



UI and UX for Responsive Web Design

UI and UX for Responsive Web Design

© 2015 Aptech Limited

All rights reserved.

No part of this book may be reproduced or copied in any form or by any means – graphic, electronic, or mechanical, including photocopying, recording, taping, or storing in information retrieval system or sent or transferred without the prior written permission of copyright owner Aptech Limited.

All trademarks acknowledged.

APTECH LIMITED

www.aptech-worldwide.com

E-mail: ov-support@onlinevarsity.com

Edition 1 – November 2015



Preface

This book, **UI and UX for Responsive Web Design**, provides an understanding of how Responsive Design works and highlights the importance of a UI design. This book also provides the learning process of creating a perfect user experience design for all kinds of users and also on how to create a responsive Website that can be viewed on devices with differing screen sizes and characteristics such as retina displays.

The knowledge and information in this book is the result of the concentrated effort of the Design Team, which is continuously striving to bring to you the latest, the best, and the most relevant subject matter in Information Technology. As a part of Aptech's quality drive, this team does intensive research and curriculum enrichment to keep it in line with industry trends and learner requirements. Please feel free to send your feedback.

Design Team

Onlinevarsity
your e-way to learning

TechnoWise



Are you a
TECHIE GEEK
looking for updates?

Logon to

www.onlinevarsity.com

Table of Contents

Session 01: Introduction to User Interface Design

Session 02: Introduction to User Experience Design

Session 03: Understanding Responsive Web Design

Session 04: Understanding the Strategies for Responsive Web Design

Session 05: Using Common RWD Patterns in Mobile Designs

Session 06: Usability Studies in Responsive Web Design

Session 1

Introduction to User Interface Design

Learning Objectives

In this session, you will learn to:

- Define User Interface (UI) and User Interface design
- List and explain the different elements of User Interface Design
- Describe the principles of User Interface Design
- Describe the types of User Interface Design
- Explain the User Interface Design process
- Describe the models in User Interface Design

The user interface is a vital part of any computer system. It determines how easily an end user can interact with the program. One of the achievements of a system depends on how well a user interface is designed and whether it creates a seamless experience for end users.

This session begins with explaining the meaning of User Interface (UI) and UI design. It provides an overview of the fundamental parts of the UI and different types of UI designs. The session also covers the UI design principles that can be utilized to improve the quality of user interface design. Further, the session outlines the user interface design process and the different UI models.

1.1 Definition of UI and UI Design

What is User Interface?

The UI of an application, also known as an 'interface', is the means by which a user and a computer system interact. It can comprise both software and hardware components.

In particular, UI includes:

- The textual, graphical, and auditory information that the program presents to the user.
- The control sequences that a user employs to control the program. For example, mouse movements, keystrokes with the computer keyboard, and selections through the touchscreen.

A simple example of a UI in the real world is an Automatic Teller Machine (ATM). As shown in figure 1.1, it consists of a keypad, a display window, a selection of choice options, and a help screen that displays instructions for completing an ATM transaction.



Figure 1.1: ATM User Interface

What is User Interface Design?

User Interface Design is the process of designing user interfaces for Websites, appliances, computers, and software applications. It focuses on anticipating an end user's requirement, that is, what the users might need to do and then, ensuring that the UI has all elements to facilitate those actions. The goal of UI design is to maximize user's experience and interaction.

UI design brings together the following concepts:

- **Information Architecture** focuses on organizing, structuring, and labelling content in an effective and sustainable way.
- **Interaction Design** concentrates on setting up an engaging interface with logical behaviors.
- **Visual Design** concentrates on the style of a site and its related materials through pictures, hues, text styles, and different components.

The overall goal of UI design is to make the user's experience and interaction as simple and efficient as possible.

1.2 Parts of User Interface Design

The fundamental parts or elements of most UIs are as follows:

- Input Controls
 - Navigational Components
 - Informational Components
 - Containers
- **Input Controls**

An input control defines the way in which the system captures information and it is the most interactive component of the UI. Some common interface input controls consist of elements such as buttons, radio buttons, check boxes, drop-down lists, drop-down buttons, list boxes, and text fields.

Table 1.1 lists the input control elements and their descriptions along with examples.

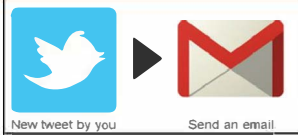

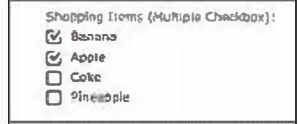
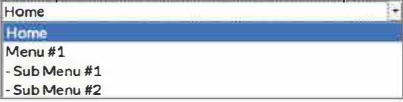

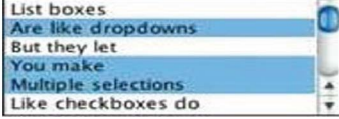

Element	Description	Example
Button	A button can be pressed or clicked by the user to perform an action. It is typically labelled using text, an icon, or both.	
Radio button	Radio buttons present a set of choices from which the user can choose one.	
Check box	Check boxes also present a set of choices but allow the user to select one or more options from the set.	
Drop-down list	Drop-down list comprise a list of items that are displayed upon clicking the arrow and enable one item at a time to be selected. They are preferred when many items in a set need to be presented to the user.	
Drop-down button	Drop-down buttons when clicked display a drop-down list of items that are mutually exclusive.	
List box	List box contains a list of options. It allows the user to select one or more items from a list.	 <i>Image Courtesy: http://www.usability.gov/how-to-and-tools/methods/user-interface-elements.html</i>
Text field	Text field is a basic text control that allows the user to enter a small amount of text. When a user indicates that text entry is complete usually by pressing Enter, the text field fires an action event.	

Table 1.1: Description of Input Control Elements

▪ Navigational Components

Navigational Components provide a way for the users to tell the system what to do. Some of these include search fields, breadcrumbs, pagination, tags, icons, and image carousel.

Table 1.2 lists the navigational components and their descriptions along with examples.




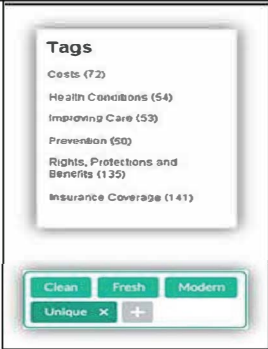


Element	Description	Example
Search Field	Search field enable users to enter a keyword or phrase and submit it with the objective of getting back the most significant results. It is generally a single-line content box that can accept user input to be searched within a database.	
Breadcrumb	Breadcrumb is a navigation aid that allows the user to keep track of user's location within programs. Breadcrumbs show up on a level plane over the highest point of a Web page, below the title bars or headers. They show links for previous page the user navigated through to get to the present page or in hierarchical site structures, the parent pages of the current one.	
Pagination	Pagination enables you to divide content into distinct pages. It allows the user to skip between pages or go in sequential order through the content.	
Tags	Tags allow users to identify content in the same classification. Some tagging framework also allows users to apply their own tags to content by entering them into the framework.	
Icon	An icon acts as a natural symbol to represent some functionality of the system, such as save or open, and so on. Icons are often hyperlinked.	
Image Carousel	An Image carousel allows users to browse through a set of items and choose one. Typically, the images are hyperlinked.	

Table 1.2: Description of Navigational Components

Informational Components

Some common interface information components are tooltips, notifications, progress bars, message boxes, and modal window.

Table 1.3 lists the informational components and their descriptions along with examples.




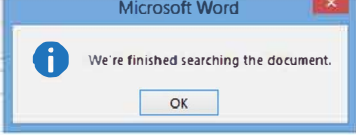
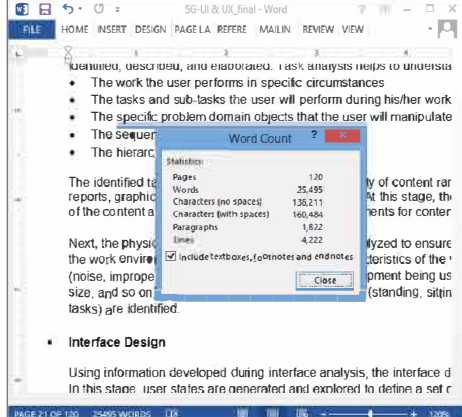
Element	Description	Example
Tooltip	A tooltip is a common UI element used in conjunction with a cursor, usually a pointer. Tooltips allow the user to see clues when they hover over an item demonstrating the name or reason of the item.	
Notification	Notifications are update message that announces something new for the user to see. Notifications are typically used to show items such as, the successful completion of a task, or an error, or warning message.	
Progress Bar	Progress bar shows where a user is as they progress through a series of steps in a process.	
Message Box	Message box is a small window that gives information to users or gets a confirmation from the user. A message box requires users to take an action before they can move forward.	
Modal Window (pop-up)	Modal window is a child window that requires users to associate with it before it can return to operating the parent application, thus, ensuring the user will perform the operation it is intended for.	

Table 1.3: Description of Information Components

Containers

Containers are designed to contain page elements to a reasonable maximum width based on the size of a user's screen. The most common interface container includes Accordion.

Table 1.4 lists the most common type of container and its description along with an example.

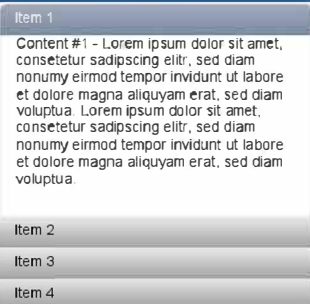
Element	Description	Example
Accordion	An accordion is a component that show/hides a vertical list of items and content based on user action. When a label is clicked, it expands the section showing the content within.	

Table 1.4: Containers

1.3 Principles and Attributes of User Interface Design

UI design principles focus on improving the quality of user interface design. Some of these include:

The Structure Principle

This principle is concerned with overall UI architecture.

The design of the interface should be visually, theoretically, and linguistically clear. It should provide clear and user-specific paths to useful and relevant information.

Tips/Techniques to support the Structure Principle

The techniques that help to support this principle are as follows:

- You should group the logically connected items to communicate and separate unrelated items to achieve visual organization.
- Ensure good cross-linkages and quick jumps to important sections of the Websites.
- Design page-specific navigation and access to information.
- Keep the screen less cluttered and easier to understand.
- Present the flow of actions, information, responses, and visual preparations in a sensible order that is easy to remember and place in context.

Refer to figure 1.2 to view an example of a clearly structured UI.



Figure 1.2: Example of a Clearly Structured UI

Image Courtesy: <https://www.amazon.com>

Amazon.com is a perfect example of a clearly structured UI. It has a clean interface, less clutter, and a clear hierarchy of the content.

■ **Simplicity Principle**

The design should be simple to learn and use. It should include only the elements that are most important for communication. It should also make common tasks easy to perform and provide good shortcuts that are meaningfully related to longer procedures.

Tips/Techniques to support the Simplicity Principle

The techniques that help to support this principle are as follows:

- Ensure that you create natural designs, which can be easily understood by the users based on their past experiences
- Make use of colors and fonts carefully
- Avoid acronyms and terminology with no clear explanation, which are likely to confuse the users
- Ensure that the messages and labels are written in a clear manner
- Use direct icons that are consistent with Web terminology and use less space on screen
- Make use of familiar concepts and use a language that is known to the user
- Use consistent layout, navigation, messages, labels, and representation of the information
- Do not use jarring animations and advertisements
- Use white/blank space effectively
- Avoid elements that may cause compatibility or support issues across different environments

Refer to figure 1.3 to view an example of a simple UI.

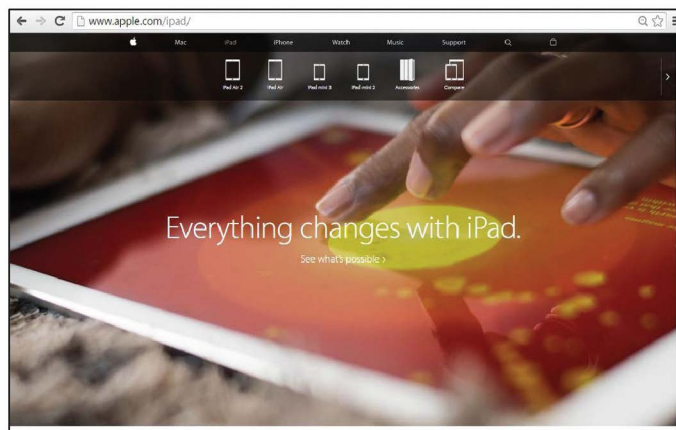


Figure 1.3: Example of a Simple UI

Image Courtesy: <http://www.apple.com/ipad/>

The Website of Apple.com is a great example to explain the simplicity principle.

The UI of Apple.com is very simple. To avoid any distraction, only the relevant content is presented in a clean and straightforward way. The clear UI design makes the Website intuitive to use.

▪ **Visibility Principle**

Visibility ensures that the user clearly sees the interface and all the possible actions. The UI design should make all required options for a given task visible without confusing the user with superfluous or redundant information. It needs to be straightforward to let users easily comprehend the interface and navigate through it more efficiently.

Tips/Techniques to support the Visibility Principle

The techniques that help to support this principle are as follows:

- Have limited entry points on the interface
- Make the entry points descriptive and 'task-oriented'
- Avoid creating UIs that will always be busy and unresponsive to users
- Do not overwhelm users with too many alternatives or confuse with unnecessary information
- Do not direct users into pages that have no navigational options
- Create levels of importance. Elements that are mission critical to the Website or application should be designed to be more visible. However, the less important elements can be designed for less visibility
- Use colors appropriately. Do not use too many colors as it may distract users

Refer to figure 1.4 to view an example of a clearly visible UI.

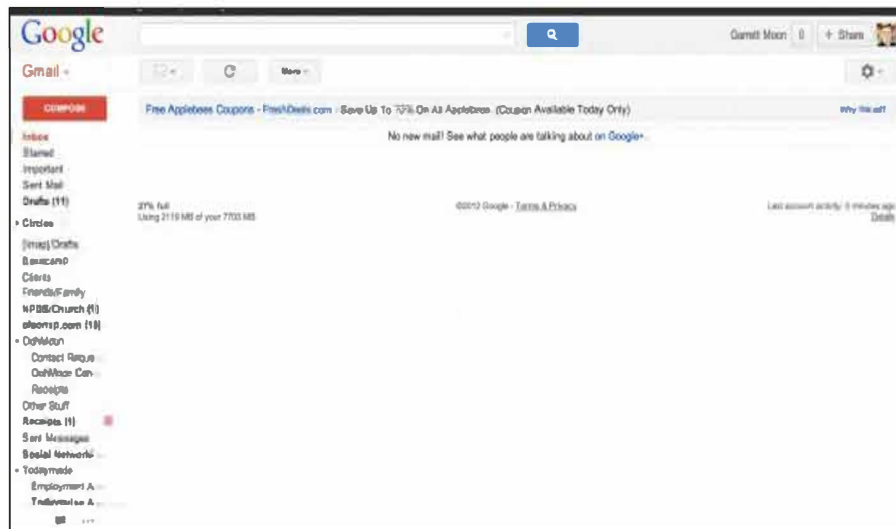


Figure 1.4: Example of a Clearly Visible UI

Image Courtesy: <http://todaymade.com/blog/wp-content/uploads/2012/04/Gmail-Inbox-garrett.t.moon@gmail.com-1-560x283.jpg>

The Website of Gmail is a great case to explain the Visibility principle.

The design of the interface is very straightforward. All the options are clearly visible. Users can quite effortlessly, and perhaps, instinctively distinguish different section and guess what their functions are, such as Inbox, Sent items, Deleted items, Important mails, and so on. By making the button large, placing it in a noticeable position and giving it a distinct color, it stands out in the design.

■ Feedback Principle

Feedback means to give information to the user based on some action. The UI design should clearly inform users what is going on and show them the result of their actions such as connecting, waiting to download, processing, or downloading. It should also inform users about the actions and changes of state or condition. Further, it should also inform about errors or exceptions that users might face on performing particular actions by using a clear, concise, and understandable language.

Feedback can come in many forms, such as a color change of an interface element, sound alerts, pop-up windows with action buttons, notification bubbles, and so on.

Tips/Techniques to support the Feedback Principle

The techniques that help to support this principle are as follows:

- Write your messages and labels clearly
- Respond to user action as close as possible to the point of interaction (that is, within a reasonable amount of time)
- Use colors appropriately

Refer to figures 1.5 (a) and 1.5 (b) to view an example of a UI displaying feedback.

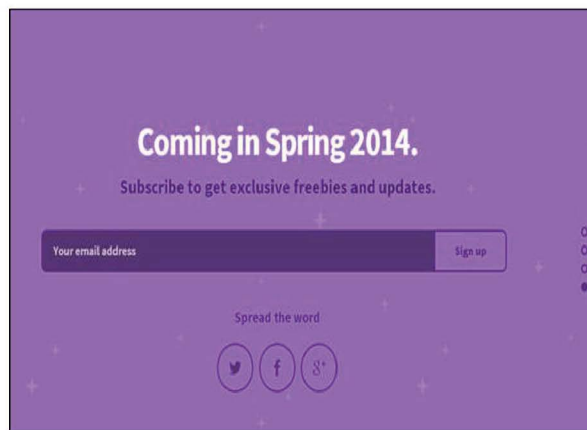


Figure 1.5 (a): UI of 'Coming Soon' Page from Kickdrop.me

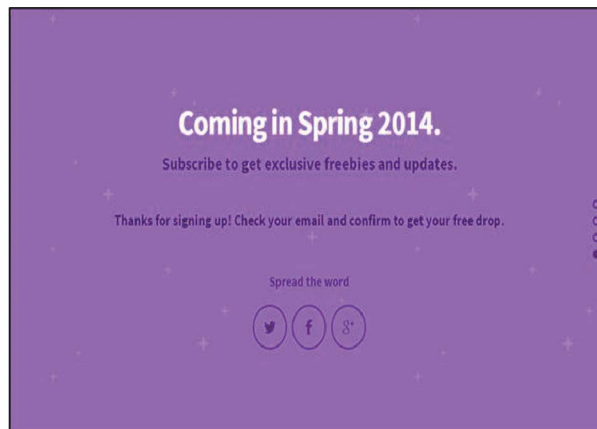


Figure 1.5 (b): Example of UI Displaying Feedback

Images Courtesy: <http://www.howdesign.com/web-design-resources-technology/donald-normans-design-principles-modern-web-design/>

The 'Coming Soon' page with the **Thanks** message from Kickdrop.me is a great example to explain the Feedback principle. To subscribe, the user has to enter the required details (in this case, an email ID) and then click the Sign-up button. When a user clicks the button, the email form slides out and a confirmation message slides into view. This grabs the user's attention and emphasizes that clicking the button has produced a result. In response to the signing up, a Thank you message is displayed.

▪ **Tolerance Principle**

The Tolerance UI design principle emphasizes the importance of designing the UI to prevent users from making errors. It allows the user to learn how to use the site and informs them of errors.

Tips/Techniques to support the Tolerance Principle

The techniques that help to support this principle are as follows:

- Display error messages that provide the information necessary for recovery
- Use clear and familiar language that can be easily understood by the user
- Use specific and constructive terms to avoid any ambiguity
- If the user requests, provide additional explanation during error connection

- Ensure that users never lose their work as a result of an error on their part or system or communication problems

Refer to figure 1.6 to view an example of a tolerant UI.

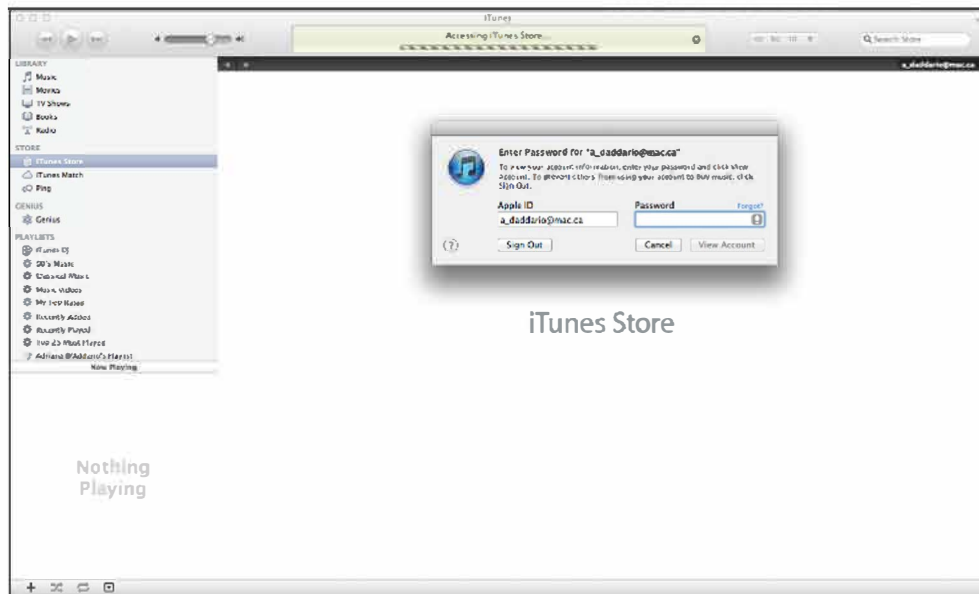


Figure 1.6: Example of a Tolerant UI

Image Courtesy: <https://adaddario16.wordpress.com/2012/06/07/principles-and-techniques-of-user-interface-design/>

The iTunes interface is an example to explain the Tolerance principle.

The iTunes interface allows the user to be notified of certain aspects. For example, when the user is keying in their password in capital letters, it notifies the user with a simple symbol. When the user is missing an attachment, another symbol such as an exclamation point comes up, that explains which file is missing or lost. Each feature tells the user that something needs to be done to fix the issue and the interface assists by offering recommendations.

▪ **Reuse Principle**

The UI design should reuse internal and external components and behaviors to maintain consistency with purpose. This reduces the need for users to rethink and remember. A simple example of this is the keyboard shortcuts.

The techniques that help to support this principle are as follows:

- Group elements effectively
- Ensure that the same action should always yield the same result
- Do not change the function of the elements
- Do not change the position of standard elements
- Set UI design standards and then stick to them

Refer to figure 1.7 to view an example of a reusable UI.

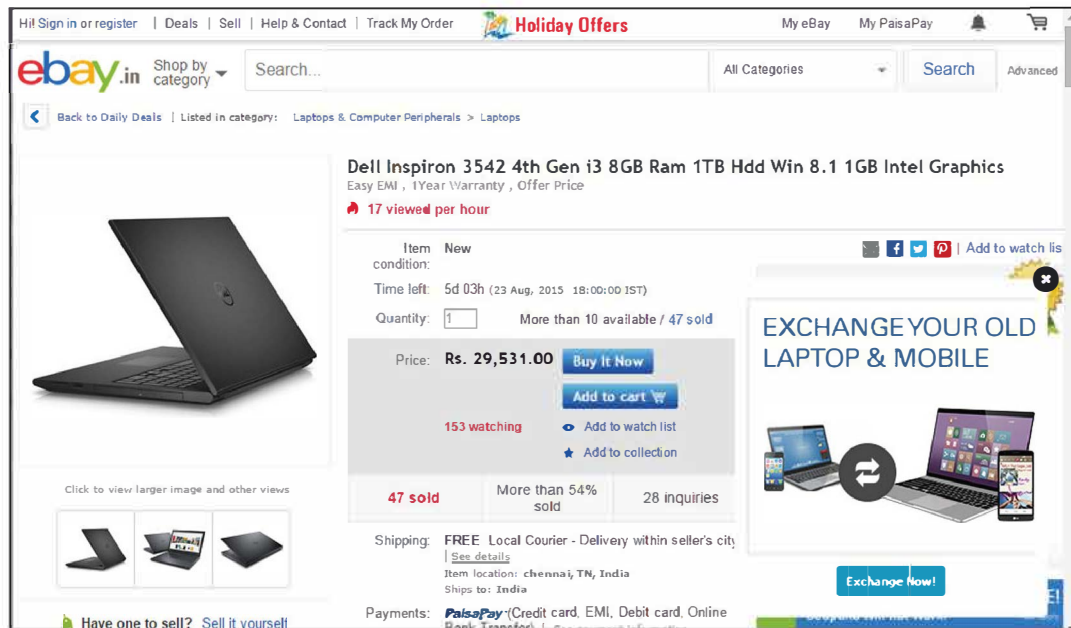


Figure 1.7: Example of a Reusable UI

Image Courtesy: <http://www.ebay.in/>

An example of the Reuse principle is ebay.in. The process in which a user goes through to purchase a product is very simple and organized. In addition, once the user goes through it, he/ she is able to repeat the actions easily.

Quick Test 1.1

1. Input controls define the way the system captures the information. Input controls are the interactive components of the user interface.
 - a. True
 - b. False
2. Interaction Design avoid on creating engaging interfaces with well-thought-out behaviors.
 - a. True
 - b. False

1.4 Types of UIs

UIs can be classified into six categories:

■ Command Language-Based Interface

A command language-based interface is a means of interacting with a computer program where the user issues commands to the program in the form of successive lines of text. The computer displays a prompt, the user keys in the command, and presses Enter key. After receiving the command, the command language-based interface processes it accordingly and shows the output/result on the same screen. The most common example of the command language-based interface is MS-DOS.