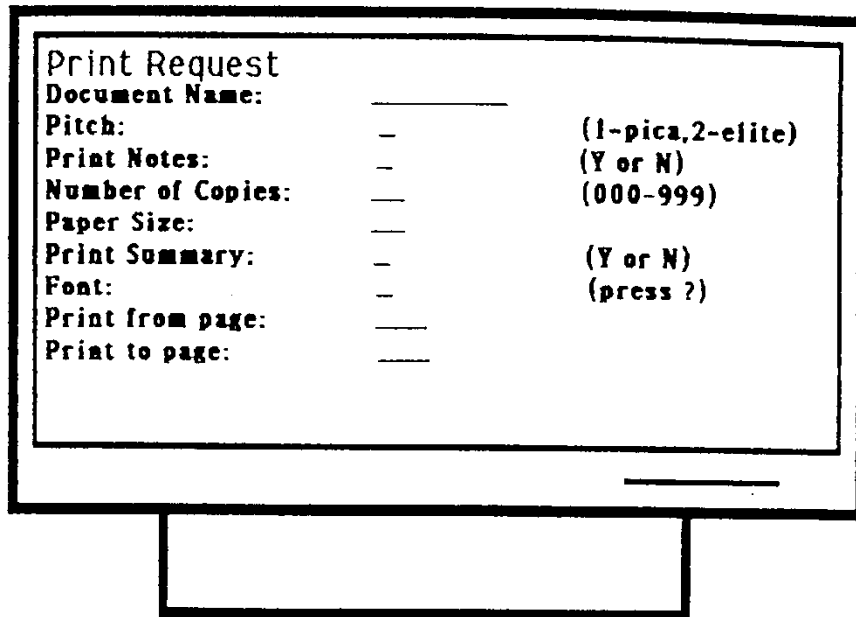
[taken from W.O. Galitz (1972) IBM 3270 on-line evaluation. INA technical report E5320-A02/M72-001, January 20, 1972]

Fill-In Forms: layout (1)

- Design and organise the form to support the task (e.g. data entry from paper form versus customer support or phone order);
- organise groups of items related semantically, by order of use, by frequency of use, and/or importance;
- keep related and interdependent items on the same screen;
- minimise number of screens for high frequency users/slow system response time; maximise screen clarity for infrequent users/fast system response time;
- use white space to create balance and symmetry, and lead the eye in the appropriate direction;
- separate logical groups by spaces, lines, colour or other visual cues.

Fill-In Forms: layout (2)

poor



Print Request

Document Name: _____

Pitch: _____ (1-pica,2-elite)

Print Notes: _____ (Y or N)

Number of Copies: _____ (000-999)

Paper Size: _____

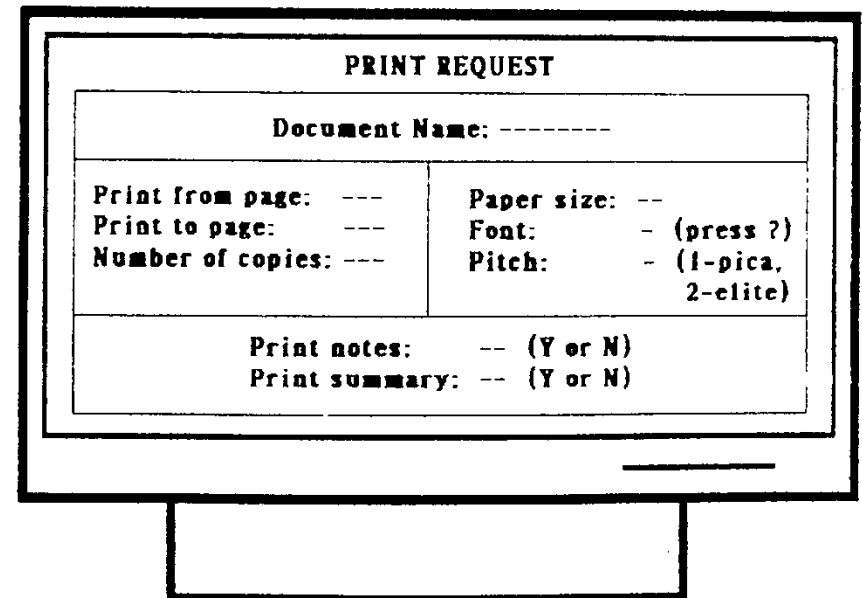
Print Summary: _____ (Y or N)

Font: _____ (press ?)

Print from page: _____

Print to page: _____

improved



PRINT REQUEST	
Document Name: _____	
Print from page: ---	Paper size: --
Print to page: ---	Font: -- (press ?)
Number of copies: ---	Pitch: -- (1-pica, 2-elite)
Print notes: -- (Y or N)	
Print summary: -- (Y or N)	

Fill-In Forms: captions & fields (1)

- Left justify alpha fields; left justify numeric input fields, but right justify or decimal-align numeric fields upon display; right justify captions when data entry is from a source document; otherwise left justify captions, unless captions vary considerably in length.
- Separate the (longest) caption (in a right justified column) from its field by one space (following the delimiter).
- Separate one caption-field group from another by three spaces horizontally, one line vertically.
- Break up long columnar fields or long columns of single-field items into groups of five separated by a blank line.
- Provide distinctive field group and section headings in complex forms.
- For single fields, place the caption to the left; for list fields, place caption above, left justified.

Fill-In Forms: captions & fields (2)

- Distinguish captions and fields with a visual cue (bold versus plain, upper versus lower case) and a delimiter (e.g. colon); highlight fields when a data entry is from a source document; otherwise highlight captions (but be consistent if a system contains both kinds of forms).
- Captions should be brief, familiar and descriptive; use abbreviations and contractions when data entry is from a source document; otherwise spell out captions in full; avoid computer jargon, but exploit user jargon; use consistent terminology.
- Indicate the number of character spaces in a field with underscores (versus column separators, brackets or reverse video) which are overstruck when data is entered.
- Indicate when fields are optional.

Fill-In Forms: captions & fields (3)

Four types of input field indicators

<i>Performance measure</i>	Dot column separator	Reverse image	Broken underline	brackets
Field length estimation (errors)	4.5	36.5	1.8	59.5
Field length estimation (seconds)	3.9	3.2	3.0	3.4
Finding block cursor (errors)	0.3	0.2	0.1	0.2
Finding block cursor (seconds)	2.4	3.2	2.3	2.4
Readability (errors)	1.8	2.7	1.7	1.7

[taken from R.E. Savage, J.K. Habinek & N.J. Blackstad (1982) An experimental evaluation of input field and cursor combinations. Proceedings of the Human Factors Society 26th Annual Meeting, pp. 629-633]

Fill-In Forms: captions & fields (4)

poor

Name:.....		Major:..	Year: Status:
ID:.....		Coop Track:...
Number	Title	Section	Sequence
.....
.....
.....
.....
.....
.....

improved

STUDENT REGISTRATION				
STUDENT				
NAME: -----		* MAJOR: ---	YR.: ----	
ID: --- -- ----		TRACK: -	STA.: ---	
COURSES				
NO.	* TITLE	SEC.	SEQ.	
---	---	-----	--	--
---	---	-----	--	--
---	---	-----	--	--
---	---	-----	--	--
---	---	-----	--	--

Fill-In Forms: input formats (1)

- Make high frequency inputs easy to express (.e.g Y/N, OO, defaults).
- Let the user specify the unit of measurement; don't require transformations or calculations!
- Provide meaningful groupings to break up long input formats.
- Allow abbreviated input when it can be unambiguously interpreted (Yes or Ye or Y).
- System should be 'case blind' when it really doesn't matter (YES or Yes or yes).
- Provide defaults whenever possible; allow simple (single key) acceptance of defaults; decide whether system, session or file defaults are appropriate; make it clear which is which, if systems provides both kinds of different contexts.

Fill-In Forms: input formats (2)

- Keep input fields short if possible.
- Do not combine letters and numbers in a single field.
- Avoid frequent shifts between upper and lower case characters.
- Avoid uncommon letter sequences ([‘an’ or ‘th’] versus [‘gp’ or ‘xz’]).
- Do not require leading zeroes.

Fill-In Forms: input formats (3)

Error percentages for varying field lengths

Average number characters per field	Percent of fields in error
3	1.4
5	2.0
7	2.6
9	3.1
11	3.6

Frequency of character substitutions

Character pairs	Cumulative frequency
I - 1	25 %
O - 0	50 %
B - 8	60 %
Z - 2	70 %

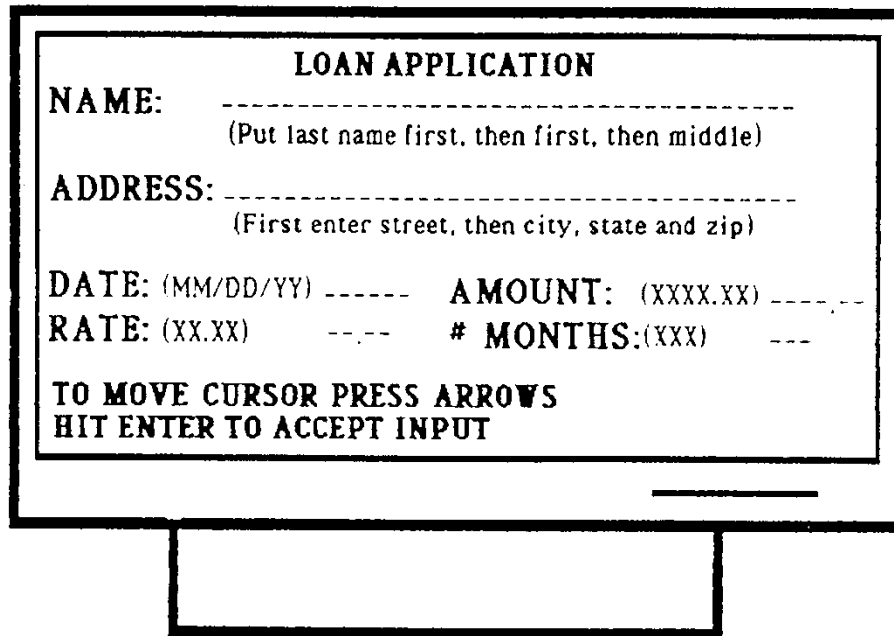
[taken from W.O. Galitz (1985) Handbook of screen design. (pp. 74-75); Wellesley Hills, QED Information Sciences, Inc.]

Fill-In Forms: prompts & instructions (1)

- Prompts should be brief and unambiguous; use examples when possible to supplement formalisms; provide ranges of possible values when appropriate and possible.
- Place prompts to the right of fields, on in a 'Micro Help' line at the bottom of the screen.
- Provide instructions for navigation and completion on the screen, or through on-line help.
- Place instructions in a consistent location across screens, and make them visually distinctive.
- Use consistent terminology and consistent grammatical form and style in instructions.

Fill-In Forms: prompts & instructions (2)

poor



LOAN APPLICATION

NAME: -----
(Put last name first, then first, then middle)

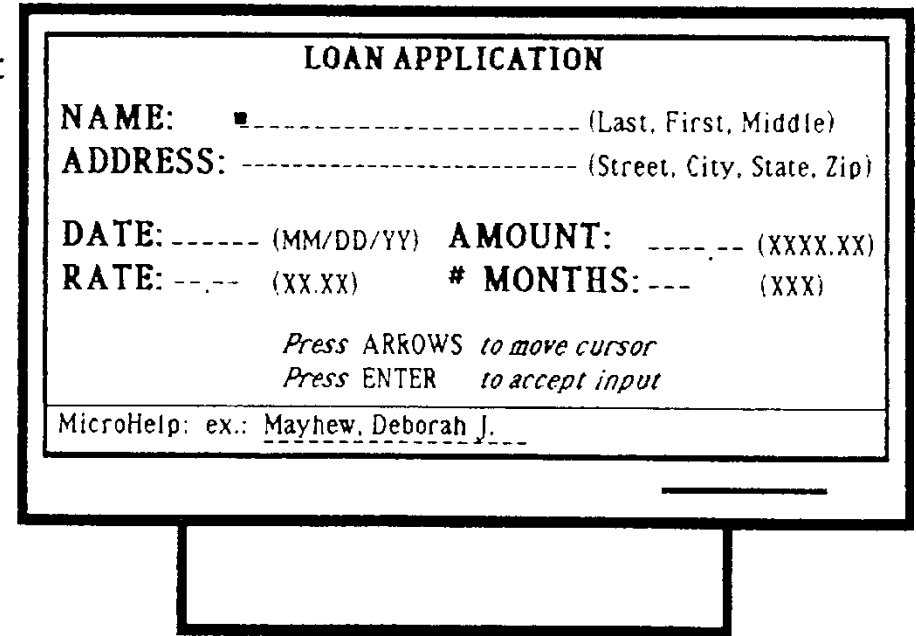
ADDRESS: -----
(First enter street, then city, state and zip)

DATE: (MM/DD/YY) ----- **AMOUNT:** (XXXX.XX) -----

RATE: (XX.XX) ----- **# MONTHS:** (XXX) -----

TO MOVE CURSOR PRESS ARROWS
HIT ENTER TO ACCEPT INPUT

improved



LOAN APPLICATION

NAME: ■ ----- (Last, First, Middle)

ADDRESS: ----- (Street, City, State, Zip)

DATE: ----- (MM/DD/YY) **AMOUNT:** ----- (XXXX.XX)

RATE: ----- (XX.XX) **# MONTHS:** ----- (XXX)

Press ARROWS to move cursor
Press ENTER to accept input

MicroHelp: ex.: Mayhew, Deborah J.

Fill-In Forms: navigation (1)

- Direct manipulation increases flexibility, speed and easy of learning for navigation through fields.
- Allow forward and backward movement by field and within fields; in complex forms, provide a hierarchy of navigation techniques (e.g. next character, next word, next field, next field group, next section, next form).
- Do not use autotab unless fields have fixed lengths, and users are high frequency and experienced.
- Provide titles and page numbers or place markers on screen in a multi-screen form.

Fill-In Forms: navigation (2)

poor

POLICY APPLICATION

APPLICANT	VEHICLE
NAME: _____	YEAR: _____
POLICY #: _____	MAKE: _____
	NEW/USED: -
DRIVER	
NAME: _____	LIENHOLDER
BIRTHDATE: _____	VEHICLE #: _____
MARITAL STATUS: -	NAME: _____
GENDER: -	ADDRESS: _____

Press TAB to move cursor forward by field

improved

POLICY APPLICATION ** Page 1 of 4 **

APPLICANT	VEHICLE
NAME: _____	YEAR: _____
POLICY #: _____	MAKE: _____
	TYPE: <input type="radio"/> NEW <input checked="" type="radio"/> USED
DRIVER	
NAME: _____	LIENHOLDER
BIRTHDATE: _____	VEHICLE #: _____
MARITAL ST.: <input type="radio"/> M <input type="radio"/> S	NAME: _____
GENDER: <input type="radio"/> M <input type="radio"/> F	ADDRESS: _____

Press TAB to move cursor forward by field
Press HELP for additional navigation commands

Fill-In Forms: error handling (1)

- Allow character edits in fields (versus re-type whole field).
- Place cursor in error field after error detection; highlight error field if possible.
- For independent fields, withhold error reporting until user request.
- Provide semantic and syntactic information in error messages depending on user knowledge.

Fill-In Forms: error handling (2)

poor

LOAN APPLICATION

NAME: Mayhew, Deborah, J.-- (Last, First, Middle)
ADDRESS: 1 C St., Town, MA, 00000 (Street, City, State, Zip)
DATE: 2/I/87 (MM/DD/YY) **AMOUNT:** 3500.00 (XXXX.XX)
RATE: 11.5- (XX.XX) **# MONTHS:** 36- (XXX)

Press ARROWS to move cursor
Press ENTER to accept input

Illegal Date

improved

LOAN APPLICATION

NAME: Mayhew, Deborah, J.-- (Last, First, Middle)
ADDRESS: 1 C St., Town, MA, 00000 (Street, City, State, Zip)
DATE: 2/I/87 (MM/DD/YY) **AMOUNT:** 3500.00 (XXXX.XX)
RATE: 11.5- (XX.XX) **# MONTHS:** 36- (XXX)

Press ARROWS to move cursor
Press ENTER to accept input

****ERROR**** Alph characters not accepted in DATE field

Layout of the Form (1)

- Not too much on a Form
 - Split, logically, over several Forms
 - hide/reveal controls
- Information in centre of visual field is most likely to be seen.
 - Put important info in obvious positions

Layout of the Form (2)

- Arrange controls in a logical sequence
 - especially for data entry
 - work from left - right, top-bottom
 - consistency of layout over Forms
 - e.g. “exit” button in same position on all Forms
- Arrange order using “Tab-key”
- Set Focus after major operation
 - e.g. after pressing a Command button
 - when loading/returning to a Form
 - clear text boxes on data entry forms?

Form fill-in (1)

- good in appropriate applications
- easy, little training required
- can be used to interface with query languages
 - QBE [query by example]

Form fill-in: example


Microsoft Access

File Edit View Insert Format Records Tools Window Help Show Me

Categories

Category Name: Beverages

Description: Soft drinks, coffees, teas, beers, and ales

Picture: 

Product Name:	Chai	<input type="checkbox"/> Discontinued
Quantity Per Unit:	10 boxes x 20 bags	Unit Price: \$18.00
Product Name:	Chang	<input type="checkbox"/> Discontinued
Quantity Per Unit:	24 - 12 oz bottles	Unit Price: \$19.00

Record: 1 of 8

A picture representing the food category.

Form fill-in (2)

- Principle: user's main task is to provide data in labeled fields clustered in one or more screens
- Data
 - Binary choice (Yes/No, Female/Male)
 - Selection from brief list (days of week, set of colours)
 - Large domain (personal names, chemical formulae)
 - Essential unbounded (exploratory paragraphs, meteorological data)
- Can replace form fill-in with a series of menu choices -> however may become extremely tedious

Form fill-in (3)

- Advantages:
 - simplifies data entry
 - Requires modest training
 - Shows context for activity
 - Permits form dialog management tools
- Disadvantages:
 - Consumes screen space
 - Requires typing skills

Form fill-in (4)

- resembles familiar paper forms
- simplifies data entries with pre-defined structures
- gives convenient assistance
- problems
 - visual layout and organization
 - screen space

Screen Design and Layout (1)

- Presenting information:
 - match presentation to purpose
 - e.g.,
 - sort file listing by name, or by date, or ...
 - alignment of text versus numbers
- Entering information:
 - use clear, logical layout of form fields
 - requires task analysis
 - alignment is important

Screen Design and Layout (2)

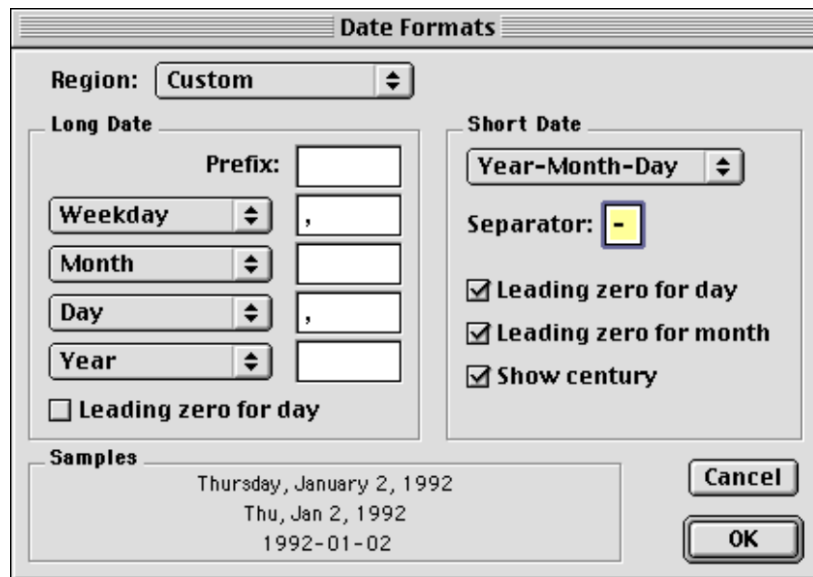
- Aesthetics and utility:
 - beauty versus utility
 - a pretty interface is not necessarily a good interface
 - still, good graphic design can increase user satisfaction
- Knowing what to do:
 - style guides
 - platform or company standards for the design of user interfaces
 - e.g., Macintosh Human Interface Guidelines

Screen Design and Layout (3)

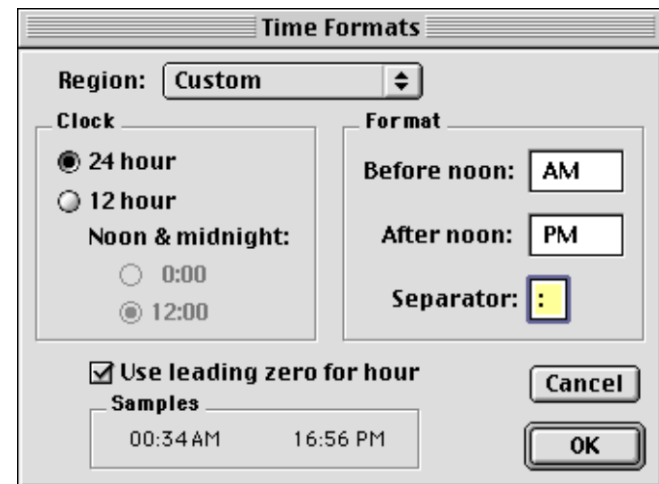
- Knowing what to do:
 - affordances
 - “where do I click?”
 - elements should suggest, by their shape and other attributes, what you can do with them
 - e.g., a button affords pushing
- Localization/internationalization:
 - change of language for text
 - alignment and layout
 - date formats
 - time formats
 - number formats

Screen Design and Layout (4)

- Date/time formats:



The 'Date Formats' dialog box is titled 'Date Formats'. It features a 'Region:' dropdown menu set to 'Custom'. The 'Long Date' section includes a 'Prefix:' text field, a 'Weekday' dropdown, a 'Month' dropdown, a 'Day' dropdown, a 'Year' dropdown, and a checkbox for 'Leading zero for day'. The 'Short Date' section includes a 'Year-Month-Day' dropdown, a 'Separator:' dropdown set to '-', and three checkboxes: 'Leading zero for day', 'Leading zero for month', and 'Show century'. A 'Samples' text area at the bottom displays 'Thursday, January 2, 1992', 'Thu, Jan 2, 1992', and '1992-01-02'. 'Cancel' and 'OK' buttons are located at the bottom right.

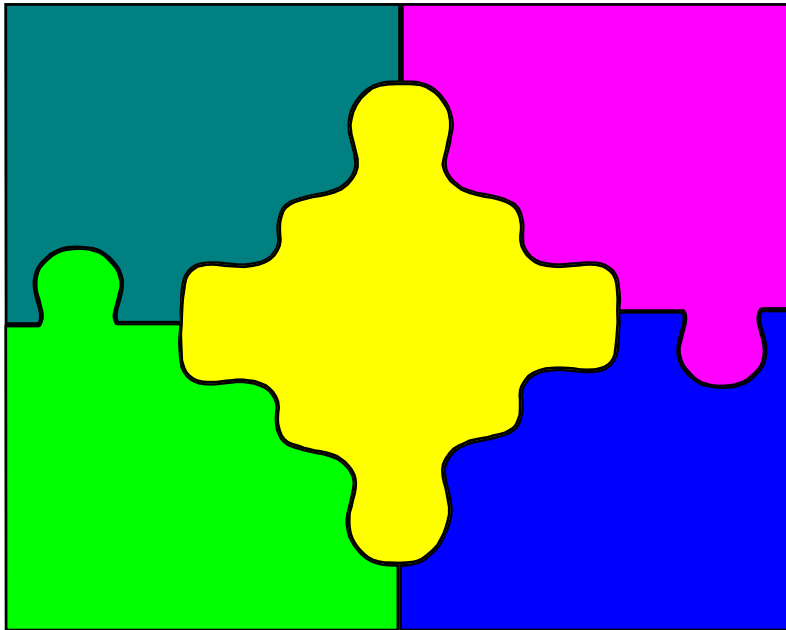


The 'Time Formats' dialog box is titled 'Time Formats'. It features a 'Region:' dropdown menu set to 'Custom'. The 'Clock' section includes radio buttons for '24 hour' (selected) and '12 hour', and a 'Noon & midnight:' section with radio buttons for '0:00' and '12:00' (selected). The 'Format' section includes text fields for 'Before noon:' (AM) and 'After noon:' (PM), and a 'Separator:' dropdown set to ':'. A checkbox for 'Use leading zero for hour' is checked. A 'Samples' text area at the bottom displays '00:34 AM' and '16:56 PM'. 'Cancel' and 'OK' buttons are located at the bottom right.

Visual Display Elements to Consider

- Color
 - Can have unwanted emotional meaning
 - Incorrectly connect parts of the interface
- Typography
 - Affects legibility, understanding

Organization of Screen Elements



- Balance
- Symmetry
- Regularity
- Predictability
- Sequentiality
- Economy
- Unity
- Proportion
- Simplicity
- Groupings

Balance

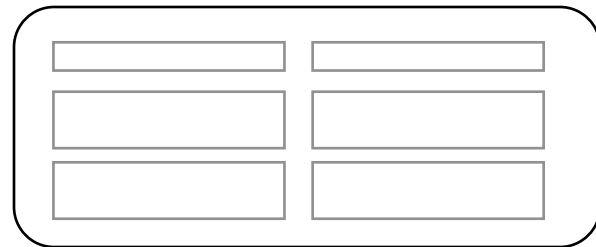
- Equal weight of screen elements
 - Left to right, top to bottom

Balance

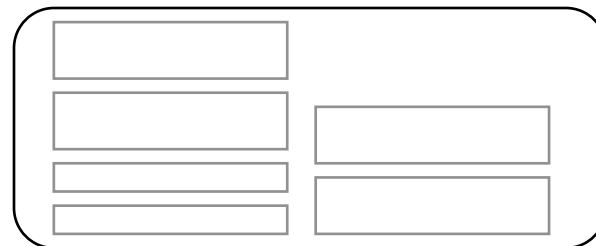
Unstable

Balance

- Left column processed
- Right column noted
as same



- Both columns need to
be understood by
visual processing
system



Symmetry

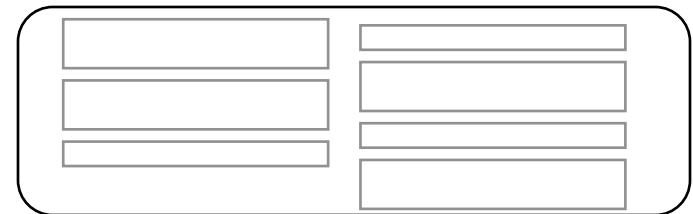
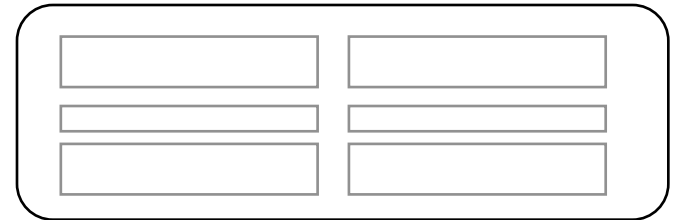
- Replicate elements left and right of the center line

Symmetric

Asymmetric

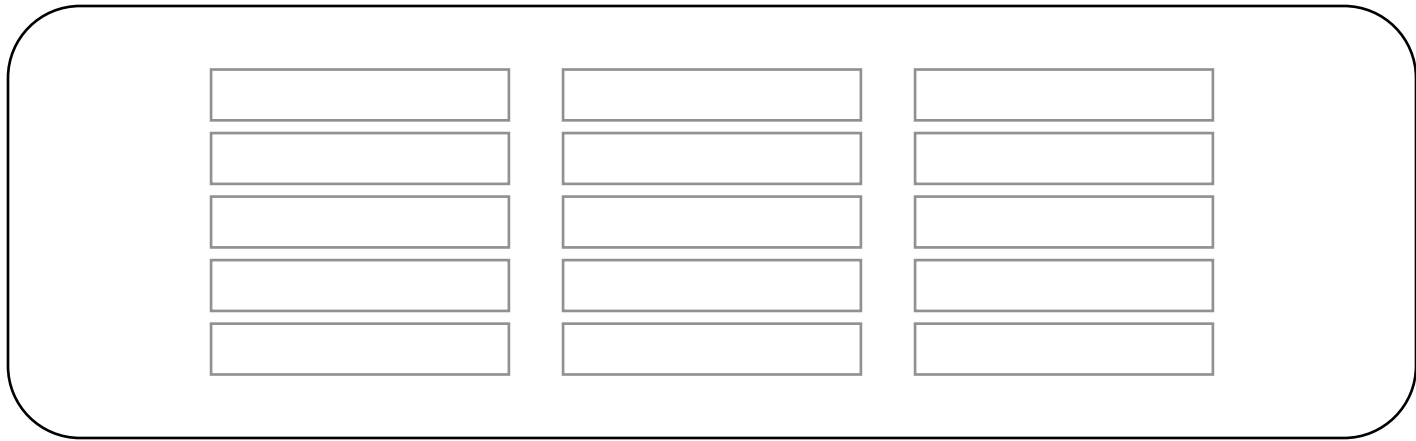
Symmetry

- Left column processed - Right column noted as same
- Both right & left columns processed plus relationship of right to left

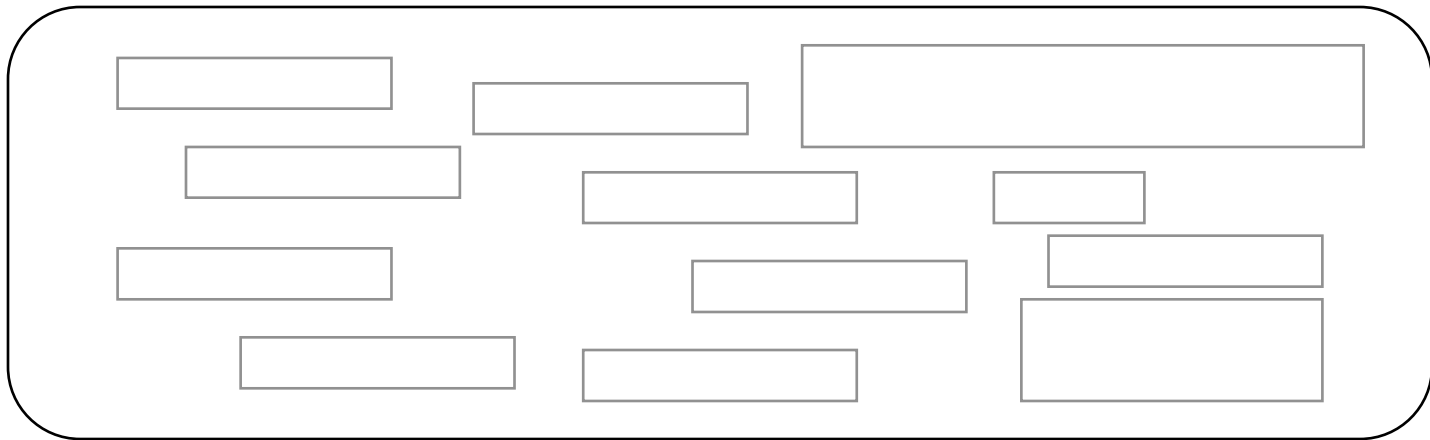


Regularity

- Create standard and consistent spacing on horizontal and vertical alignment points



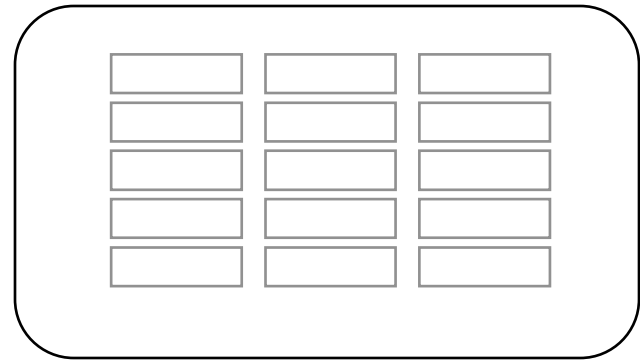
Regular



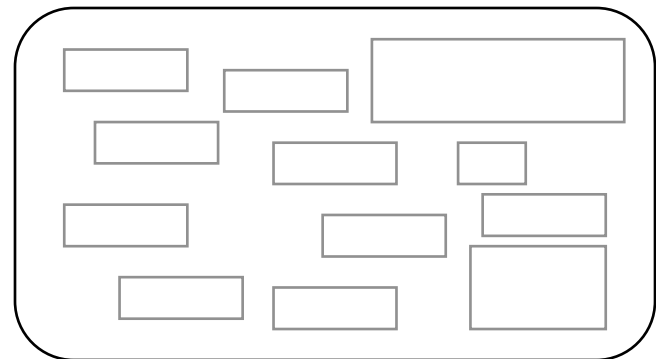
Irregular

Regularity

- Left column processed
- 2 right columns
noted as same

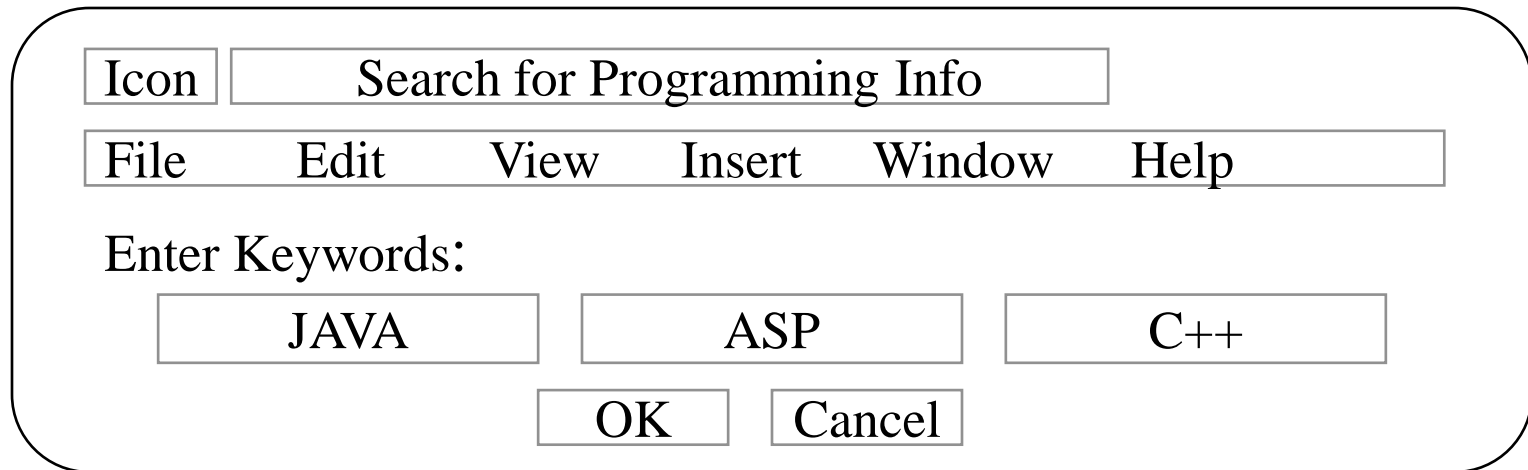


- Location & size of
each object processed



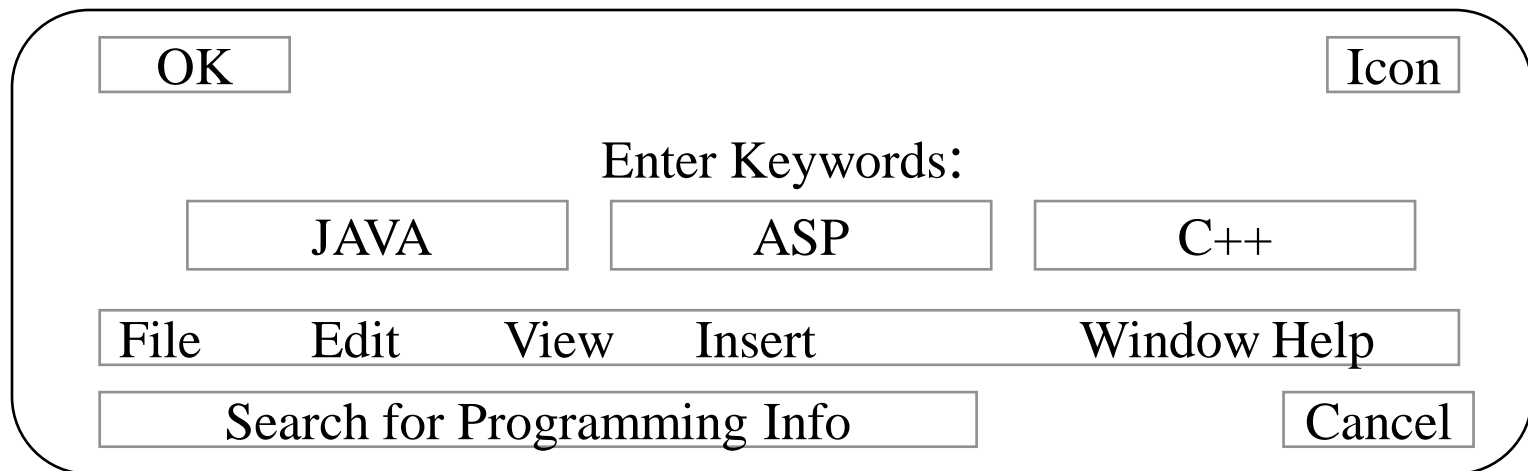
Predictability

- Put things in predictable locations on the screen



A dialog box with a rounded rectangle border. At the top left is a small 'Icon' button. To its right is a wide text input field containing the text 'Search for Programming Info'. Below this is a menu bar with the items 'File', 'Edit', 'View', 'Insert', 'Window', and 'Help'. Below the menu bar is the text 'Enter Keywords:'. Underneath this text are three text input fields containing 'JAVA', 'ASP', and 'C++'. At the bottom are two buttons: 'OK' and 'Cancel'.

Predictable

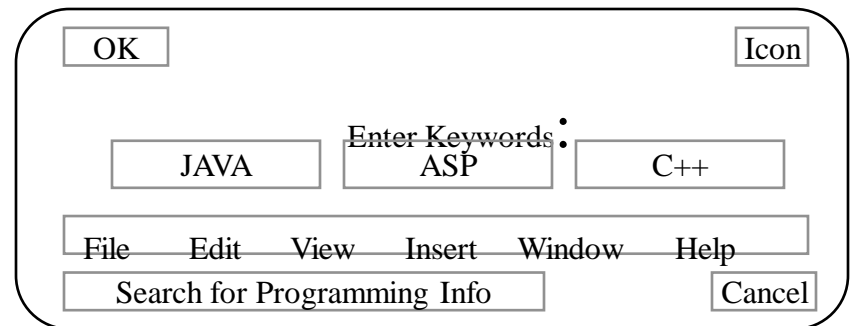
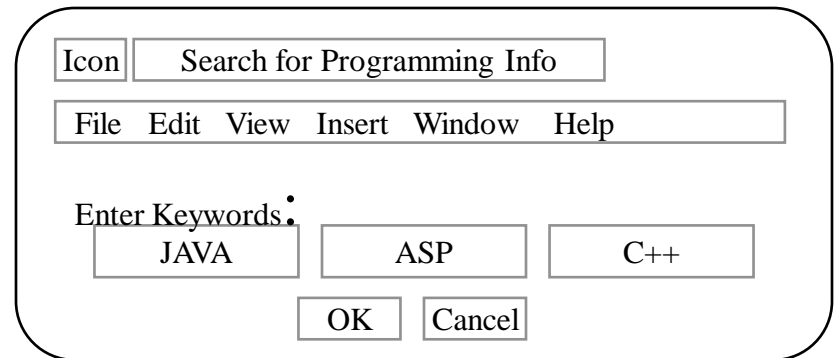


A dialog box with a rounded rectangle border. At the top left is a small 'OK' button. At the top right is a small 'Icon' button. In the center is the text 'Enter Keywords:'. Below this text are three text input fields containing 'JAVA', 'ASP', and 'C++'. Below these fields is a menu bar with the items 'File', 'Edit', 'View', 'Insert', 'Window', and 'Help'. At the bottom are two text input fields: 'Search for Programming Info' on the left and 'Cancel' on the right.

Spontaneous

Predictability

- User expects title & menu bar on top of screen
- Visual scene needs to be completely processed - objects not in expected places



Sequentiality

- Guide the eye through the task in an obvious way
 - The Eye is attracted to:
 - bright elements over less bright
 - Isolated elements over grouped
 - graphics before text
 - color before monochrome
 - saturated vs. less saturated colors
 - dark areas before light
 - big vs. small elements
 - unusual shapes over usual ones



Membership Form

Name:	<input type="text"/>	Dues:	<input type="text"/>
Address:	<input type="text"/>	Pubs:	<input type="text"/>
City:	<input type="text"/>	Total:	<input type="text"/>
State:	<input type="text"/>		
Zip:	<input type="text"/>	<input type="button" value="OK"/>	<input type="button" value="Cancel"/>

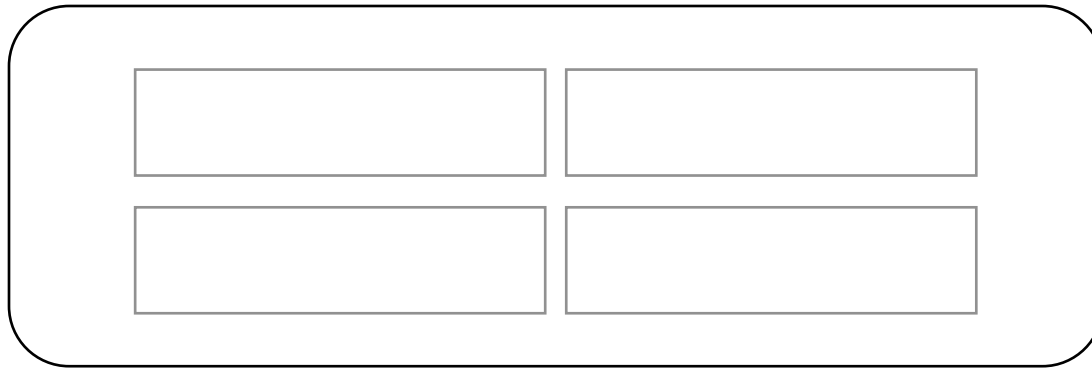
Sequential

Membership Form	<input type="button" value="Cancel"/>	Name:	<input type="text"/>	<input type="button" value="OK"/>
Address:	<input type="text"/>	Pubs:	<input type="text"/>	
	Dues:	<input type="text"/>	State:	<input type="text"/>
Zip:	<input type="text"/>	City:	<input type="text"/>	
		Total:	<input type="text"/>	

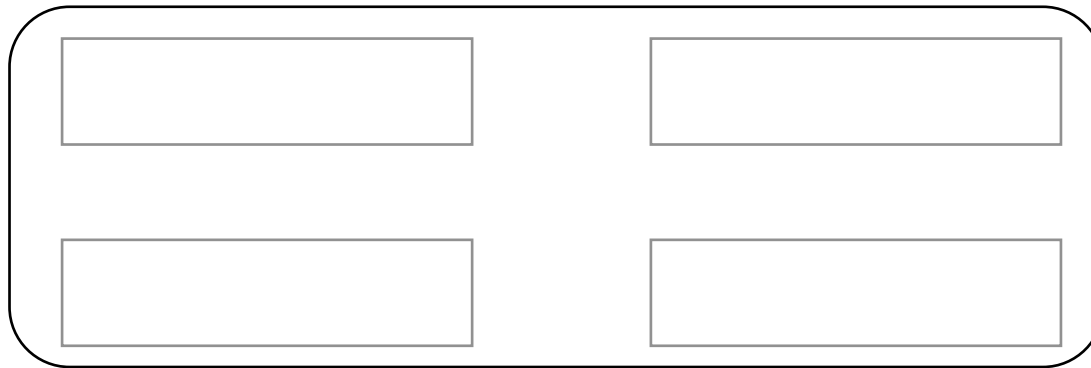
Random

Unity

- Make items appear as a unified whole (for visual coherence)
 - Use similar shapes, sizes, or colors
 - Leave less space between screen elements than at the margin of the screen



Unity

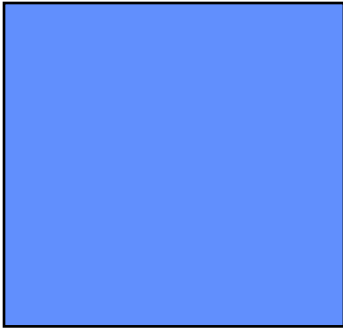


Fragmentation

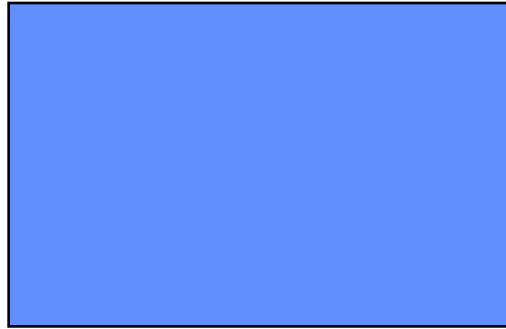
Proportion

- Create groupings of data or text by using aesthetically pleasing proportions

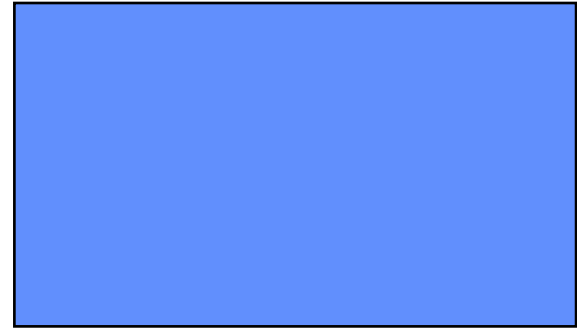
Pleasing Proportions



Square - 1:1



Square Root of 2 - 1:1.414



Golden Triangle - 1:1.618



Square Root of 3 - 1:1.732



Double Square - 1:2

Simplicity

- Minimize the number of aligned points
 - Use only a few columns to display screen elements
- Combine elements to minimize the number of screen objects
 - Within limits of clarity

Membership Form

<p>Name: <input style="width: 150px;" type="text"/></p> <p>Address: <input style="width: 280px;" type="text"/></p> <p>City: <input style="width: 280px;" type="text"/></p> <p>State: <input style="width: 60px;" type="text"/></p> <p>Zip: <input style="width: 200px;" type="text"/></p>	<p>Dues: <input style="width: 150px;" type="text"/></p> <p>Pubs: <input style="width: 150px;" type="text"/></p> <p>Total: <input style="width: 150px;" type="text"/></p>
--	---

Simple

Membership Form

<p>Name: <input style="width: 150px;" type="text"/></p> <p>Address: <input style="width: 280px;" type="text"/></p> <p>City: <input style="width: 280px;" type="text"/></p> <p>State: <input style="width: 60px;" type="text"/></p> <p>Zip: <input style="width: 200px;" type="text"/></p>	<p>Dues: <input style="width: 150px;" type="text"/></p> <p>Pubs: <input style="width: 150px;" type="text"/></p> <p>Total: <input style="width: 150px;" type="text"/></p>
--	---

Complex

Simplicity

- Only four alignments need to be processed

Membership Form

Name:	<input type="text"/>	Dues:	<input type="text"/>
Address:	<input type="text"/>	Pubs:	<input type="text"/>
City:	<input type="text"/>	Total:	<input type="text"/>
State:	<input type="text"/>		
Zip:	<input type="text"/>		

- A total of nine alignments need to be processed

Membership Form

Name:	<input type="text"/>	<input type="text"/>	
Address:	<input type="text"/>	Pubs:	<input type="text"/>
City:	<input type="text"/>	Total:	<input type="text"/>
State:	<input type="text"/>		
Zip:	<input type="text"/>		

Size::

--	--

☐

Preserve Proportions

☐

% of original height

☐

% of original width

Simple

Size:

--	--

Uniformity:

☐

Preserve Proportions

Height:

☐

% of original

Width:

☐

% of original

Complex

Groupings

- Use visual arrangements to provide functional groupings of screen elements
 - Align elements in a group
 - Evenly space elements in a group
 - Provide separation between groups
- Use additional group elements sparingly
 - color & borders add complexity

Simple Grouping

- Similar elements aligned vertically
- Vertical distance between similar objects small

Membership Form

Name:	<input type="text"/>	Dues:	<input type="text"/>
Address:	<input type="text"/>	Pubs:	<input type="text"/>
City:	<input type="text"/>	Total:	<input type="text"/>
State:	<input type="text"/>		
Zip:	<input type="text"/>	<input type="button" value="OK"/>	<input type="button" value="Cancel"/>

Boxed Grouping

- Boxes add additional complexity to form
- Spatial arrangement adequate

Membership Form

Name:

Address:

City:

State:

Zip:

Dues:

Pubs:

Total:

OK

Cancel

Background Grouping

- Color adds additional visual complexity
- Spatial arrangement adequate

Membership Form

Name:	<input type="text"/>	Dues:	<input type="text"/>
Address:	<input type="text"/>	Pubs:	<input type="text"/>
City:	<input type="text"/>	Total:	<input type="text"/>
State:	<input type="text"/>		
Zip:	<input type="text"/>		

OK Cancel

Form fill-in: guidelines

- Meaningful title
- Comprehensible instructions
- Logical grouping and sequencing of fields
- Visually appealing layout of the form
- Familiar field labels
- Consistent terminology and abbreviations
- Visible space and boundaries for data entry fields
- Error correction for individual character
- Error messages for unacceptable values
- Optional fields should be marked
- Explanatory messages for fields

Design Guidelines

- Be consistent
- Allow shortcuts
- Offer feedback
- Organize in logical groups (screens)
- Provide simple error handling
- Provide reversible actions