1. **BASIC COMPUTING**

**1. INSTALLATION OF WINDOWS OPERATING SYSTEM**

1. Make sure your device meets the minimum system requirements

For the latest version of Windows 10, you’ll need to have the following:

* CPU: 1GHz or faster supported processor
* RAM: 1GB for Windows 10 32-bit or 2GB for Windows 10 64-bit
* Storage: 32GB of space or more
* GPU: DirectX 9 compatible or later with WDDM 1.0 driver.
* Display: 800x600 resolution or greater
* Internet connection: Some versions of Windows 10 require an internet connection during setup

1. Create USB installation media

Microsoft has a tool specifically for creating installation media. Visit Microsoft's Windows 10 download page and select “Download tool now” under the “create Windows 10 installation media” section. Transfer the downloaded installer tool to a USB drive.

1. Run the installer tool. Open the installer tool by clicking on it. Accept Microsoft’s terms, and then select “Create installation media for another PC” on the “What do you want to do?” page. After selecting which language you want Windows 10 to run in, and which edition you want as well (32-bit or 62-bit), you’ll be asked what type of media you want to use.

Installing from a USB drive is definitely the preferred option but you can also install from a CD or ISO file. Once you choose your device, the installer tool will download the required files and put them onto your drive.

1. Use your installation media. Insert your installation media into your device and then access the computer’s BIOS or UEFI (press the F2, F12 or Delete keys as your computer boots up.). These are the systems that allow you to control your computer’s core hardware.
2. Change your computer's boot order. Once you have access to your computer’s BIOS/UEFI you’ll need to locate the settings for boot order. You need the Windows 10 installation tool to be higher up on the list than the device’s current boot drive: this is the SSD or HDD that your existing OS is stored on. You should move the drive with the installer files to the very top of the boot order menu. Now, when you restart your device the Windows 10 installer should load up first.
3. Restart your device. Save your settings in the BIOS/UEFI and reboot your device.
4. Complete the installation. Your device should now load up the Windows 10 installation tool on restart. This will guide you through the rest of the installation process.

**2. INSTALLATION OF LINUX (UBUNTU 20.04) OPERATING SYSTEM**

**Prerequisites**

* System requirements (recommended):
* 2 GHz dual-core processor
* 4GB memory
* 25GB available disk space for storage (less if installing the minimal version)
* DVD drive or USB port
* At least a 4GB USB drive

1. Download the Installation Media

In a web browser, visit the Ubuntu download page and pick a version suitable for your machine (Ubuntu 20.04 for desktops). The download is an .iso file. You can use it to create a bootable USB drive. Save the file to a location of your choice.

1. Create Bootable USB

Use the Create startup disk tool:

* 1. Open a search dialog, and type create start up.
  2. If it’s not installed, the Software Center will offer the option to install it – choose the option for USB drive, then open the utility.
  3. In the top pane, click Other, then browse and select the Ubuntu 20.04 .iso file you downloaded.
  4. In the bottom pane, select your USB drive.
  5. Click Make startup disk.

1. Boot up Ubuntu from USB
2. Turn off your system. Make sure you remove all other USB devices, such as printers, memory cards, etc.
3. Insert the Ubuntu USB drive into the system and turn on your machine.

There are two possible scenarios:

* The computer boots the USB drive automatically.
* You need to manually configure USB booting in the Boot Menu or BIOS/UEFI.

1. To manually configure the boot order, tap the boot menu key about once or twice per second as soon as the computer powers on
2. Once you see your boot menu, use the arrows to pick the Ubuntu media to boot from. For a DVD, the entry will usually have DVD or Optical in the name. USB is usually labeled USB.

Your system should start loading the Ubuntu live disc menu.

1. Run Ubuntu

You can test Ubuntu 20.04 before you commit to installing it. The .iso includes a live mode that only runs in memory.

Launch this mode by clicking Try Ubuntu

1. Install Ubuntu 20.04 LTS Desktop

To begin the installation, click Install Ubuntu.

Choose Keyboard Layout.

By default, the system will select English and English.

If you have a non-standard keyboard, you can select it in the list. Alternately, click Detect Keyboard Layout and the system will automatically choose your keyboard. If you need to test your keyboard, use the labeled field.

When you’re ready, click Continue.

Choose Starting Applications

* Normal Installation – This is the full Ubuntu Desktop experience, with office software, games, and media players.
* Minimal Installation – Choose this to save disk space, especially if you won’t be using media players or productivity software.

You’ll also be asked to confirm other options:

* Download updates while installing Ubuntu – This does the work of downloading large package files during the installation. Once the installation finishes, the packages will be ready to apply as updates.
* Install third-party software for graphics and Wi-Fi hardware and additional media formats – Some hardware, like graphics cards and wi-fi cards, do not have open-source driver support. Also, some media formats, such as .wmv, do not fall under the GPL license. If you need support for these, you’ll need to agree to additional terms of use.

Disk Partitioning

Next, you’ll be presented with an Installation Type dialog. You can wipe the hard drive clean prior to installing Ubuntu by clicking Erase disk and install Ubuntu. If you go this route, skip ahead to the next step.

Advanced users may want to edit Advanced Features. Use this to specify your own disk partitions or set other advanced options:

* Use LVM with the new Ubuntu installation: LVM stands for Logical Volume Management. This is a tool for dynamically managing different virtual drives on your system. It’s much like an enhanced version of the gparted tool.
* Encrypt the new Ubuntu installation for security: This will encrypt the drive’s contents. You’ll choose a security key, which will be required to decrypt and use the drive.
* Experimental: Erase disk and use ZFS: ZFS refers to Zettabyte File System, but it has grown into a hybrid file system and volume manager. Since it’s still being tested, avoid this setting on mission-critical production systems.

If you’d rather create your own hard drive partitions, click Something Else.

The next screen will allow you to create your own partition table and logical drives. This lets you divide a physical hard drive into different partitions. The operating system sees partitions as individual drives.

Click Continue to apply your changes to the drive partitions.

Select Time Zone

Once the system formats the disk partitions, the installer will ask Where are you?

Type the nearest large city into the box, and the system will set your local time zone.

Click Continue.

Create User Account

Next, you’ll need to configure a user account. Fill in the following fields:

* Name: Your actual name.
* Computer name: This is the hostname or network name.
* Username: The user account name you want to use.
* Password: Enter and confirm a strong password – the installer will automatically evaluate your password strength.
* Log in automatically: This is not recommended for publicly accessible servers.
* Require my password to log in: This is recommended for publicly accessible servers.

Click Continue to install Ubuntu.

Once the installer finishes, remove the Ubuntu installation media. You’ll be prompted to Restart Now.

The system should boot into your fresh install of Ubuntu 20.04.

**BASIC DOS COMMANDS**

1. ***md or mkdir***

Makes a new directory. The parent of the directory specified will be created if it does not already exist.

1. ***cd***

Changes current working directory.

1. ***dir***

Used to list the contents of a directory. used to display a list of all files and subdirectories contained in a specific directory.

1. ***copy***

The copy command allows users to copy one or more files to an alternate location. we can copy files from one directory to another directory.

1. ***del***

The del command is used delete files from a computer.

1. ***edit***

Edit allows a user to view, create, or modify their computer files.

1. ***move***

The move command allows users to transfer files or directories from one directory to another, or from one drive to another

1. ***rn or rename***

Used to rename a file or folder from Command Prompt.

1. ***deltree***

Deletes (erases) a directory including all files and subdirectories that are in it (new with DOS Version 6). Unlike the RMDIR command, the DELTREE command allows to delete a directory even if it contains files and subdirectories.

1. ***cls***

Allows users to clear all the contents on the screen and leave only a prompt

1. ***format***

Formats a disk to accept DOS files.

**BASIC LINUX COMMANDS**

1. ***ls –***

List directory contents. The ls command is used to list out files and directories.

* ls -R will list all the files in the sub-directories as well
* ls -a will show the hidden files
* ls -l will list details of the contents in the directory

It will list down the

* Permissions associated with the file
* The owner of the file
* The group associated with the file
* The size of the file
* The time stamp
* The name of the file

1. ***cd***

Change the current directory.

There are some shortcuts to help you navigate quickly:

* cd ..(with two dots) to move one directory up
* cd to go straight to the home folder
* cd- (with a hyphen) to move to your previous directory

1. ***clear –***

Clear the terminal screen.

1. ***su / sudo***

su command changes the shell to be used as a super user and until you use the exit command you can continue to be the super user

sudo– if you just need to run something as a super user, you can use the sudo command. This will allow you to run the command in elevated rights and once the command is executed you will be back to your normal rights and permissions.

1. ***pwd – Print Working Directory***

It displays the current working directory path and is useful when directory changes are often.

1. ***passwd –***

This command is used to change the user account password. You could change your password or the password of other users. Note that the normal system users may only change their own password, while root may modify the password for any account.

passwd [username]- changes the password for the user.

1. ***mv – Move a file***

To move a file or rename a file.

1. ***cp – Copy a file***

cp source file destination file.

1. ***rm –***

This command is used to remove files in a directory or the directory itself. A directory cannot be removed if it is not empty.

* rm [name of the file]
* rm –r removes all the contents in a directory and the directory as well.

1. ***mkdir – to make a directory.***

mkdir [directory name]

1. ***rmdir – to remove a directory.***

rmdir [directory name]

1. ***chmod –***

To change mode of a file system object. Files can have r – read, w- write and x-execute permissions.

For example:

* chmod mode FILE
* chmod 744 script.sh

The first number stands for the user who is associated with the file

The second number is for the group associated with the file

The third number is associated with everyone else who is not a part of the user or group

1. ***grep –***

Find text in a file. The grep command searches through many files at a time to find a piece of text you are looking for.

grep PATTERN [FILE]

1. ***cat -***

The cat command (short for “concatenate “) is one of the most frequently used commands in Linux. Cat command allows you to create single or multiple files, view contents of file, concatenate files and redirect output in terminal or files.

To create a new file, use the command

cat > filename

Add content

Press 'ctrl + d' to return to command prompt.

To view a file, use the command

cat filename

To combine 2 files

cat file1 file2 > newfilename

1. ***echo –***

This command is used to display a text or a string to the standard output or a file.

1. ***wc-***

Used to find out the number of new lines, word count, byte and characters count in a file specified by the file arguments.

wc [options] filenames.

* wc -w: prints the number of words in a file.
* wc -l: Prints the number of lines in a file.
* wc -c: Displays the count of bytes in a file.
* wc -m: prints the count of characters from a file.
* wc -L: prints only the length of the longest line in a file.

1. ***man –***

Used to view the on-line reference manual pages for commands/programs.

1. ***history –***

Used to show previously used commands or to get information about the commands executed by a user.

1. ***apt –get***

It is used to install new software packages, remove available software packages, upgrade existing software packages as well as upgrade the entire operating system. apt – stands for advanced packaging tool.

1. ***touch***

Used to create empty files. We can create multiple empty files by executing it once.

touch <file name>

touch <file1> <file2>....

1. ***date***

Used to display date, time, time zone, and more.

1. ***cal***

Used to display the current month's calendar with the current date highlighted.

1. ***host***

Used to display the IP address for a given domain name and vice versa. It performs the DNS lookups for the DNS Query.

host <domain name> or <ip address>

1. ***df***

Use df command to get a report on the system’s disk space usage, shown in percentage and KBs. If you want to see the report in megabytes, type df -m.

1. ***diff***

Short for difference, the diff command compares the contents of two files line by line. After analyzing the files, it will output the lines that do not match. Programmers often use this command when they need to make program alterations instead of rewriting the entire source code.

The simplest form of this command is diff file1.ext file2.ext

**SHELL SCRIPTING**

**4.1 Shell script to find largest among three numbers.**

**Program:**

#!/bin/sh

echo "enter first number"

read a

echo "enter second number "

read b

echo "enter third number "

read c

if [ $a -gt $b ] && [ $a -gt $c ]

then

echo "$a is largest number"

elif [ $b -gt $a ] && [ $b -gt $c ]

then

echo "$b is largest number"

elif [ $c -gt $a ] && [ $c -gt $b ]

then

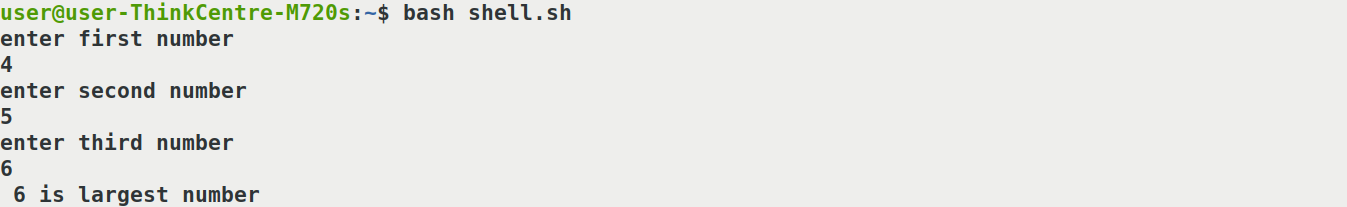
echo " $c is largest number"

else

echo " check the input"

fi

**Output:**



**4.2  Shell script to perform all arithmetic operations using case statement.**

**Program:**

#!/bin/sh

echo enter two numbers

read a

read b

echo 1.add 2. subtract 3. multiply 4. divide

echo enter your choice

read ch

case $ch in

1)

Sum=`expr $a +$b`

echo $sum;;

2)

diff=`expr $a - $b`

echo  $diff;;

3)

product=`expr $a \\* $b`

      echo $product;;

4)

quotient=`expr $a / $b`

echo $quotient;;

\*) echo invalid choice;;

esac

**Output:**



**4.3  Shell script to check whether the given number is prime or not.**

**Program:**

num=29

for((i=2; i<=num/2; i++))

do

  if [ $((num%i)) -eq 0 ]

  then

    echo "$num is not a prime number."

    exit

  fi

done

echo "$num is a prime number."

**Output :**



**4.4 Shell script to print natural numbers from 1 to 10 using while loop.**

**Program:**

#!/bin/sh

n = 0

while [ $n –le 10]

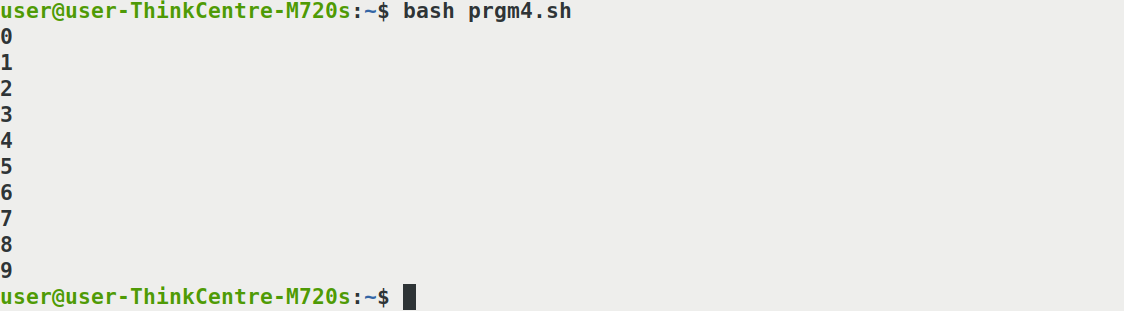
do

echo $n

n= ‘expr $n + 1’

done

**Output:**



**4.5 Shell script to find the sum of first 10 even numbers using for loop.**

**Program;**

**#**!/bin/sh

n = 0

sum=0

for(( i=0; i<=10; i++)) ;

do

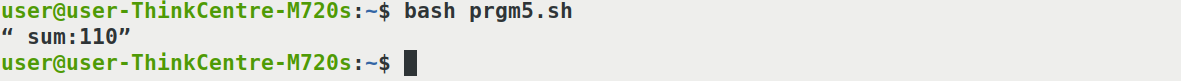
(( sum=sum+n))

((n=n+2))

done

echo “ sum:$sum”

**Output:**



**4.6 Shell script to display the IP address of a set of domain names. Store the domain names in a file and pass the filename as command line argument.**

**Program**

#! /bin/bash

ip\_addresses=$(hostname -I)

ip\_addresses=(${ip\_addresses//" "/ })

echo "LIST OF IP ADDRESSES"

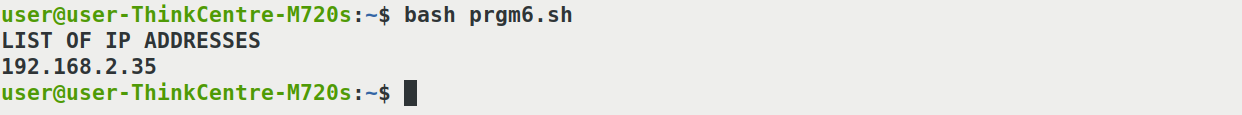
for ip in "${ip\_addresses[@]}";

do

printf "$ip\n"

done

**Output:**



**Network configuration in windows through Network Interface Card (NIC)**

To connect the Xilinx® Zynq® hardware board to the development computer, you must configure an available network connection on the development computer. Follow the steps outlined for your specific operating system.

1. Open the **Control Panel**.
2. Set **View by** to Category.
3. Click **Network and Internet**.
4. Click **Network and Sharing Center**.
5. On the left pane, click **Change adapter settings**.
6. Right-click the local area network connection that is connected to the radio hardware and select **Properties**.
   * If an unused network connection is available, the local area connection appears as Unidentified network.
   * If you plan to reuse your network connection, select the local area connection that you plan to use for the radio hardware.
   * If you have only one network connection, check if you can connect wirelessly to the existing local area network. If you can, you can use the network connection for the radio hardware.
   * You can use a pluggable USB to Gigabit Ethernet LAN adapter instead of a NIC. The instructions are the same.
7. On the **Networking** tab of the **Properties** dialog box, clear all options except **Internet Protocol Version 4 (TCP/IPv4)**. Other services, particularly antiviral software, can cause intermittent connection problems with the radio hardware.
8. Double-click **Internet Protocol Version 4 (TCP/IPv4)**.
9. On the **General** tab, select **Use the following IP Address**.
10. The default IP address of the Xilinx Zynq hardware board is 192.168.1.101. The development computer network connection must be on the same subnet as the hardware board. To meet this requirement, a compatible IP address must be assigned to the development computer network connection. Set the network IP address to 192.168.1.x, where x is any number in the range 1 through 255, apart from 101.

**Note**

Instead of 192.168.1, use the subnet address given by your hardware in the IP address.

1. Leave the subnet mask set to the default value of 255.255.255.0 and click **OK**.

**Network configuration in Linux through Network Interface Card (NIC)**

You can configure a network interface from the command line. You can configure your network client hosts with the command line by using commands to change your current settings or by editing a number of system files. This has several advantages over network managers in the GUI.

This guide had been developed for Ubuntu and Debian, other distributions can use a similar setup, but be aware that file locations may differ.

**Requirements**

An editor. This guide will use vi, but you can replace it with any other editor of your liking, gedit, nano, gvim, pico, emacs..

Have a working internet connection during the installation. We need to install packages and those are fetched from the online repositories.

Backup any files we touch. Just easier to recover when you have a backup.

Read this guide in full before actually configuring your network.

**STEPS**

## 1.Setting/changing the hostname

The hostname command allows you to directly query, or set, the hostname from the command line.

You can see your current hostname by running hostname. To set the hostname directly you can run hostname newname as root. If you do this however, make sure you change the /etc/hosts file first. This is needed because otherwise you will need to boot into single user mode and change your hostname in /etc/hosts. This is because sudo requires DNS lookups in certain cases and it cannot resolve your newly set hostname.. To circumvent this you need to add the new name to the hosts file prior to setting the hostname and remove it once that action has succeeded.

sudo vi /etc/hosts

127.0.1.1 ubuntu newname

Set the hostname to newname.

sudo hostname newname

# or

echo newname | sudo tee /etc/hostname

Now you can remove the old hostname

sudo vi /etc/hosts

127.0.1.1 newname

When your system boots it will automatically read the hostname from the file /etc/hostname. You can add hostname and IP addresses to the file /etc/hosts for static lookups.

## 2.Finding your network interface

When setting up your network you will need to know the network interface cards on your computer. The interface name of cards for different vendors may be different, which is why this step is needed.

ls /sys/class/net

# or

ip addr

This will list the interface names for all NICs on your computer. It will probably include eth0 (hardwired NIC), lo (loopback interface for the localhost), and something for your wireless card (like wifi0, or wlan0).

## 3.Disable network managers and/or wicd

Configuring your network via the CLI will likely interfere with Network Manager or wicd. To counter this, disable the startup script of either application or completely remove the packages. Be aware that the rest of this guide will require packages which need to be installed from the online repositories.

## Network manager

# Stop the current process

sudo /etc/init.d/NetworkManager stop

# Start: sudo /etc/init.d/NetworkManager start

# The inofficial way:

sudo chmod -x /etc/init.d/NetworkManager

# Reverse: sudo chmod +x /etc/init.d/NetworkManager

# The official way:

sudo update-rc.d -f NetworkManager remove

# Reverse: sudo update-rc.d -f NetworkManager defaults 50

# Or remove the network manager package.

sudo aptitude purge network-manager

# Reverse: sudo aptitude install network-manager

## wicd

# Stop the current process

sudo /etc/init.d/wicd stop

# Start: sudo /etc/init.d/wicd start

# The inofficial way:

sudo chmod -x /etc/init.d/wicd

# Reverse: sudo chmod +x /etc/init.d/wicd

# The official way:

sudo update-rc.d -f wicd remove

# Reverse: sudo update-rc.d -f wicd defaults 20

# Or remove the wicd package.

sudo aptitude purge network-manager

# Reverse: sudo aptitude install network-manager

Note: At some point Ubuntu started using upstart instead of the /sbin/init/ daemon. Thus if the first command above does not stop Network Manager, try this.

sudo stop network-manager

## 4.Configuring an interface

**Word of wisdom**

We will use eth0 in this example, your interface can be named differently, see Finding your network interface.

If you have disabled the either wicd or the network manager you probably don't have a network connection anymore. Connect via a regular UTP cable to your router, and assuming you have DHCP enabled do the following:

sudo ip link set dev eth0 down

sudo dhclient eth0

This will bring your eth0 up by using DHCP. Your network is now configured (for the time being).

If you don't have DHCP enabled configure your network by issueing the commands below, the gateway address is the IP address of your router. And your IP should be in the same range as the router is.

sudo ip addr add 192.168.1.14/24 dev eth0

sudo ip link set dev eth0 up

sudo ip route add default via 192.168.1.1

These commands configure your interface but these changes will not survive a reboot, since the information is not stored anyhwere. This is where the interfaces file comes in handy. To configure a interface permanently you'll need to edit the interfaces file, /etc/network/interfaces.

sudo vi /etc/network/interfaces

## To configure a dynamic IP address

auto eth0

iface eth0 inet dhcp

## Or configure a static IP

auto eth0

iface eth0 inet static

address 192.168.1.14

gateway 192.168.1.1

netmask 255.255.255.0

network 192.168.1.0

broadcast 192.168.1.255

For these settings to take effect you need to restart your networking services.

sudo /etc/init.d/networking restart

## 5. Setting up a second IP address or Virtual IP address

If you need to set up a second ip address you need to edit the /etc/network/interfaces.

sudo vi /etc/network/interfaces

auto eth0:1

iface eth0:1 inet static

address 192.168.1.24

netmask 255.255.255.0

network 192.168.1.1

broadcast 192.168.1.255

gateway 192.168.1.1

For these new settings to take effect you need to restart networking services using the following command

sudo /etc/init.d/networking restart

## 6. setting up MTU for a connection

You can set the MTU for an interface, you could do this by using the mtu keyword in the interface file or by using the ip link command.

iface eth0 inet dhcp

# via mtu keyword

mtu 1492

# Via ip link command

pre-up /sbin/ip link set $IFACE mtu 1492

The above example sets the MTU for device eth0 to 1492, the usual MTU for a PPPoE ISP connection. This however is only needed if connections seem to hang otherwise (with the default of 1500).

This tip was found on <http://glasnost.beeznest.org/articles/290>.

## 7. Setting up DNS

To cause your machine to consult with a particular server for name lookups you simply add their addresses to /etc/resolv.conf.

For example a machine which should perform lookups from the DNS server at IP address 192.168.1.1 would have a resolv.conf file looking like this

sudo vi /etc/resolv.conf

enter the following details

search example.com

domain example.com

nameserver 192.168.1.1

# In the Devices tab, click the Ethernet connection.

1. Click **Activate**.
2. Click **Yes**to confirm the network card activation.

To configure the network card from the command line interface:  
The below command is just an example and this may vary depending on the guest operating system type.

* The acpiphp module must be loaded for Linux to detect hot-added devices.  
    
  To load the acpiphp module, run this command:  
    
  modprobe acpiphp
* Run the kudzu command to ensure that the new hardware is detected and configured.
* Run this command to ensure that the network card is enabled:

ifup eth2  
  
**Note**: eth2 is the new network card that has been added.

* To configure the network card, edit the ifcfg-eth2 file using a text editor.  
    
  To edit the file using vi editor, run this command:

          Example below :-

# vi /etc/sysconfig/network-scripts/ifcfg-eth2

**Telnet and FTP**

[TELNET (TELecommunication NETwork)](https://practice.geeksforgeeks.org/problems/explain-telnet) and [FTP (File Transfer Protocol)](https://www.geeksforgeeks.org/computer-network-file-transfer-protocol-ftp/) both are the application layer protocol. They are connection oriented protocols as they create a connection between remote host and a server. AS name File Transfer Protocol, FTP is used for transferring the files from one system to another system. and TELNET is used for remote login for a system.

**Telnet**

A terminal emulation that enables a user to connect to a remote [host](https://www.computerhope.com/jargon/h/hostcomp.htm) or [device](https://www.computerhope.com/jargon/d/device.htm) using a telnet client, usually over [port](https://www.computerhope.com/jargon/p/port.htm) 23. For example, typing **telnet hostname** would connect a user to a [hostname](https://www.computerhope.com/jargon/h/hostname.htm) named **hostname**. **Telnet** enables a user to manage an account or device remotely. For example, a user may telnet into a computer that hosts their [website](https://www.computerhope.com/jargon/w/website.htm) to manage his or her [files](https://www.computerhope.com/jargon/f/file.htm) remotely. A telnet session is a [command line](https://www.computerhope.com/jargon/c/commandi.htm) interface.

**To use telnet, follow the steps below:**

1. First, find out the ip address of the server/main computer. For this you need to access the server and use the ipconfig command in MS-DOS. See Additional Information section for more details about this command.
2. Select the **Windows** key and the **R** key.
3. In the Run box type **CMD.**
4. Select **OK.**
5. Type **Telnet <IP Address> 13531.**

**Note:** Do not include the <> when entering the IP Address.

1. If you see a blank cursor then the connection is fine. You can close the command prompt window.
2. If you get the message that 'telnet' is not recognized as an internal or external command, operable program or batch file. you will want to enable Telnet. See Additional Information on how to Enable telnet.
3. If you get an error or are unable to telnet to the server please contact your Network Administrator.

**FTP**

File transfer protocol is a way to download, upload, and transfer files from one location to another on the internet and between computer systems. FTP enables the transfer of files back and forth between computers or through the cloud. Users require an internet connection in order to execute FTP transfers. FTP is an essential tool for those who build and maintain websites. Many FTP clients are free to download, although most websites already have the FTP built-in.

## How to Open an ftp Connection to a Remote System

1. Ensure that you have ftp authentication.

You must have ftp authentication, as described in [Authentication for Remote Logins (ftp)](https://docs.oracle.com/cd/E19253-01/816-4555/remotehowtoaccess-28951/index.html).

1. Open a connection to a remote system by using the ftp command.

|  |
| --- |
| $ **ftp** remote-system |

1. If the connection succeeds, a confirmation message and prompt are displayed.
2. Type your user name.

|  |
| --- |
| Name (remote-system:user-name): user-name |

1. If prompted, type your password.

|  |
| --- |
| Password required for user-name:  Password: password |

1. If the system you are accessing has an established anonymous ftp account, you are prompted for an email address for the password. If the ftp interface accepts your password, it displays a confirmation message and the (ftp>) prompt.
2. You can now use any of the commands that are supplied by the ftp interface, including help.

This ftp session was established by the user smith on the remote system pluto:

|  |
| --- |
| $ **ftp pluto**  Connected to pluto.  220 pluto FTP server ready.  Name (pluto:smith): **smith**  331 Password required for smith:  Password: password  230 User smith logged in.  ftp> |

## How to Copy Files to a Remote System (ftp)

1. Change to the source directory on the local system.

The directory from which you type the ftp command is the local working directory, and thus the source directory for this operation.

1. Establish an ftp connection.
2. Change to the target directory.

|  |
| --- |
| ftp> **cd** target-directory |

1. Remember, if your system is using the automounter, the home directory of the remote system's user appears parallel to yours, under /home.
2. Ensure that you have write permission to the target directory.

|  |
| --- |
| ftp> **ls -l** target-directory |

1. Set the transfer type to binary.

|  |
| --- |
| ftp> **binary** |

1. To copy a single file, use the put command.

|  |
| --- |
| ftp> **put** filename |

1. To copy multiple files at once, use the mput command.

|  |
| --- |
| ftp> **mput** filename [filename ...] |

1. You can supply a series of individual file names and you can use wildcard characters. The mput command copies each file individually, asking you for confirmation each time.
2. To close the ftp connection, type bye.

|  |
| --- |
| ftp> **bye** |

In this example, the user kryten opens an ftp connection to the system pluto, and uses the put command to copy a file from his or her system to the /tmp directory on system pluto.

|  |
| --- |
| $ **cd /tmp**  **ftp pluto**  Connected to pluto.  220 pluto FTP server (SunOS 5.8) ready.  Name (pluto:kryten): **kryten**  331 Password required for kryten.  Password: **xxx**  230 User kryten logged in.  ftp> **cd /tmp**  250 CWD command successful.  ftp> **put filef**  200 PORT command successful.  150 ASCII data connection for filef (129.152.221.238,34356).  226 Transfer complete.  ftp> **ls**  200 PORT command successful.  150 ASCII data connection for /bin/ls (129.152.221.238,34357) (0 bytes).  dtdbcache\_:0  filea  filef  files  ps\_data  speckeysd.lock  226 ASCII Transfer complete.  60 bytes received in 0.058 seconds (1.01 Kbytes/s)  ftp> **bye**  221 Goodbye. |

**EXERCISE ON HTML**

**8.1 Design a simple web page using basic tags and formatting tags.**

**[Hint: Header tags(h1,h2,...), paragraph tag(<p>), line break(<br />), font tag(<font>), bold tag(<b></b>, preformatted tag(<pre>)etc. Document prepration]**

**Program:**

<!DOCTYPE HTML>

<html>

<head>

<title>simple web page</title>

</head>

<body>

<h1 style="text-align:center,">MY HOME</h1>

<h2><u>"Heart is where the home is"</u></h2>

<P><Pre><h2><b>My home is brown and biege in colour.My home consist of 5 rooms, a kitchen and 2 bathroom.<br>

Our house becomes a happy and a sweet when we have our lovedones with us.

<br>we always keep our home neat and clean.My home is the only place where I feel safe and very comfortable.

My home is located in Bandra on carter road.I live in a magnificent row house,

which flaunts a small yet dainty garden at the entrance to my home.<br>

As you enter you will see an elegant formal living room attatched to which is our stylish and chic family dining room.

Adjacent to our munching area is a healthy

kitchen that is a fashionable open one.

<br>there are translucent curtains running throughout the house and the entire house is

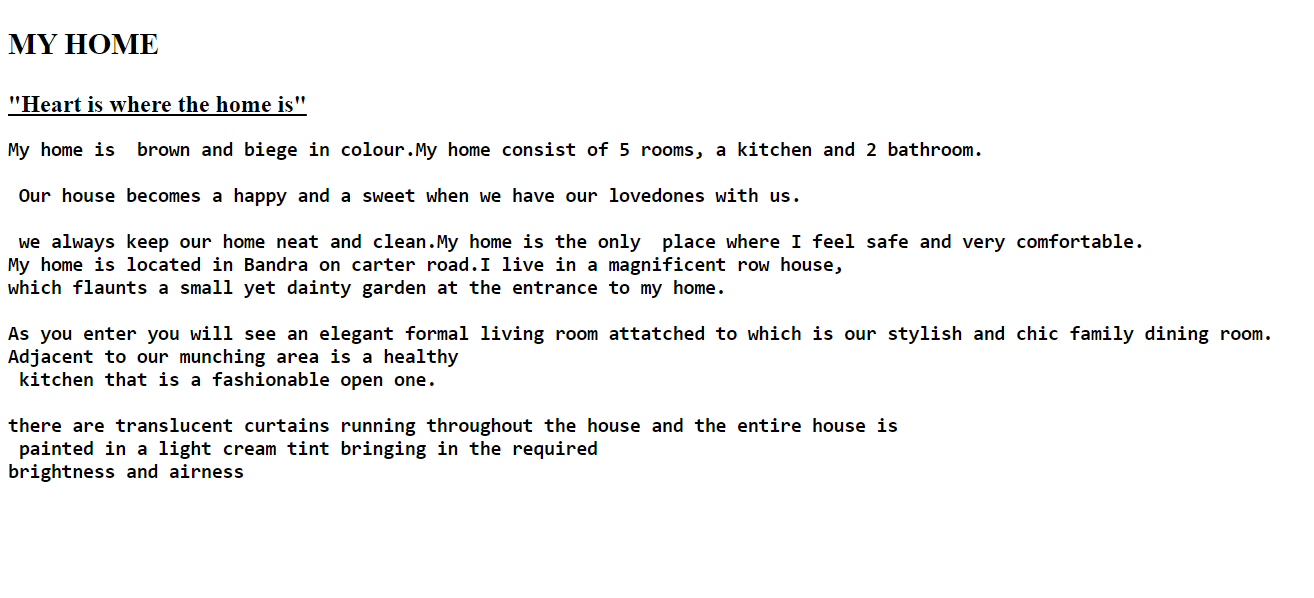
painted in a light cream tint bringing in the required

brightness and airness</b></h2></P>

</body>

</html>

**Output:**



**8.2 Design a simple web page using frameset for a Restaurant Menu. Create minimum four frames. Each frame can be used to mention different categories of food items. Use list tags and image tag with attributes.**

**Program:**

<!DOCTYPE html>

<html>

<head>

<title>RESTAURANT MENU </title>

</head>

<body>

<frameset cols="25%,25%,25%,25%">

<frame name="VEGETARIAN" src="vegetarian.html">

<frame name="NON-VEGETARIAN" src="nonvegetarian.html">

<frame name="JUICE&SHAKES" src="juice&shakes.html">

<frame name="DESSERT" src="dessert.html">

</frameset>

</body>

</html>

**FRAME 1 – VEGETARIAN**

<!DOCTYPE html>

<html>

<head>

<title>Frame1</title>

</head>

<body>

<h2 style="text-align:center;">VEGETARIAN</h2>

<img src="images/veg.jpg" alt="vegimage" width="300" height="200">

<ol>

<li>INDIAN

<ul>

<li>Chickpea Curry</li>

<li>Dal Curry</li>

<li>Gobi 65</li>

<li>Jeera Rice</li>

<li>Palak Paneer</li>

</ul>

</li>

<li>ITALIAN

<ul>

<li>Italian Oven Roasted Vegetables</li>

<li>Italian Vegetable Soup</li>

<li>Italian Ratatouille</li>

<li>Pasta Primavera</li>

<li>Veggie Skillet</li>

</ul>

</li>

<li>CHINESE

<ul>

<li>Chinese Vegetable Stir Fry</li>

<li>Chinese Vegetable Fried Rice</li>

<li>Choy Sum</li>

<li>Vegetables In Hot Garlic Sauce</li>

<li>Vegetable Kung Pao</li>

</ul>

</li>

</ol>

</body>

</html>

**FRAME 2 – NON VEGETARIAN**

<!DOCTYPE html>

<html>

<head>

<title>Frame2</title>

</head>

<body>

<h2 style="text-align:center;">NON VEGETARIAN</h2>

<img src="images/nonveg.jpeg" alt="nonvegimage" width="300" height="200">

<ol>

<li>CHICKEN

<ul>

<li>Butter Chicken</li>

<li>Chicken Biriyani</li>

<li>Chicken Nuggets</li>

<li>Chicken Shawarma</li>

<li>Chicken 65</li>

</ul>

</li>

<li>EGG

<ul>

<li>Egg Bhurji</li>

<li>Egg Butter Masala</li>

<li>Egg Fried Rice</li>

<li>Egg Noodles</li>

<li>Egg Masala</li>

</ul>

</li>

<li>FISH & SEA FOOD

<ul>

<li>Baked Tilapia</li>

<li>Salmon With Creamy Dill sauce</li>

<li>Tuna Crescent Ring</li>

<li>Lobster Mac & Chesse</li>

<li>Prawn Tikka Masala</li>

</ul>

</li>

</ol>

</body>

</html>

**FRAME 3 - JUICE&SHAKES**

<!DOCTYPE html>

<html>

<head>

<title>Frame3</title>

</head>

<body>

<h2 style="text-align:center;">JUICE&SHAKES</h2>

<img src="images/juice&shakes.jpg" alt="Juice&shakesimages" width="300" height="200">

<ol>

<li>JUICE

<ul>

<li>Carrot-Orange Juice</li>

<li>Mango Juice</li>

<li>Pineapple Juice</li>

<li>Strawberry-Apple Juice</li>

<li>Super Green Juice</li>

</ul>

</li>

<li>SHAKES

<ul>

<li>Cashew Fig Milkshake</li>

<li>China Grass Milkshake</li>

<li>Dark Chocolate Milkshake </li>

<li>Tender Coconut Milkshake</li>

<li>Vanila Protein Shake</li>

</ul>

</li>

</ol>

</body>

</html>

**FRAME 4 – DESSERT**

<!DOCTYPE html>

<html>

<head>

<title>Frame4</title>

</head>

<body>

<h2 style="text-align:center;">DESSERT</h2>

<img src="images/dessert.jpg" alt="dessertimage" width="300" height="200">

<ol>

<li>CAKES

<ul>

<li>Chocolate Mousse Cake</li>

<li>Eclair Cake</li>

<li>Hot Fudge Chocolate Cake</li>

<li>Ice Cream Cake</li>

<li>Mini Blueberry Mousse Cake with Mirror Glaze</li>

</ul>

</li>

<li>ICECREAM

<ul>

<li>Caramel Topped Icecream</li>

<li>Fudge Icecream Dessert</li>

<li>Hot Fudge Icecream Bar Dessert</li>

<li>Cookie Icecream</li>

<li>Mixed Fruits Custard Icecream</li>

</ul>

</li>

<li>SWEETS

<ul>

<li>Gulab Jamun</li>

<li>Gajar Ka Halwa</li>

<li>Kaju Ki Barfi</li>

<li>Sandesh</li>

<li>Modak</li>

</ul>

</li>

</ol>

</body>

</html>

**Output :**



**8.3 Design a web page containing a table of student list of your class. The table should have a title, and Roll no, Name and qualifying degree of each student.**

**Program:**

<html lang="en">

<head>

<title>Student Details</title>

<style>

table, th, td {

border: 1px solid black;

}

</style>

</head>

<body>

<table>

<tr>

<th>Roll No.</th>

<th>Name</th>

<th>Qualifying Degree</th>

</tr>

<tr>

<td>001</td>

<td>Radul R Dev</td>

<td>BSc. Botany</td>

</tr>

<tr>

<td>002</td>

<td>Shasna Serin</td>

<td>BSc. Botany</td>

</tr>

<tr>

<td>003</td>

<td>Keerthana T P</td>

<td>BSc. Zoology</td>

</tr>

<tr>

<td>004</td>

<td>Fathimath Henna T P</td>

<td>BSc. Chemistry</td>

</tr>

<tr>

<td>005</td>

<td>Amal Krishnan C</td>

<td>BSc. Plant Science</td>

</tr>

<tr>

<td>006</td>

<td>Aswini N</td>

<td>BSc. Zoology</td>

</tr>

<tr>

<td>007</td>

<td>Nayana C P</td>

<td>BSc. Zoology</td>

</tr>

<tr>

<td>008</td>

<td>Anusree Babu P</td>

<td>BSc. Zoology</td>

</tr>

<tr>

<td>009</td>

<td>Kajal Prakash </td>

<td>BSc. Physics</td>

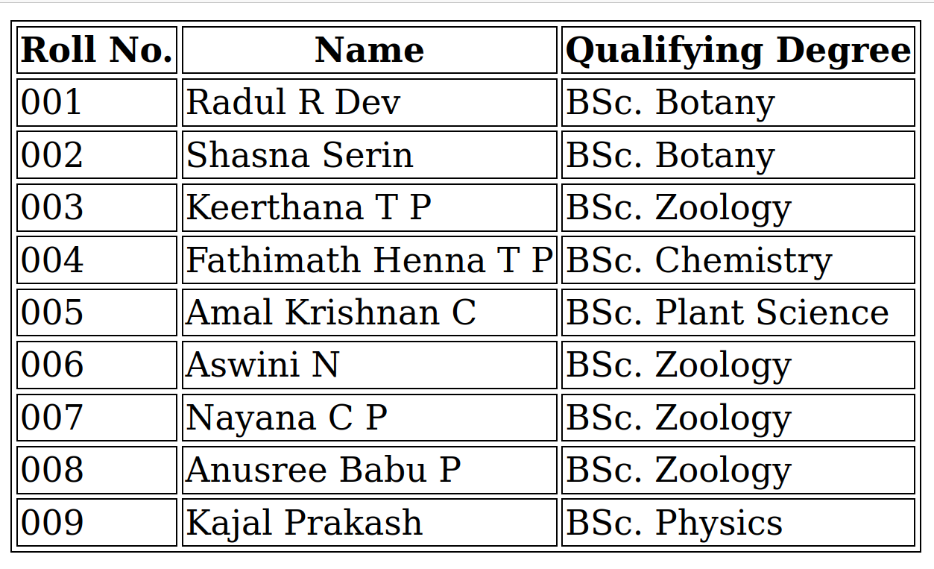
</tr>

</table>

</body>

</html>

**Output:**



**8.4 Design simple login page using form with attributes. The form should also contain submit button and reset button.**

**Program:**

<Html>

<head>

<title >html form</title>

</head>

<body>

<body bgcolor="skyblue">

<form action=”” >

<p>

<label> Name:

<input type =”text” name=”Name” size=”30”/>

<br/><br/>

<label>Roll no:

<input type =”text” name=”roll” size=”10”/>

<br/><br/>

<label>Password:

<input type =”text”/>

<br/><br/>

<label>Gender : </label>

<input type="radio" name="male"/> Male

<input type="radio" name="female"/> Female

<input type="radio" name="other"/> Other

<br>

<br>

<button type="submit" value="Login ">Login</button>

<button type="submit" value="Reset">Reset</button>

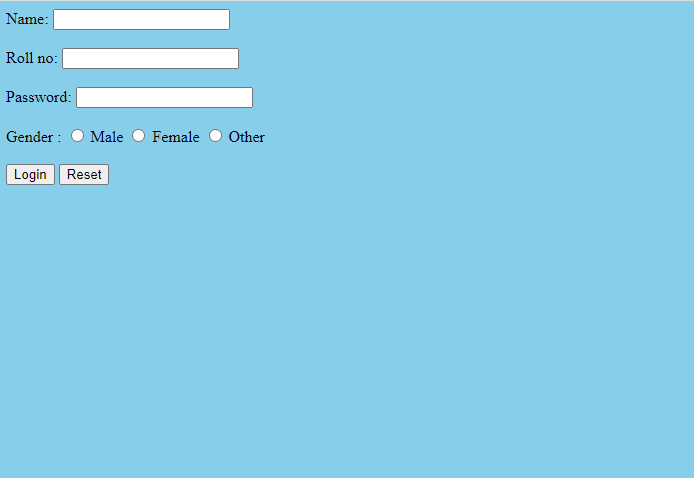
</p>

</form>

</body>

</html>

**Output:**



**8.5 Design simple PG admission registration form using all form tags**

**Program:**

<!DOCTYPE HTML>

<Html>

<head>

<title> Registration Page </title>

</head>

<h2>PG REGISTRATION FORM</h2>

<body bgcolor="skyblue">

<br>

<br>

<form>

<label> Firstname </label>

<input type="text" name="firstname" size="15"/> <br> <br>

<label> Lastname: </label>

<input type="text" name="lastname" size="15"/> <br> <br>

<label>Course : </label>

<select>

<option value="Course">Course</option>

<option value="BCA">MSC</option>

<option value="BBA">MA</option>

<option value="MBA">MBA</option>

<option value="MCA">MCA</option>

<option value="M.Tech">M.Tech</option>

</select>

<br> <br>

<label> Date of birth</label>

<input type="text" name="Date of birth" size="15"/>

<br> <br>

<label>Gender : </label>

<input type="radio" name="male"/> Male

<input type="radio" name="female"/> Female

<input type="radio" name="other"/> Other

<br> <br>

<label>Hobbies : </label>

<br>

<input type="checkbox" name="sports"/> sports

<input type="checkbox" name="reading"/> reading

<input type="checkbox" name="other"/> Other

<br>

Graduation details:

<br>

<textarea cols="60" rows="4" value="Graduation details">

</textarea>

<br> <br>

<label>Phone no : </label>

<input type="text" name="phone" size="10"/> <br> <br>

Address

<br>

<textarea cols="60" rows="5" value="address">

</textarea>

<br><br>

Email:

<input type="email" id="email" name="email"/> <br> <br>

<button type="submit" value="submit">Submit</button>

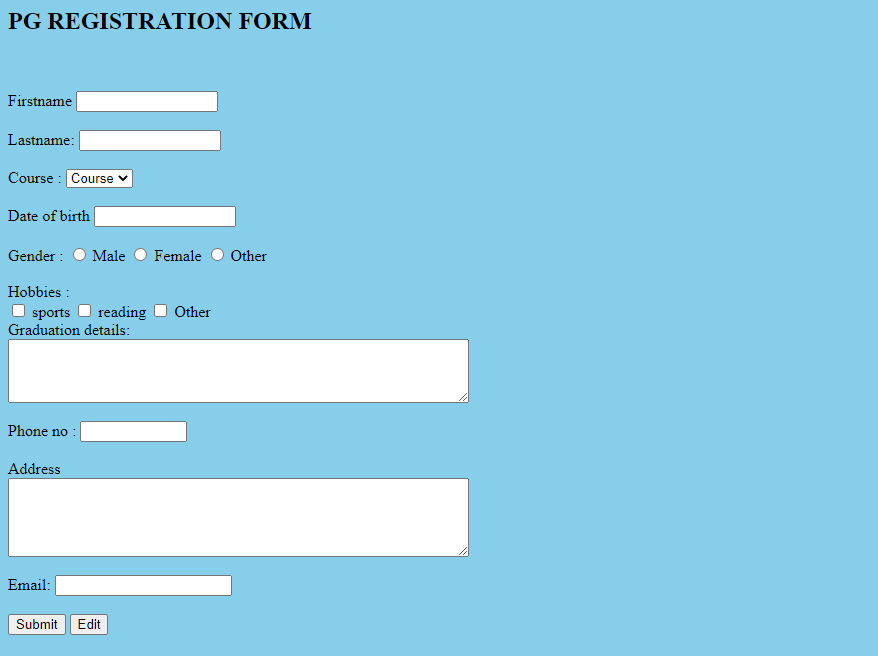
<button type="submit" value="Edit">Edit</button>

</form>

</body>

</html>

**Output:**



**8.6 Design a web page to enter two numbers and display their sum. Use JavaScript to find the sum of entered numbers.**

**Program:**

<html>

<head>

<title>Javascript</title>

<h1>Programme to find sum of two numbers using Javascript</h1>

<script>

function add()

{

var a,b,c

a= Number (document.getElementById("first").value);

b= Number (document.getElementById("second").value);

var c=a+b;

document.getElementById("answer").value= c;

}

</script>

</head>

<body style="background-color:powderblue;">

Enter the first number: <input id="first"> </p>

Enter the second number: <input id="second"> </p>

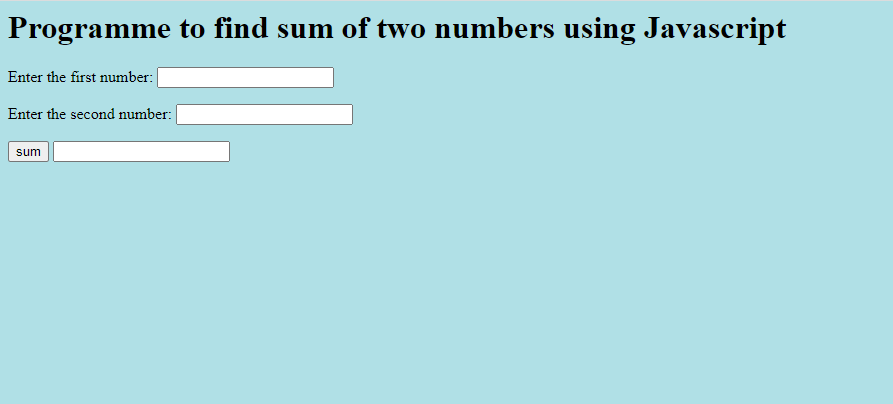
<button onclick="add()"> sum </button>

<input id="answer"> </p>

</body>

</html>

**Output:**



**8.7 Design simple website using hyperlink.**

**Program 1:**

<html>

<head>

<title>link</title>

<metacharset ="utf-8"/>

</head>

<body>

<h1>TRAVELS</h1>

<ul><p><li>Our Travel Leaders agents are experts on specific cruise lines, hotels, resorts, tour operators and more. These agents have received education on and personally sailed on the ships, and stayed in the hotels and resorts. </p></li>

<li> <p>Our Travel Leaders agents are certified specialists who have completed training and traveled worldwide to most popular destinations, such as maldive and the paris.</p></li>

<li>Aller deplacer is one of the most respected and successful names in the global travel industry,with a long and rich company heritage spanning 15 years</br>

<li>we have been travelling in the world for over 15 years building a reputation for providing quality travel content that's inspiring,engaging and informative<br/>

<li><p>we improves your health. </p><br>

<br>

<br>

<img src="images/manaliresort2.jfif"cols="400" rows="400"alt="">

<img src="images/resort6.jfif"cols="400" rows="400"alt="">

<img src="images/balcony.jfif"cols="400" rows="400"alt="">

<img src="images/balcony2.jfif"cols="450" rows="350"alt="">

<br/>

<br/>

<h2>THANK YOU </h2>

<a href="hyperlink1.html"><button class="btn btn1">readmore</button></a>

</ul>

</body>

</html>

**Program 2:**

<!DOCTYPE HTML>

<HTML>

<head>

<title>hyperlink</title>

</head>

<BODY>

<section id="about" class="about">

<h1 class="heading"><span>a</span>bout <span>u</span>s</h1>

<div class="row">

<div class="image">

</div>

<div class="content">

<h3>why choose us?</h3>

<ul>

<li> <p>Our Travel Leaders agents are experts on specific cruise lines, hotels, resorts, tour operators and more. These agents have received education on and personally sailed on the ships, and stayed in the hotels and resorts. </p></li>

<li> <p>Our Travel Leaders agents are certified specialists who have completed training and traveled worldwide to most popular destinations, such as Australia and the islands of Hawaii.</p></li>

</ul>

</div>

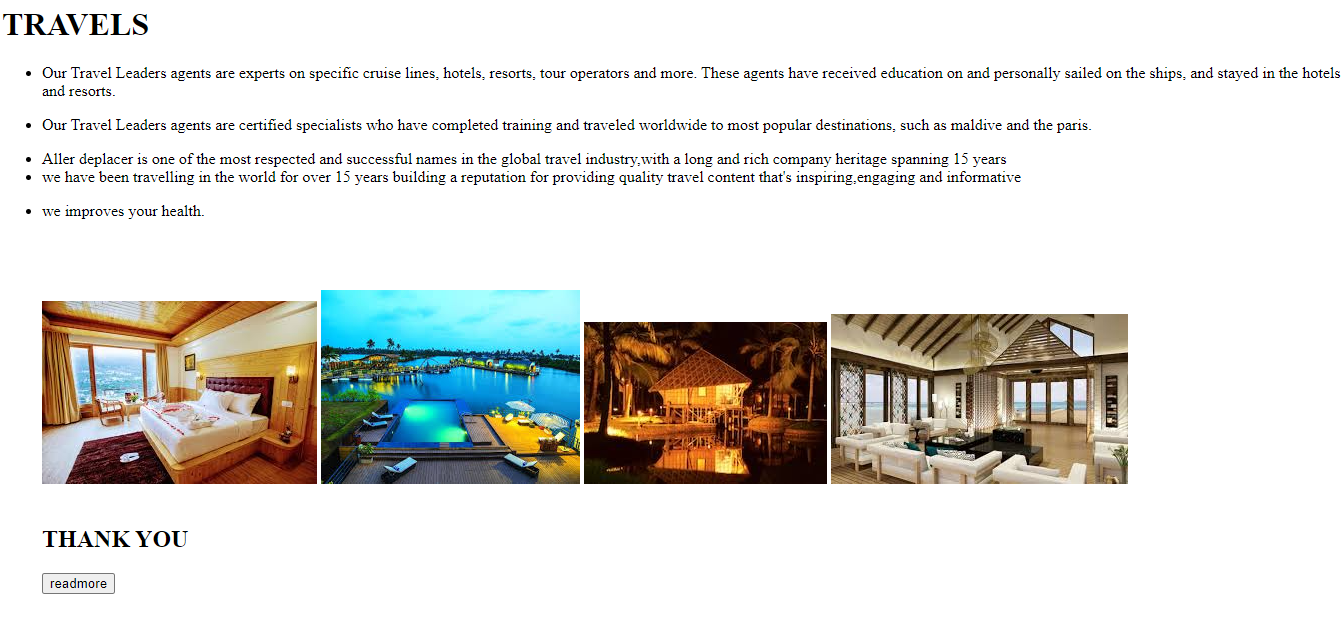
</div>

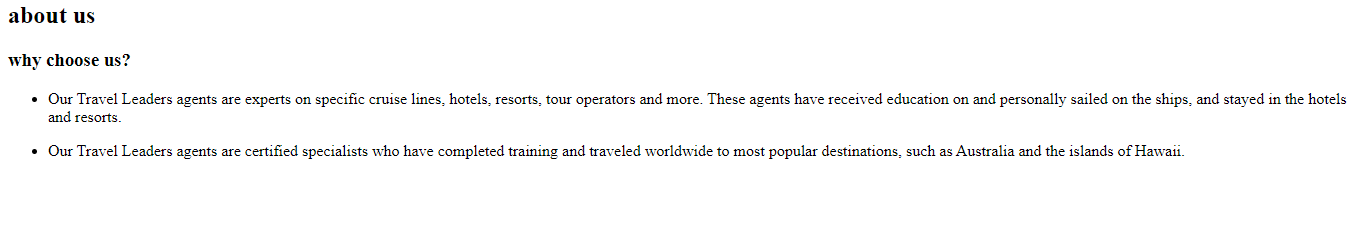
</section>

</body>

</html>

**Output:**





1. **APPLICATIONS OF**

**R PROGRAMMING**

**INSTALLING R AND RSTUDIO**

R is an open-source programming language and free environment that specializes in statistical computing and graphical representation. It is mainly used by statisticians and data miners for developing statistical software and performing data analysis.

R is maintained by an international team of developers who make the language available through the web page of The Comprehensive R Archive Network. The top of the web page provides three links for downloading R. Follow the link that describes your operating system: Windows, Mac, or Linux.

**Linux**

R comes preinstalled on many Linux systems, but may be out of date. The CRAN website provides files to build R from source on Debian, Redhat, SUSE, and Ubuntu systems under the link “Download R for Linux.” Click the link and then follow the directory trail to the version of Linux you wish to install on. The exact installation procedure will vary depending on the Linux system you use. CRAN guides the process by grouping each set of source files with documentation or README files that explain how to install on your system.

1. **Steps to Install R on Ubuntu 20.04 put this heading in index page also**

**Prerequisites**

* Your system has at least 1G of RAM. Otherwise, create a swap file.
* You are logged in as a user with sudo privileges.

The R packages included in the default Ubuntu repositories are often outdated. So install R from the CRAN repository.

**Step1**: Install the dependencies necessary to add a new repository over HTTPS:

*$ sudo apt install dirmngr gnupg apt-transport-https ca-certificates software-properties-common*

**Step2**: Add the CRAN repository to your system sources’ list:

*$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys* E298A3A825C0D65DFD57CBB651716619E084DAB9

*$ sudo add-apt-repository 'deb https://cloud.r-project.org/bin/linux/ubuntu focal-cran40/'*

**Step3**: Install R by typing:

*$ sudo apt install r-base*

**Step4**: The installation may take a few minutes to complete. Once completed, verify it by printing the R version:

*$ R –version*

**Output**

R version 4.0.1 (2020-06-06) -- "See Things Now"

Copyright (C) 2020 The R Foundation for Statistical Computing

Platform: x86\_64-pc-linux-gnu (64-bit)

R has been installed and can start using it.

1. **Using R put this heading in index page also**

R is a computer language, like C, C++, or UNIX. You use R by writing commands in the R language and asking your computer to interpret them. In the old days, people ran R code in a UNIX terminal window. Now almost everyone uses R with an application called Rstudio.

**R and UNIX**

You can still run R in a UNIX or BASH window by typing the command:

*R*

which opens an R interpreter. You can then do your work and close the interpreter by running

*q()*

when you are finished.

1. **Rstudio put this heading in index page also**

RStudio is an integrated development environment (IDE) for R. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management. The RStudio interface looks the same for Windows, Mac OS, and Linux.

You can download Rstudio for free. Just click the “Download RStudio” button and follow the simple instructions that follow. Once you’ve installed RStudio, you can open it like any other program on your computer—usually by clicking an icon on your desktop.

**2. Mathematical operations and string manipulation**

2.1 **Mathematical operations (Calculator )**

**Program:**

add <- function(x, y) {

  return(x + y)

}

subtract <- function(x, y) {

  return(x - y)

}

multiply <- function(x, y) {

  return(x \* y)

}

divide <- function(x, y) {

  return(x / y)

}

modulo <- function(x,y){

  return(x %% y)

}

exponent <- function(x,y){

  return(x \*\* y)

}

print("Select operation.")

print("1.Add")

print("2.Subtract")

print("3.Multiply")

print("4.Divide")

print("5.Modular Division")

print("6.Exponent")

opt = as.integer(readline(prompt="Enter option[1/2/3/4/5/6]: "))

num1 = as.integer(readline(prompt="Enter first number: "))

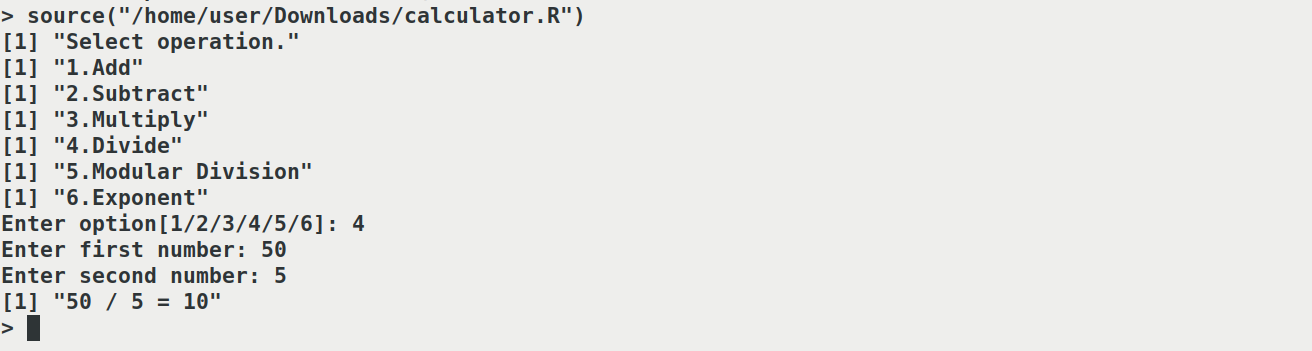
num2 = as.integer(readline(prompt="Enter second number: "))

operator <- switch(opt,"+","-","\*","/","%%","\*\*")

result <- switch(opt, add(num1, num2), subtract(num1, num2), multiply(num1, num2), divide(num1, num2),modulo(num1,num2),exponent(num1,num2))

print(paste(num1, operator, num2, "=", result))

**Output:**



# 2.2. STRING MANIPULATION IN R PROGRAMING

**Program:**

d <- "computationalbiology"

g <- c("department","of","biotechnology","and","microbiology")

print(paste("word is ",d))

print("list of words")

print(g)

print("")

substring\_of\_d <- substr(d,0,13)

print(paste("substring extraxted of d is: ",substring\_of\_d))

presence <-grep("biology",d)

print(paste("substring present in d: ",presence))

present <-grepl("biology",d)

print(paste("substring present in d: ",present))

sentance <- paste(g, collapse = " ")

print(paste("joining list of words to form a sentance :",sentance))

split\_string <- strsplit(sentance," ")

print("spliting a sentance to list of words:")

print(split\_string)

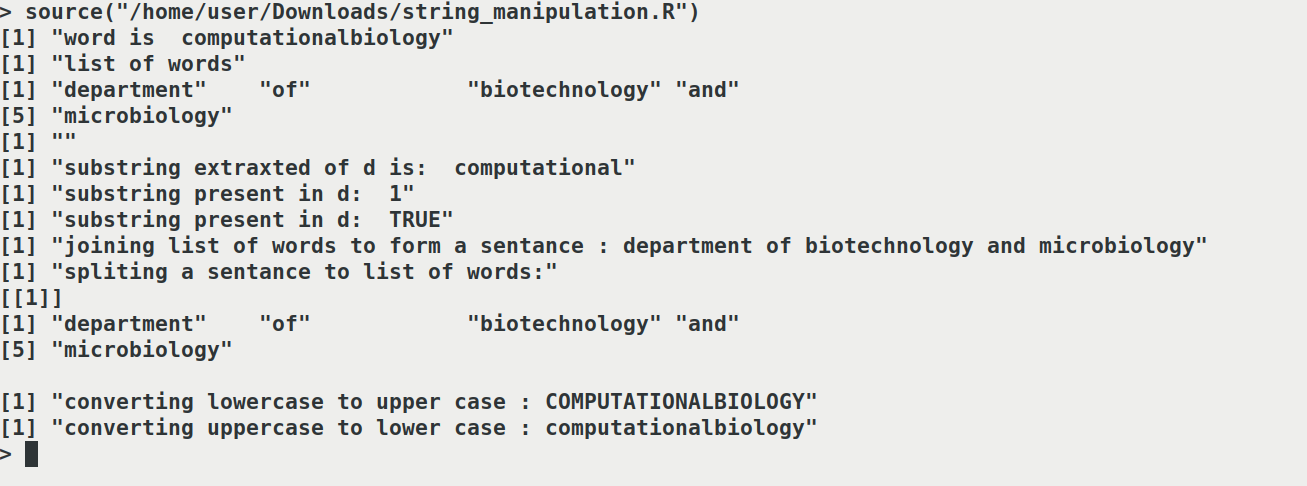
upperCase <- toupper(d)

print(paste("converting lowercase to upper case :",upperCase))

lowerCase <- tolower(upperCase)

print(paste("converting uppercase to lower case :",lowerCase))

**Output:**



**8.STATISTICAL ANALYSIS USING R PROGRAMMING**

**Program:**

marks <- c(98,27,56,65,59,35,82,69,45,25,87,75,64,59,92,80,67,65,58,65)

print("observations")

print(marks)

data\_count <- length(marks)

print(paste("no of observation ",data\_count))

minimum\_marks <- min(marks)

print(paste("minimum marks : ",minimum\_marks))

maximum\_marks <- max(marks)

print(paste("maximum marks : ",maximum\_marks))

range\_of\_marks <- range(marks)

print(paste("range of marks : ",range\_of\_marks))

cumilative\_marks <- sum(marks)

print(paste("sum of marks : ",cumilative\_marks))

median\_value <- median(marks)

print(paste("median : ",median\_value))

class\_average <- mean(marks)

print(paste("class avarage  : ",class\_average))

absolute\_mark <- abs(class\_average)

print(paste("absolute value of class average  : ",absolute\_mark))

trunctated\_decimal\_piont <- trunc(class\_average)

print(paste("truncating decimal point of class average  : ",trunctated\_decimal\_piont))

round\_marks\_next\_small\_whole\_number <- floor(class\_average)

print(paste("round class average lower whole number  : ",round\_marks\_next\_small\_whole\_number))

round\_marks\_next\_largest\_whole\_number <- ceiling(class\_average)

print(paste("round class average to higher whole number  : ",round\_marks\_next\_largest\_whole\_number))

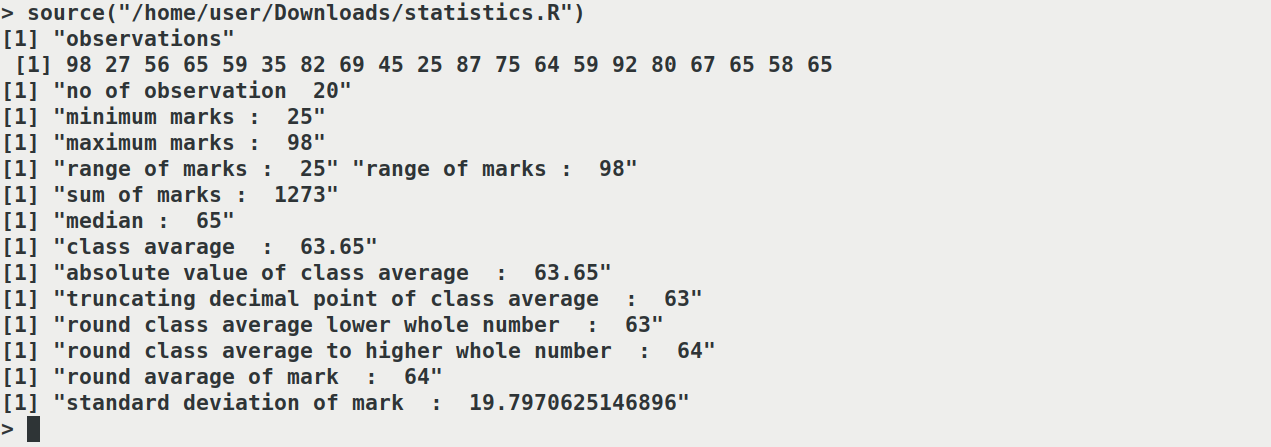
normal\_round\_marks <- round(class\_average)

print(paste("round avarage of mark  : ",normal\_round\_marks))

standard\_deviation <- sd(marks)

print(paste("standard deviation of mark  : ",standard\_deviation))

**Output:**



**3. Basic data structures: Vectors, data frames, lists and matrices**

**3.1 VECTORS**

**Program 1**:

fruits <- c("banana", "apple", "orange")

fruits

# Vector of numerical values

numbers <- c(1, 2, 3)

# Print numbers

Print(numbers)

# Vector with numerical values in a sequence

Numbers <- 1:10

# Print numbers

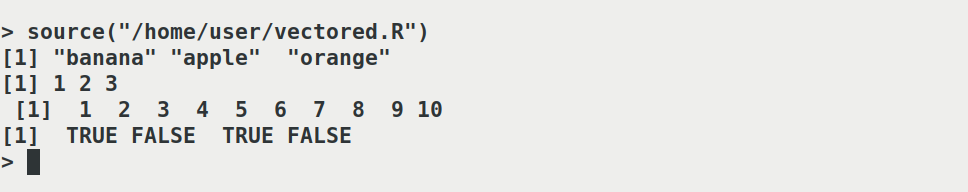
Print(Numbers)

# Vector of logical values

log\_values <- c(TRUE, FALSE, TRUE, FALSE)

print(log\_values)

**Output:**



**Vector Length**

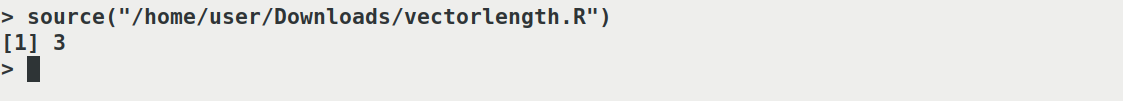
To find out how many items a vector has, use the **length()** function

**Program 2:**

fruits <- c("banana", "apple", "orange")

print(length(fruits))

**Output:**



**LISTS**

**Program 3:**

# List of strings

thislist <- list("apple", "banana", "cherry")

# Print the list

print(thislist)

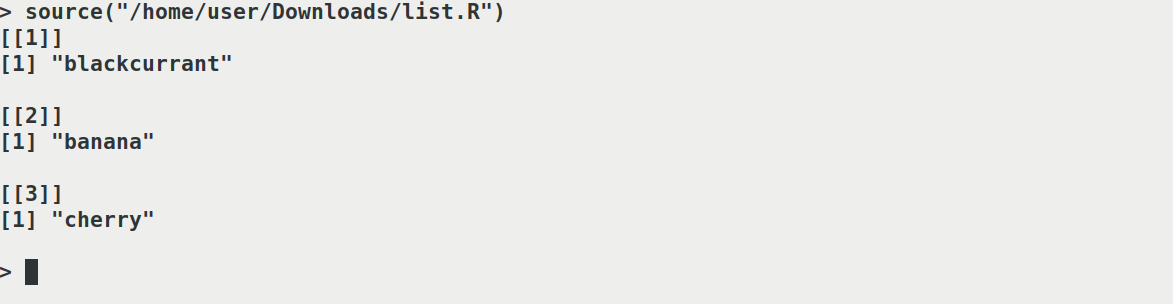
thislist[1]

thislist[1] <- "blackcurrant"

# Print the updated list

print(thislist)

**Output:**



**DATA FRAMES**

**Program 4:**

# Create a data frame

Data\_Frame <- data.frame (

Training = c("Strength", "Stamina", "Other"),

Pulse = c(100, 150, 120),

Duration = c(60, 30, 45)

)

# Print the data frame

print(Data\_Frame)

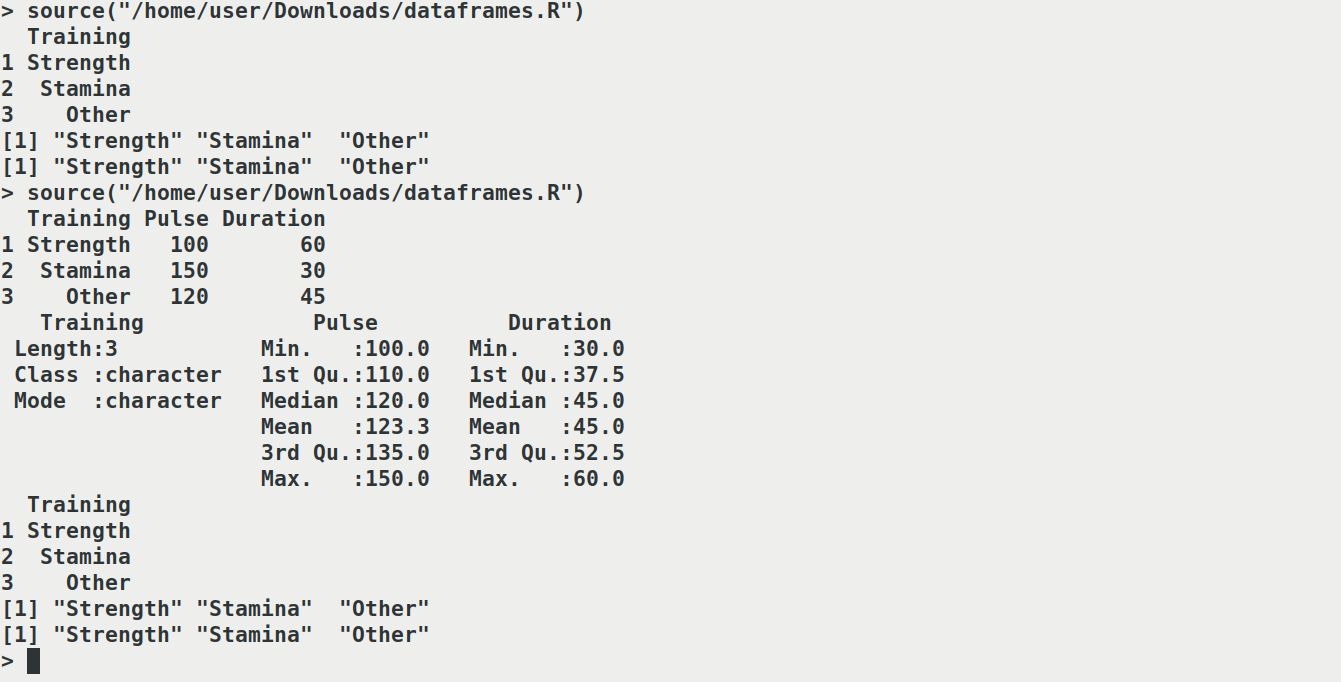
print(summary(Data\_Frame))

print(Data\_Frame[1])

print(Data\_Frame[["Training"]])

print(Data\_Frame$Training)

**Output:**

****

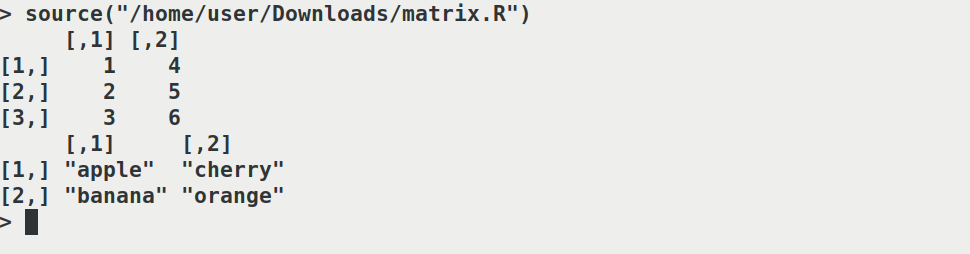
**MATRICES**

**Program 5:**

# Create a matrix  
thismatrix <- matrix(c(1,2,3,4,5,6), nrow = 3, ncol = 2)  
  
# Print the matrix  
print(thismatrix)

thismatrix <- matrix(c("apple", "banana", "cherry", "orange"), nrow = 2, ncol = 2)  
  
print(thismatrix)

**Output:**



**4. Logical statements and loops: IF-else statements, for and while loops, break**

**Problem:**

number1 <- readline(prompt = "enter number 1 :")

number2 <- readline(prompt = "enter number 2 :")

number3 <- readline(prompt = "enter number 3 :")

if(number1 > number2 && number1 > number3){

print("number1 is greater")

}else if(number2 > number3 && number2 > number1){

print("number2 is greater")

}else if(number3 > number1 && number3 > number2){

print("number3 is greater")

}else if(number3 == number1){

print("number3 is equal to number1")

}else if(number2 == number1){

print("number2 is equal to number1")

}else if(number2 == number3){

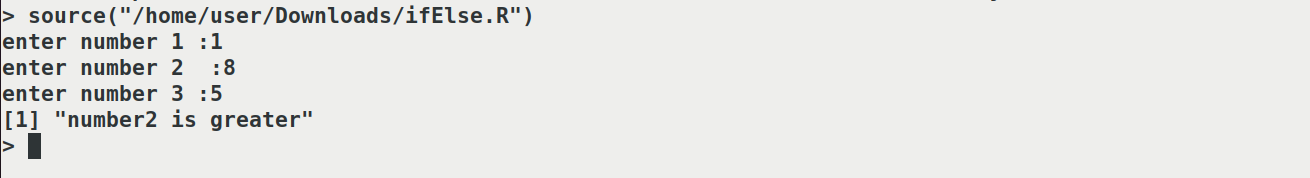
print("number2 is equal to number3")

}else{

print("please check number")

}

**Output**:



**Program:**

getNum <- as.numeric( readline(prompt = "enter a number : "))

fib <- 0

m <- 1

l <- 0

for (i in 0:getNum) {

print(fib)

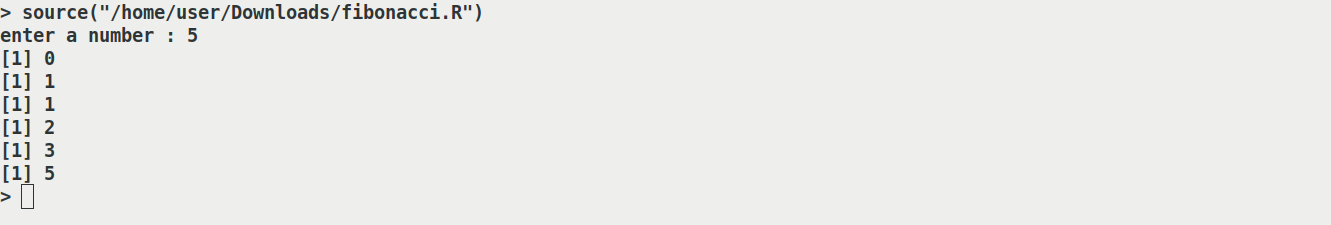
l = m + fib

fib = m

m = l

}

**Output:**



**Program:**

num = as.integer(readline(prompt="Enter a number: "))

sum = 0

temp = num

while(temp > 0) {

digit = temp %% 10

sum = sum + (digit ^ 3)

temp = floor(temp / 10)

}

if(num == sum) {

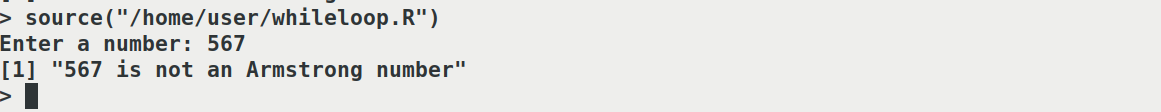
print(paste(num, "is an Armstrong number"))

} else {

print(paste(num, "is not an Armstrong number"))

}

**Output:**



**5.Writing user defined functions and packages**

factorial <- function(x) {

if(x==0)

return(1)

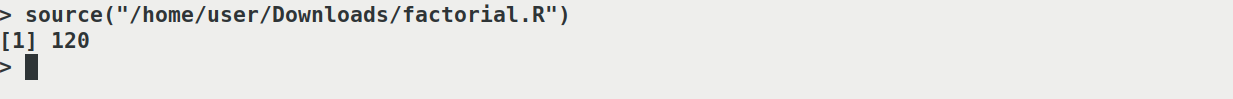
else

return(x\*factorial(x-1))

}

factorial(5)

**Output:**



**6. Reading and writing tables and files**

**TABLES**

**Program 1:**

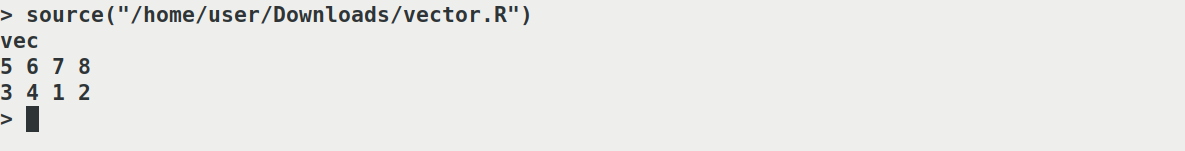
# Creating a vector

vec = c(6, 8, 7, 5, 6, 5, 6, 5, 8, 6)

# Calling table() Function

print(table(vec))

**Output :**



**Program 2:**

# Creating a data frame

df = data.frame(

"Name" = c ("v", "w", "x", "y", "z"),

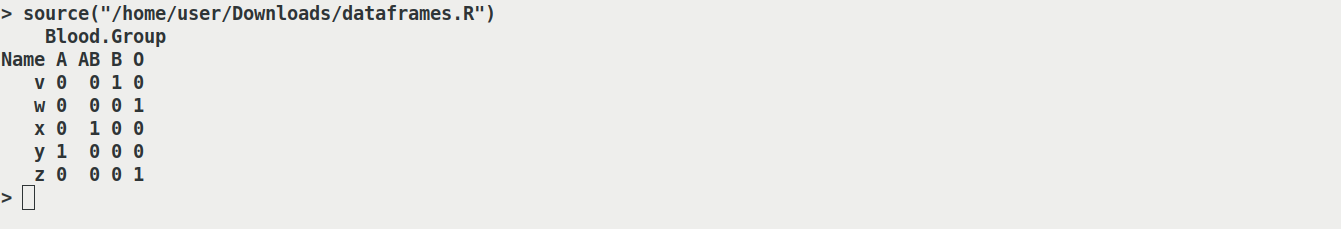
"Blood Group" = c ("B", "O", "AB", "A", "O")

)

# Calling table() function

print(table(df))

**Output :**

****

**FILE**

Creating a File

**Example :**

# Create a file

# The file created can be seen

# in your working directory

file.create("GFG.txt")

**Output :**

[1] TRUE

Writing into a File

**Example :**

# Write iris dataset

# into the txt file

write.table(x = iris[1:10, ], file = "GFG.txt")

**Output :**



Reading a File

**Program :**

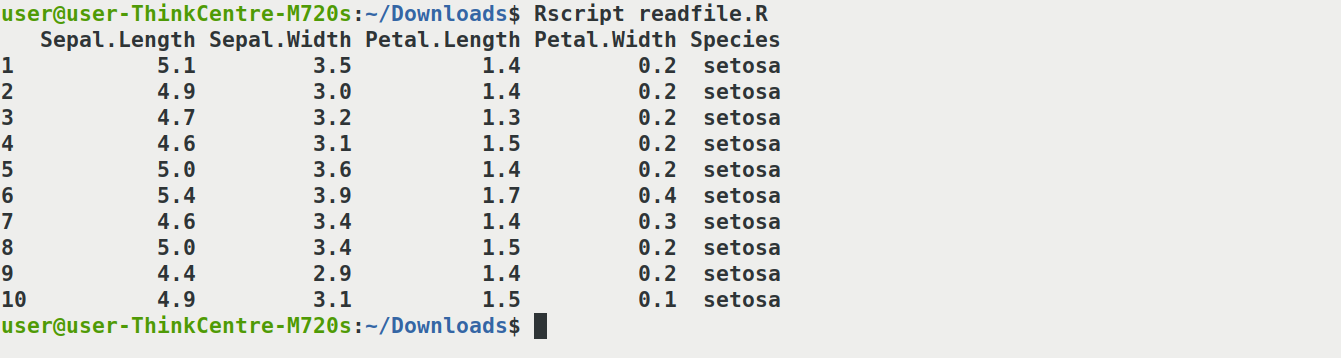
# Reading txt file

new.iris <- read.table(file = "GFG.txt")

# Print

print(new.iris)

**Output :**

****

1. **R graphics library: Line plots, histograms, pie charts, bar plots and other plot.**

Plot

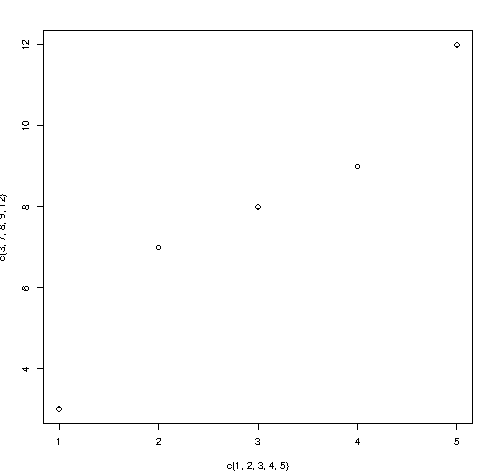
**Program:**

x <- c(1, 2, 3, 4, 5)

y<- c(3, 7, 8, 9, 12)

plot(x,y)

**Output:**



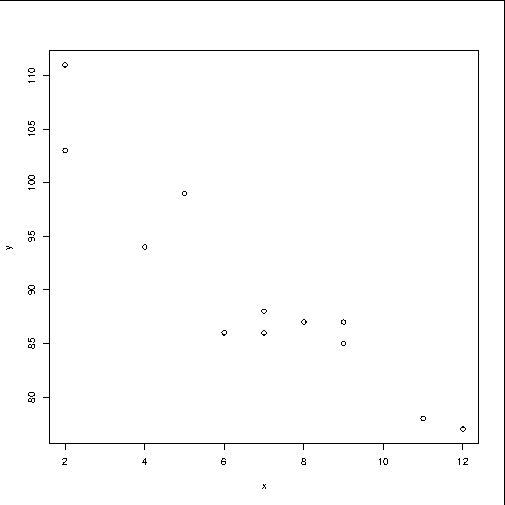
Scatterplot

x <- c(5,7,8,7,2,2,9,4,11,12,9,6)

y <- c(99,86,87,88,111,103,87,94,78,77,85,86)

plot(x,y)

**Output:**



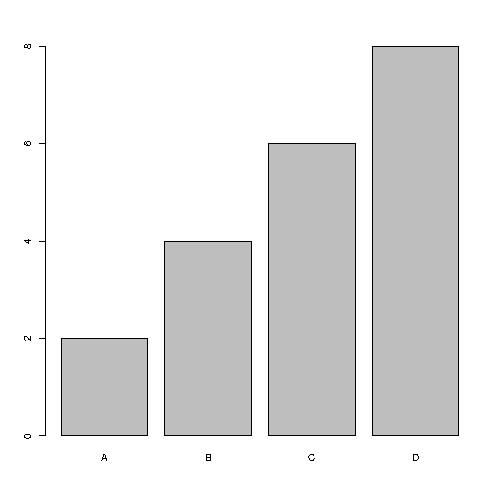
Barchart

x <- c("A", "B", "C", "D")

y <- c(2, 4, 6, 8)

barplot(y, names.arg = x)

output



Pie chart

**Program:**

x <- c(10,20,30,40)

pie(x)

**Output:**

