

# System Assistant

Evolution of Computers & its applications:

<https://www.gkseries.com/mcq-on-computer-history/multiple-choice-questions-and-answers-on-computer-history>

Basics of Hardware and Software: <https://testbook.com/objective-questions/mcq-on-hardware--5eea6a1639140f30f369f54e>

CPU: <https://letsfindcourse.com/computer-fundamentals-mcq/cpu-mcq-questions-and-answers>

## CMD

Command Prompt- Directory navigation, path setting, creating and using batch files. Drives, files, directories, directory structure

Certainly! The Command Prompt in Windows provides various functionalities for navigating directories, setting paths, creating batch files, managing drives, files, and directories. Here's a breakdown of some commonly used commands and their functionalities:

### Directory Navigation:

#### 1. `cd` (Change Directory):

- Used to change the current directory.
- Example: `cd Documents` navigates to the Documents directory.

#### 2. `dir` (Directory Listing):

- Lists the contents of the current directory.
- Example: `dir` lists all files and directories in the current directory.

### 3. `mkdir` (Make Directory):

- Creates a new directory.
- Example: `mkdir NewFolder` creates a new directory named "NewFolder".

### 4. `rmdir` (Remove Directory):

- Deletes a directory.
- Example: `rmdir OldFolder` deletes the directory named "OldFolder".

## Path Setting:

### 1. `set` (Set Environment Variables):

- Used to set environment variables including PATH.
- Example: `set PATH=%PATH%;C:\NewPath` appends "C:\NewPath" to the PATH variable.

## Creating and Using Batch Files:

### 1. Creating Batch Files:

- Create a new text file with a `.bat` extension.
- Write commands in the file, one per line.
- Save the file.

### 2. Running Batch Files:

- Simply type the name of the batch file in the command prompt and press Enter.

## Drives:

### 1. `cd` with Drive Letter:

- You can switch between drives by specifying the drive letter followed by a colon.
- Example: `D:` switches to the D drive.

### 2. `vol` (Volume Label):

- Displays the volume label and serial number of a disk.
- Example: `vol` shows information about the current drive.

## Files:

### 1. `copy` (Copy Files):

- Copies one or more files to another location.
- Example: `copy file1.txt C:\Destination` copies "file1.txt" to "C:\Destination".

### 2. `del` (Delete Files):

- Deletes one or more files.
- Example: `del file1.txt` deletes "file1.txt".

### 3. `move` (Move Files):

- Moves one or more files from one directory to another directory.
- Example: `move file1.txt C:\Destination` moves "file1.txt" to "C:\Destination".

## Directory Structure:

### 1. `tree` (Directory Tree):

- Displays the directory structure of a path or of the disk.
- Example: `tree /F` shows the directory tree structure with files.

## Linux

Linux- What is Open Source, Overview of Linux, Linux Origins, Running Commands, Getting Help, Linux File Hierarchy Concepts, Some Important Directories, Current Working Directory, File and Directory Names, Absolute and Relative Pathnames, Changing Directories, Listing Directory Contents, Copying Files and Directories, Moving and Renaming Files and Directories, Creating and Removing Files, Creating and Removing Directories

## Open Source:

Open Source refers to software where the source code is freely available to the public for use, modification, and distribution. This allows developers to collaborate, improve, and customize software according to their needs. Linux is one of the most famous examples of open-source software.

## Overview of Linux:

Linux is a Unix-like operating system kernel that serves as the foundation of various Linux distributions, or "distros". It is renowned for its stability, security, and flexibility. Linux provides a command-line interface (CLI) where users interact with the system using commands.

## Linux Origins:

Linux was created by Linus Torvalds in 1991 as a hobby project. He developed it to be a free operating system kernel compatible with Unix. Over time, it has grown into a powerful and versatile platform, powering servers, desktops, embedded systems, and more.

## Running Commands:

Commands are executed in the terminal. They perform various tasks such as managing files, directories, processes, and system configurations.

## Getting Help:

- `man` : Displays the manual pages for commands.
  - Example: `man ls` displays the manual for the `ls` command.
- `-help` : Many commands support the `-help` option to display usage information.
  - Example: `ls --help` displays help for the `ls` command.

## Linux File Hierarchy Concepts:

Linux follows a hierarchical file system structure:

- `/` : The root directory.
- `/bin` , `/usr/bin` : Binaries and executable files.
- `/etc` : System configuration files.
- `/home` : User home directories.

- `/var` : Variable data such as logs and temporary files.

## Important Directories:

- `/bin` : Essential system binaries.
- `/etc` : Configuration files.
- `/home` : User home directories.
- `/var/log` : System log files.
- `/tmp` : Temporary files.

## Current Working Directory:

The current directory is the directory in which commands are executed. You can check it with the `pwd` command.

## File and Directory Names:

Linux file and directory names are case-sensitive and can contain letters, numbers, underscores, and hyphens. They can also begin with a dot, making them hidden.

## Absolute and Relative Pathnames:

- Absolute: Specifies the full path from the root directory.
  - Example: `/home/user/file.txt`
- Relative: Specifies the path relative to the current directory.
  - Example: `../parent_directory/file.txt`

## Changing Directories:

Use the `cd` command to change directories.

- Example: `cd /home/user`

## Listing Directory Contents:

Use the `ls` command to list directory contents.

- Example: `ls -l` for a detailed listing.

## Copying Files and Directories:

Use the `cp` command to copy files and directories.

- Example: `cp file1.txt /path/to/destination`

## Moving and Renaming Files and Directories:

Use the `mv` command to move or rename files and directories.

- Example: `mv file1.txt /path/to/destination`

## Creating and Removing Files:

- Use the `touch` command to create empty files.
  - Example: `touch newfile.txt`
- Use the `rm` command to remove files.
  - Example: `rm file.txt`

## Creating and Removing Directories:

- Use the `mkdir` command to create directories.
  - Example: `mkdir new_dir`
- Use the `rmdir` command to remove empty directories.
  - Example: `rmdir empty_dir`

## Data Communication and Network Technologies:

Introduction to Computer Networks, LAN, WAN, MAN etc.

Communication Systems, Signal and data, Transmission modes, Synchronous and asynchronous transmission.

Transmission Media: Guided Media (Twisted pair, Co-axial cable, Optical fiber).

Introduction to Communication Protocols.

Internet Addressing Scheme: IP and MAC.

Networks Layers/Models, Networking Devices.

Network security.

Introduction to Firewalls and its utility.

## Introduction to Computer Networks:

- **Definition:** A computer network is a collection of interconnected devices that can communicate with each other to share resources and information.
- **Types:**
  - LAN (Local Area Network): Covers a small geographical area.
  - WAN (Wide Area Network): Spans large distances and may include multiple LANs.
  - MAN (Metropolitan Area Network): Intermediate between LAN and WAN, covering a city or large campus.

## Communication Systems:

- **Signal and Data:** Signals are physical representations of data. Data is the information being transmitted.
- **Transmission Modes:**
  - Simplex: One-way communication.
  - Half-duplex: Two-way communication, but only one direction at a time.
  - Full-duplex: Two-way communication simultaneously.

## Transmission Media:

- **Guided Media:**
  - Twisted Pair: Copper wires twisted together, commonly used in Ethernet connections.
  - Coaxial Cable: Central conductor surrounded by an insulating layer and a metallic shield, used in cable TV and networking.
  - Optical Fiber: Uses light to transmit data, providing high bandwidth and immunity to electromagnetic interference.

## Introduction to Communication Protocols:

- **Definition:** Protocols are rules and conventions governing communication between devices.
- **Examples:** TCP/IP, HTTP, FTP, SMTP.

## Internet Addressing Scheme:

- **IP (Internet Protocol):** Unique address assigned to each device on a network, allowing them to communicate with each other.
- **MAC (Media Access Control):** Unique identifier assigned to network interfaces.

## Network Layers/Models:

- **OSI Model (Open Systems Interconnection):** Divides network communication into seven layers, each responsible for specific functions.
- **TCP/IP Model:** Based on four layers: Application, Transport, Internet, Link.

## Networking Devices:

- **Router:** Connects multiple networks and forwards data packets between them.
- **Switch:** Connects devices within a network and forwards data packets to their intended destination.
- **Hub:** Simplest networking device, broadcasts data to all devices connected to it.
- **Firewall:** Monitors and controls incoming and outgoing network traffic based on predetermined security rules.

## Network Security:

- **Authentication:** Verifying the identity of users or devices.
- **Encryption:** Encoding data to prevent unauthorized access.
- **Firewalls:** Act as a barrier between a trusted internal network and untrusted external networks, controlling incoming and outgoing traffic.

## Introduction to Firewalls and its Utility:

- **Firewall:** A network security device that monitors and filters incoming and outgoing network traffic based on predefined security rules.
- **Utility:** Protects networks from unauthorized access, malware, and other security threats by controlling traffic flow and enforcing security policies.



## **Data Representation:**

Number system (Binary, Decimal, octal and Hexadecimal).

ASCII code, Unicode, BCD, Grey, XS3, EBCDIC.

Complement, addition, subtraction, overflow, Floating point representation.

## **Number Systems:**

### **1. Binary (Base-2):**

- Consists of only two digits: 0 and 1.
- Used in digital electronics and computing.

### **2. Decimal (Base-10):**

- Consists of ten digits: 0 through 9.
- Commonly used in everyday mathematics.

### **3. Octal (Base-8):**

- Consists of eight digits: 0 through 7.
- Each octal digit represents three binary digits.

### **4. Hexadecimal (Base-16):**

- Consists of sixteen digits: 0 through 9 and A through F (representing 10 to 15).
- Compact representation of binary data, used in programming and digital systems.

## **Character Encoding:**

### **1. ASCII (American Standard Code for Information Interchange):**

- Standard character encoding scheme representing characters using 7 or 8 bits.
- Originally used for communication between computers and peripherals.

### **2. Unicode:**

- Character encoding standard supporting multiple languages and character sets.

- Uses variable-length encoding, with UTF-8 being the most common.

### 3. **BCD (Binary-Coded Decimal):**

- Represents decimal numbers in binary form.
- Each decimal digit is represented by a group of four binary digits.

### 4. **Gray Code:**

- Binary numeral system where two consecutive values differ in only one bit.
- Used in error detection and correction circuits.

### 5. **XS-3 (Excess-3):**

- Non-weighted code where each decimal digit is represented by adding 3 to its corresponding binary value.
- Used in digital systems for arithmetic operations.

### 6. **EBCDIC (Extended Binary Coded Decimal Interchange Code):**

- Character encoding scheme developed by IBM for mainframe computers.
- Used in older IBM systems and environments.

## **Complement, Addition, Subtraction:**

### 1. **Complement:**

- **1's Complement:** Invert all bits.
- **2's Complement:** Invert all bits and add 1.

### 2. **Addition:**

- Binary addition follows similar rules to decimal addition.
- Carry-over occurs when the sum exceeds the base.

### 3. **Subtraction:**

- Convert subtraction to addition using 2's complement.
- Subtract by adding the 2's complement of the subtrahend.

## **Overflow:**

- **Overflow** occurs when the result of an arithmetic operation exceeds the capacity of the number representation.
- In binary arithmetic, overflow happens when the result requires more bits than available.

### **Floating Point Representation:**

- **Floating Point Representation** is used to represent real numbers in computers.
- Typically consists of a sign bit, exponent, and mantissa.
- Follows IEEE 754 standard for floating-point arithmetic.