ASSIGNMENT-1

1. **What are the four fundamental parts of a computer?**

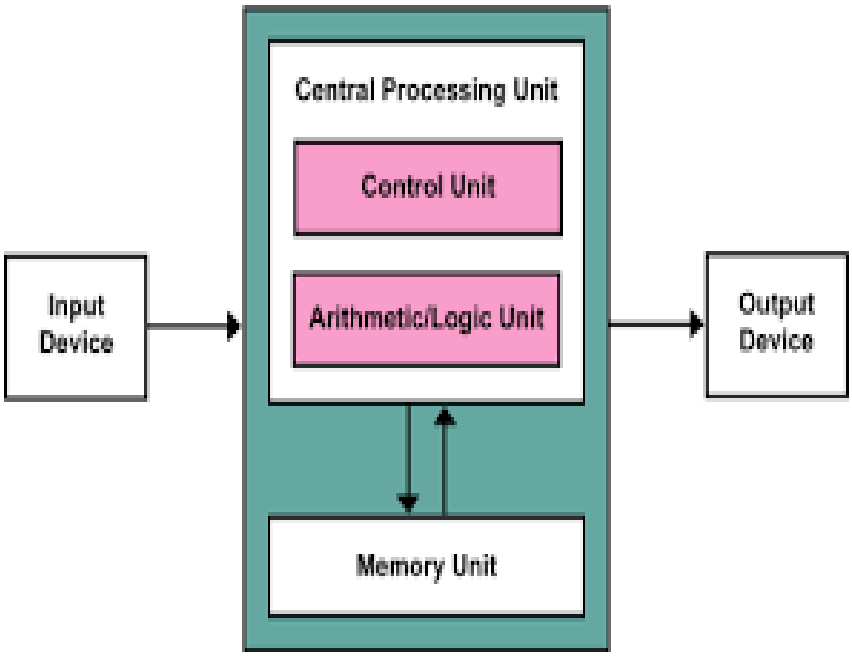
Ans: Four Fundamental Parts of Computer:

1. Central Processor Unit (**CPU**)
2. Memory (RAM)
3. Input (keyboard, mouse, etc)
4. Output (monitor, printer, etc)

Central Processor Unit (CPU):

Computer scientists typically call the CPU the "brain" of the computer because this is where programs are executed. A program is a set of instructions that tells the computer how to accomplish a specific task, such as sending a file to the printer, opening a browser window, or playing music or video.

The CPU is further broken up into three smaller components: the arithmetic unit handles all the simple mathematical computations; the control units interpret the instructions in a computer program; and the instruction decoding unit converts computer programming instructions into machine code. Machine code is the basic language understood by all the components in a computer.



Memory (RAM)

Once the CPU converts a specific set of computer program instructions into machine code, it stores that machine code in primary storage or memory. The machine code will be treated as either data or instructions. The CPU fetches data and instructions from memory, uses an instruction to manipulate the data, and then sends the result and the next set of instructions back to memory.

Memory is an ainternal storage area in the computer, which is used to store data and programs temporarily or permanently.
Memory is three types:

1. Primary Memory
2. Secondary Memory
3. Cache Memor

Input Devices:

Output units are the devices your computer uses to relay information to the user, such as a printer, monitors and speakers. For example, everything you see on your computer monitor starts as machine code in memory. The CPU takes that machine code and converts it into a format required by your monitor's hardware. Your monitor's hardware then converts that information into different light intensities so that you see words or pictures.

Examples for Input Devices:

1. Mouse
2. Key Board
3. Touch Pad
4. Bar Code Reader
5. OCR
6. OMR
7. MICR
8. Joy Stick
9. Microphone

Output :-

Output units are the devices your computer uses to relay information to the user, such as a printer, monitors and speakers. For example, everything you see on your computer monitor starts as machine code in memory. The CPU takes that machine code and converts it into a format required by your monitor's hardware. Your monitor's hardware then converts that information into different light intensities so that you see words or pictures.

Moniters:

The Moniter is a device used to display information, Images, Vedios and Graphic Information generated by the computer. The moniter is connected to either BGA or the DVU port.

Printers:

A **printer** is an output device that prints paper documents. This includes text documents, images, or a combination of both. The two most common types of **printers** are inkjet and laser **printers**. Inkjet **printers** are commonly used by consumers, while laser **printers** are a typical choice for businesses.

2. **Discuss about the classification of computers based on size and capacity.**

Ans: classification of computers based on size:

Computers can be classified into four categories based on their speed data that they can hold and price. These categories are as follows.

1. Super Computers
2. Mainframe Computers
3. Micro Computers

4. Mini Computers
    A) Desktop PC
    B) Work Stations
    C) Laptops
    D) Network Computers
    E) Hand Held Computers

classification of computers based on capacity:

Basically computers are classified into three categories

1. Analog Computers
2. Digital Computers
3. Hybrid Computers

3. What is the meaning of Computer generation? How many computer generations are defined? What technologies were/are used?

Ans. The history of computer development if often referred to in reference to the different generations of computing devices. Each generation of computer is characterized by a major technological development that fundamentally changed the way computers operate resulting in increasing and reliable devices.

There are five generations are defined, They are.

First Generation (1940-1959):

The first generation computers used vaccum tubes for circuitry and magnetic drums for memory and were often enormous, taking up entire rooms. They were very expensive to operate and in addition to using a great deal of electricity, generated a lot of heat.

1. J.P. Eckert and J.W. Mauchy invented the first successful electronic computer called ENAIC
2. It made use of vaccum tubes which are the only electronic component available during those days.
3. These computers could calculate in milliseconds
4. These were very big in size, weight was about 30 tonnes.

Second Generation (1959-1965):

1. Second generation computers were based on transistors instead of vaccum tubes.

    Eg: Honey Well 400, IBM 7094, CDC 1604, CDC 3600, UNIVAC 1108

2. Less energy and not produce as much heat as the first generation.
3. Assembly language and puch cards were used for input

4. Low cost than first generation computers
5. Better portability as compared to first generation.
6. A cooling system was required, Constant maintenance was required
7. Only used for specific purpose.

Third Generation (1965-1971):

1. These computers were based on Integrated Circuits.
2. IC was invented by Robert Noyce and Jack Kilby in 1958-1959.
3. IC was a single component containing number of transistors.

   Eg: PDP-8, PDP-11, ICL-2900, IBM-360, IBM-510.

4. These computers were cheaper as compared to second generation and were fast and reliable
5. Use of IC in the computer provides the small size of the computer.
6. This generation of computers has big storage capacity. Instead of punch cards, mouse and keyboard are used for input.
7. IC chips are difficult to maintain

Fourth Generation (1971-1980):

1. This technology is based on Microprocessor.
2. A microprocessor is used in a computer for any logical and arithmetic function to be performed in any program.
3. Graphics User Interface (GUI) technology was exploited to offer more comfort to users.
   Eg: IBM 4341, DEC 10, STAR 1000, PDP 11
4. Heat generated is negligible, less maintenance is required.
5. All types of high-level language can be used in this type of computers.
6. Air Conditioning is required in many cases due to the presence of ICs

Fifth Generation (1980-):

1. The period of Fifth Generation in 1980- onwards
2. This generation is based on Artificial intelligence
3. The aim of the fifth generation is to make a device which could respond to natural language input and are capable of learning and self-organization.
4. This generation is based on ULSI(Ultra Large Scale Integration) technology resulting in the production of microprocessor chips having ten million electronic component.
   Eg: Desktop, Laptop, Notebook, Ultra Book, Chrome Book.
5. It is more reliable and works faster.
6. It is available in different sizes and unique features.
7. It provides computers with more user-friendly interfaces with multimedia features.

4 Differentiate between Volatile and non-volatile memories.

Ans.

S.No	Volatile memory	Non-volatile memory
1.	Volatile memory is the type of memory in which data is lost as it is powered-off.	Non-volatile memory is the type of memory in which data remains stored even if it is powered-off.
2.	Contents of Volatile memory is stored temporarily.	Contents of Non-volatile memory is stored permanently.
3.	It is faster than non-volatile memory.	It is slower than volatile memory.
4.	**RAM(Random Access Memory)** is an example of volatile memory.	**ROM(Read Only Memory)** is an example of non-volatile memory
5.	In volatile memory, data can be easily transferred in comparison to non-volatile memory.	In non-volatile memory, data can not be easily transferred in comparison to volatile memory.
6.	In Volatile memory, process can read and write.	In Non-volatile memory, process can only read.
7.	Volatile memory generally has less storage capacity.	Non-volatile memory generally has more storage capacity than volatile memory.
8.	In volatile memory, the program's data are stored which are currently in process by the CPU.	In non-volatile memory, any kind of data which has to be saved permanently are stored.
9.	Volatile memory is more costly per unit size.	Non-volatile memory is less costly per unit size.
10.	Volatile memory has a huge impact on the system's performance.	Non-volatile memory has a huge impact on a system's storage capacity.
11.	In volatile memory, processor has direct access to data.	In non-volatile memory, processor has no direct access to data.
12.	Volatile memory chips are generally kept on the memory slot.	Non-volatile memory chips are embedded on the motherboard.

5 Distinguish among System software, Application Software and open source software on the basis of their features.

Ans. Application Software:

It is a set of programs, which are written to perform specific tasks. In general it is meant by computer programs. Application software can be broadly classified into 2 types.

1. Generalized Packages
2. Customized Packages

Generalized Packages:

1. Word Processing Software
2. Spread Sheets
3. Presentation

4. Data Base Management Sysytem
5. Graphic Tools

Customized Packages:

1. Student Information Details
2. Payroll Packages
3. Inventory Control

System Software:

System Software's are set of programs, responsible for running the computer, controlling various operations of computer systems and management of computer resources. Operating system(OS) falls under this category.

1. Operating System
2. Utilities

Open Source Software:

Open source is a term that originally referred to open source software (OSS). Open source software is code that is designed to be publicly accessible—anyone can see, modify, and distribute the code as they see fit.

Open source software is developed in a decentralized and collaborative way, relying on peer review and community production. Open source software is often cheaper, more flexible, and has more longevity than its proprietary peers because it is developed by communities rather than a single author or company.

Open source has become a movement and a way of working that reaches beyond software production. The open source movement uses the values and decentralized production model of open source software to find new ways to solve problems in their communities and industries.

6a. Create a file in MS-word to insert a paragraph about yourself and save it with file name "yourself". Describe all steps involved in it.

Myself

6b Write steps regarding following

- **To Change the Font Style:**

The basic steps to change the font of a text in a document are given below;

- Select the text you want to modify
- Select the Home tab and locate the Font group
- Click the drop-down arrow next to font style box
- Font style menu appears
- With a left click select the desired font style
- If you want to change the font to bold or italic, click the 'B' or 'I' icons on the format bar.

➢ To change the Font Size:

You can easily change the font size of your text in the document. The basic steps to change the Font size are listed below;

- Select the text that you want to modify
- In Home tab locate the Font group
- In Font group click the drop-down arrow next to font size box
- Font size menu appears
- Select the desired font size with a left click
- Select the text and click the increase or decrease font size buttons

➢ To change the Font Color:

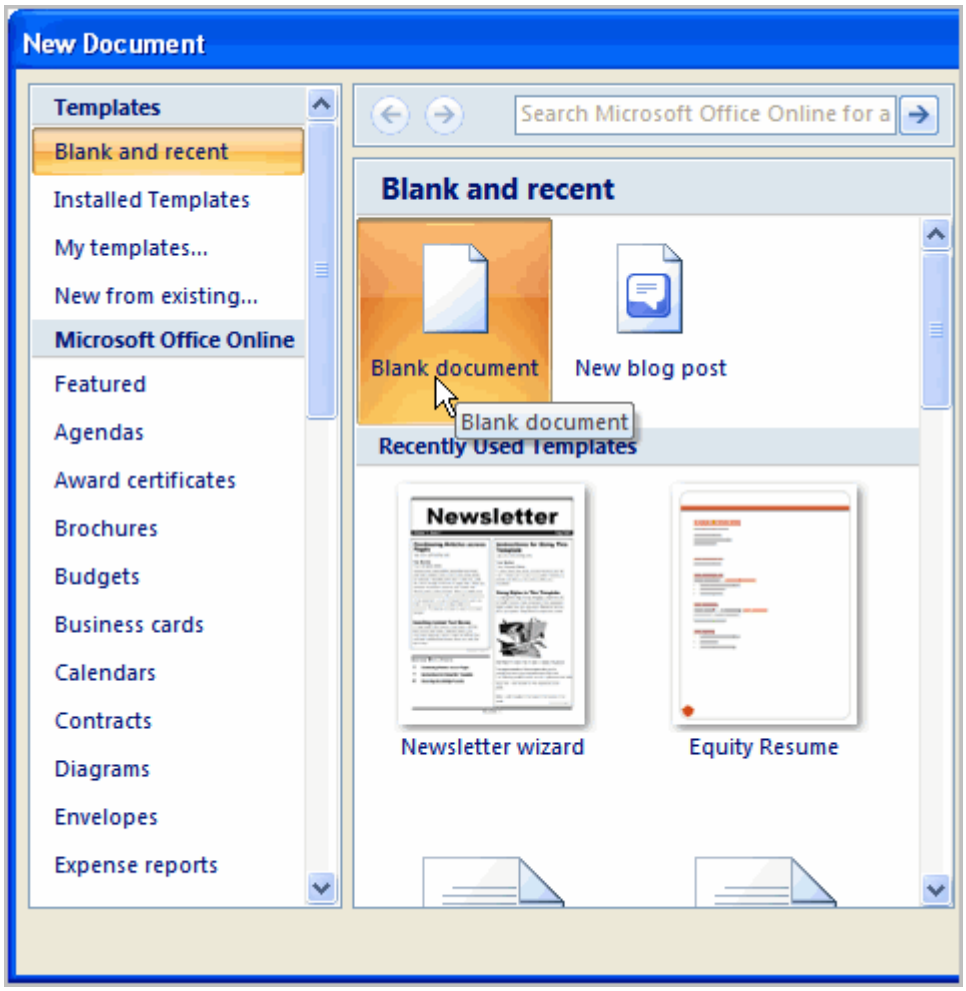
MS Word allows you to change the Font color of your text. If you want to emphasize a particular word or phrase, you can change its font color. The basic steps to change the Font color are given below;

- Select the text you want to modify
- In Home tab locate the Font group
- Click the drop-down arrow next to Font color button
- Font color menu appears
- Select the desired font color with a left click
- Word will change the Font color of the selected text.

7 Create a file in MS-Word for the following document and save it with file name ‘ms_word. Describe all steps involved in it.

Ans.
In addition to working with existing documents, you will want to be able to create new documents**. Each time you open Word, a new blank document appears; however, you will also need to know how to create new documents while an existing document is open.**
To create a new blank document:

- Click the **Microsoft Office button**.
- Select **New**. The New Document dialog box appears.
- Select **Blank document** under the **Blank and recent** section. It will be highlighted by default**.**



- Click **Create**. A new blank document appears in the Word window.

In this lesson, you will learn how to create new documents—including templates and **blank documents**—via the Microsoft Office button.

To create and design effective documents, you need to know how to **format text**. In addition to making your document more appealing, **formatted text** can draw the reader's attention to specific parts of the document and help communicate your message.

In this lesson, you will learn to format the font size, style, and color of text, as well as how to use the Bold, Italic, Underline, and Change Case commands.

Editing

Edit a document in Word

Click **Edit Document** > **Edit in Word** to make changes to a document.

When you open a document from One Drive, Word for the web displays it in Reading view. To make changes to your document, switch to Editing view, where you can add and delete content and do other things, such as:

- Add tables and pictures.
- Apply styles.
- Adjust formatting.
- Edit headers and footers.

You can do these simple tasks in both Reading view and Editing view:

- View and add comments.
- Share a document so you can work together with other people at the same time.
- Download a copy.
- Print.

Remember, to edit a document in Word, click **Edit Document** > **Edit in Word for the web**.

Save a Document:

When you save your files to the cloud, you can share and collaborate with others, and get to your files from anywhere - on your computer, tablet, or phone.

1. Select **File > Save As**.
2. Select **OneDrive**.

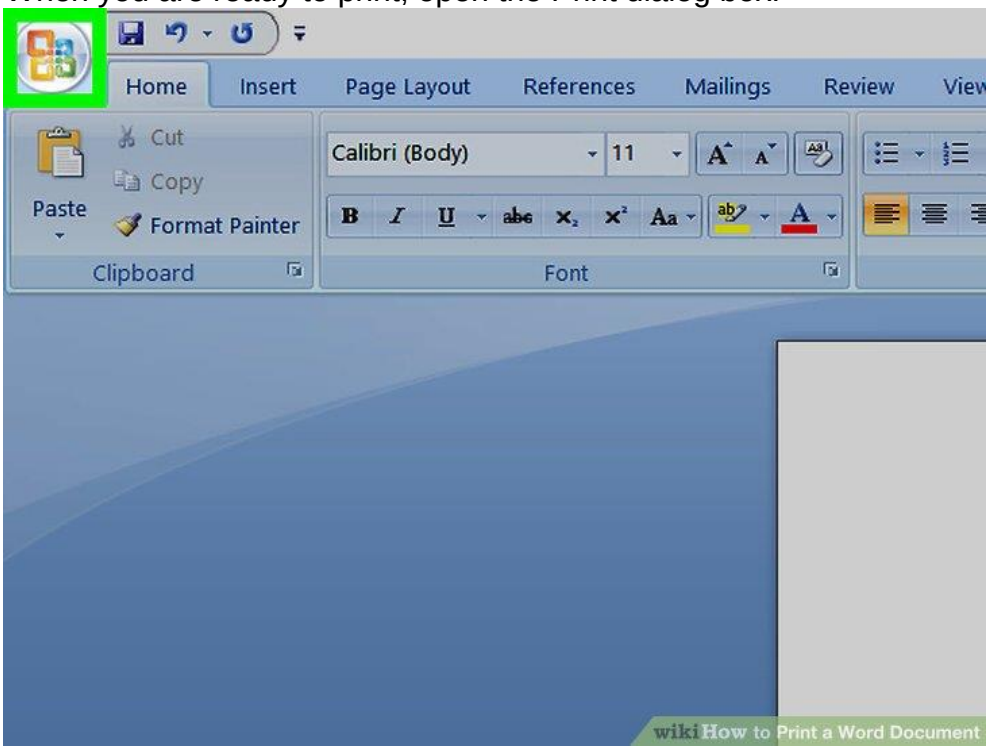
Save personal files to **OneDrive - Personal**, and work files to your company OneDrive. You can also save to another location in the list, or **Add a Place**.

3. Enter a descriptive name for the file, and select **Save**

Print a document:

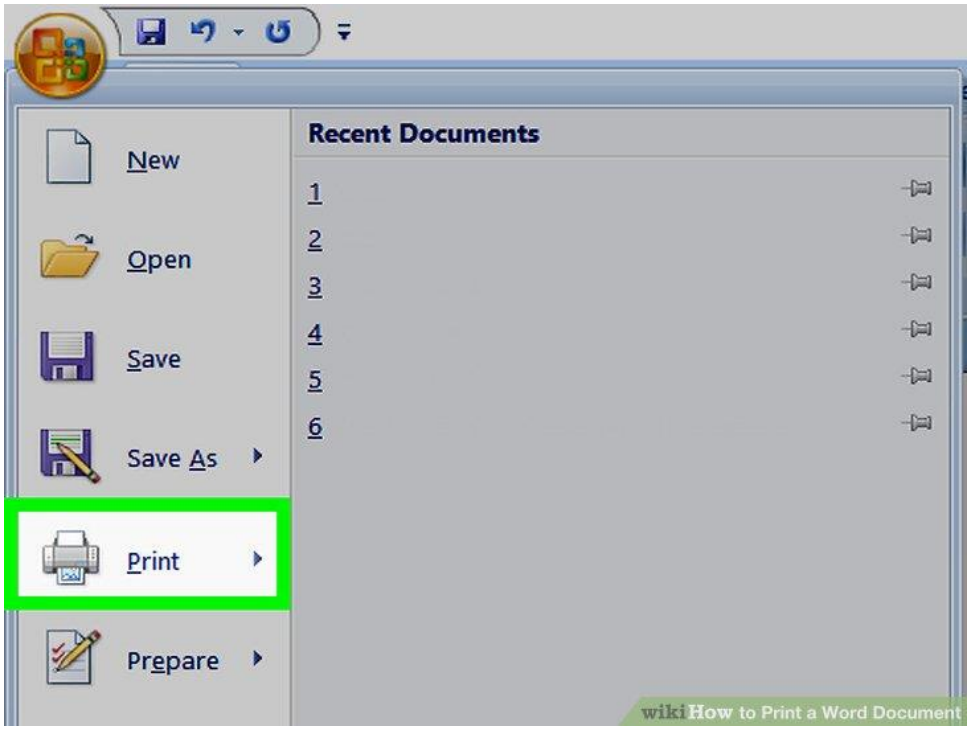
1. **Open or create a Microsoft Word document.** To do so, click on the blue app with a white document icon and bold "**W**," then click on **File** in the menu bar at the upper-left of the screen. Click on **Open...** to open an existing document or **New...** to create a new one.

- When you are ready to print, open the Print dialog box.



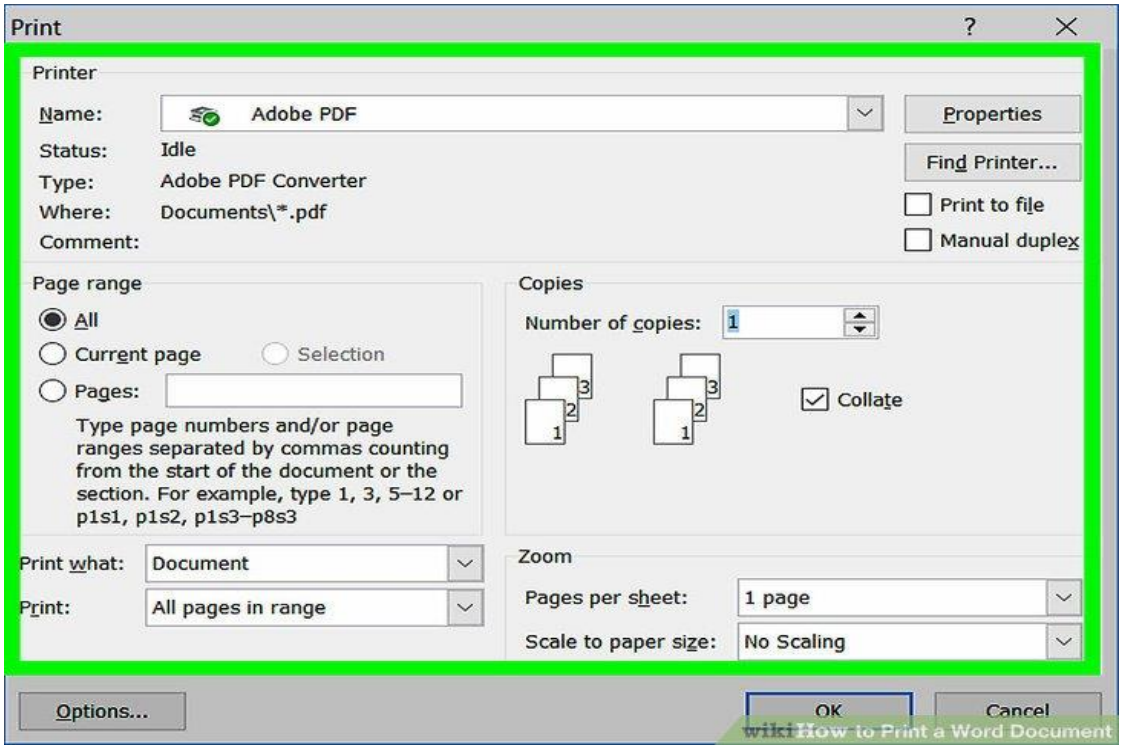
2

Click on File. It's in the menu bar at the upper-left of the screen or a tab at the upper-left of the window.



3

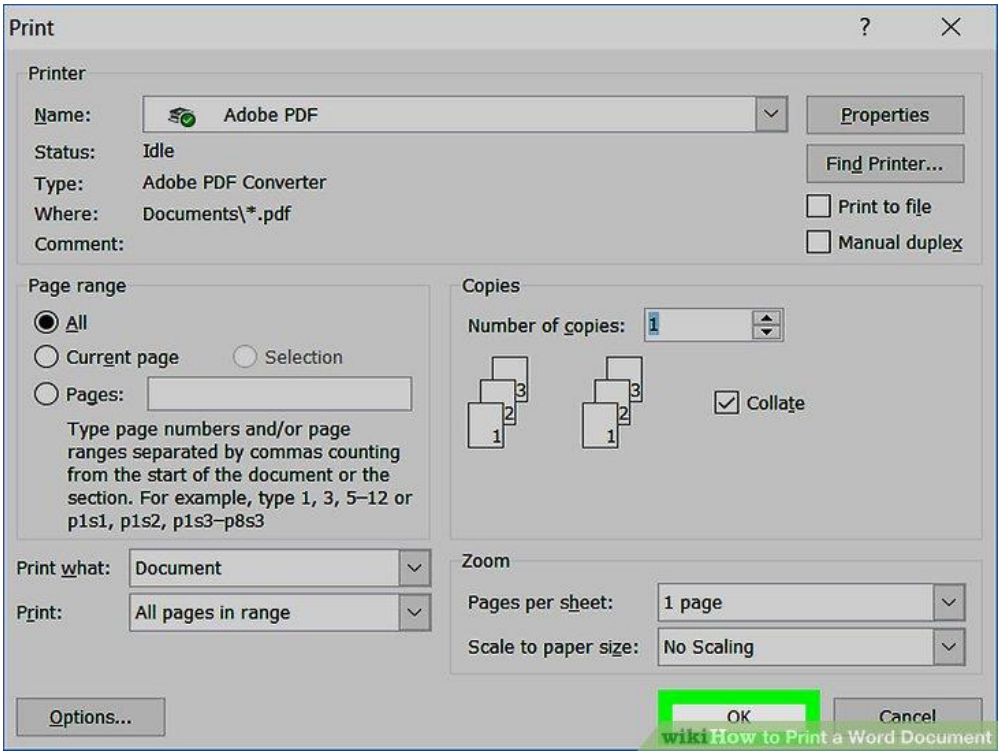
Click on Print.... The Print dialog box will open.



4

Select your printing options. Use the selections in the dialog box to select:

- Your default printer is displayed. Click on its name to select another printer from the drop-down menu.
- The number of copies to print. The default is 1; increase the quantity to print more copies.
- Which pages to print. The default is to print all the pages in the document, but you can choose to print the currently-displayed page, a highlighted selection, specific pages in the document, odd-numbered pages only, or even-numbered pages only.
- The size of paper to print on.
- The number of pages to print per sheet.
- Orientation of the paper. Select either Portrait (paper length vertical, width horizontal) or Landscape (paper width vertical, length horizontal).
- Margins. You can adjust the top, bottom, left, and right margins with the labeled up and down arrows or by typing numbers in the boxes.



5

Click on Print or OK. The button label will vary based on the version of Word you're using. Your document will print to the printer you selected.

14. What tools are available to customize our PowerPoint Presentation?

Presentation Templates

Create a **stunning presentation online** quickly and easily by getting started with one of Visme's premade presentation templates. Share content with your audience visually. Find a **free presentation template** that you can easily customize for your own business.

Slide Layouts

If you choose between one of our presentation themes, you have hundreds of slide layouts to choose from to build out your presentation in the way that you see fit.

Whether you need a title slide, a timeline, a slide for your services or an ending contact us slide, our slide library has plenty of options for you.

Fonts

Another important tool for your presentations that you have access to in Visme is free fonts. There are over 100 fonts for users to choose from in the presentation design dashboard.

Color Themes

While many of the presentation themes and templates come with an already established color scheme, absolutely nothing is set in stone. You can switch out the color of every item in your presentation to match your company or your topic.
One great presentation tool that Visme has is color themes. This allows you to change the color scheme of your entire presentation with a single click.

Icons

Using icons in your business presentation is a great way to visualize the text in your slides. You don't want to include only text for your audience to follow along, so incorporating graphics as a visual aid is a great way to make your content more engaging.
In Visme, users have access to over 10,000 icons with four different styles to choose from. Since you want to stick to a single icon style throughout your presentation, this gives you a way to differentiate each presentation you create while still having tons of icon options.

Shapes

Another great way to add a few design touches to your presentation is by incorporating shapes. These can be used to emphasize photos and text by surrounding them or to create a design by themselves.

Stock Photos

Visme partners with Unsplash to bring its users a massive library of free stock photos to incorporate into presentations, infographics, social media graphics and more.
While you may have your own photos of your team, business, products and services, there are still many other options for including stock photos in your presentation.

Charts and Graphs

There are many different chart and graph options available, from bar charts to line graphs and pie charts to funnel charts and more.

Maps

If you're talking about demographics or sharing information that pertains to a location within your presentation, consider adding in a map to visualize the area.
Visme's map maker allows you to insert a world map or pinpoint any location in the world, color coding areas and creating an engaging visualization for your audience.

Tables

There are many reasons you may need to add a table to your presentation.
You can create a schedule for the event or presentation, create a pricing table or even include a table of data next to the chart that visually represents it.

Flowcharts

A flowchart can help you visualize business processes, chains of command and more. And Visme makes it extremely easy to customize a premade flowchart or create one entirely from scratch.

Icon Charts

An icon chart or an array is a way to represent data with icons. You can use this to differentiate between the number of men and women, cats and dogs, etc., that fall under different categories.

Radials

A radial, or circular counter, is a more engaging way to represent numbers or percentages than having them stand alone on the slide.
There are many different styles to choose from in Visme's data widget sidebar.

Progress Bars

Want to let your audience know how much time there is left in the presentation? Add a progress bar to the bottom of each slide.
You can also use this to show how far along the company is with creating products or gathering funding.

Animation

To make your presentation more engaging, consider adding animation to the elements in your slides. This makes them move on the screen and helps grab the attention of your audience.

Transitions

Your transition is the way your slides appear and disappear, and it’s a great idea to give this a bit more flair than your typical slide interchange.
With Visme, you can create transitions that not only move your slide, but each of the elements simultaneously. Simply click on the gear at the right corner of the slides to access the transitions.

Interactivity

Visme also offers several great ways to make your presentation interactive.
Have Q&As, bring props, have your audience raise their hands and more. You create hyperlinks between slides and even design elements to create a unique experience for your audience.

Audio and Video

Adding an audio narrative, recording a voice-over for your slides, incorporating background music or even embedding a video can be a great way to increase engagement in your presentation.
You can record audio directly in Visme, embed video links or embed any other type of online content directly in your presentation slides

16. What is the difference between Machine Language and High level Language?

Ans **Computer Languages**

Languages are a means of communication. Normally people interact with each other through language. On the same pattern, communication with computers is carried out through a language. This language is understood both by user and the machine. Every computer language is bound by rules known as syntax of the language.

Computer languages are broadly classified as:

1. Low level language
2. High level language

Low level language:

The term low level means closeness to the way in which machine understood. The low level languages are.

Machine Language:

This is the language (in the form of 0's and 1's called binary numbers) understood directly by the computer. It is machine dependent. It is difficult to learn and even more difficult to write programs.

High level language:

To overcome the limitations of low level language, high level has been evolved which uses normal English. High level languages are computerindependent and programming becomes quite easy and simple.

Various high level languages are given below.

1. BASIC (Beginners All Purpose Symbolic Instruction Code)
2. COBOL (Common Business Oriented Language)
3. FORTRAN (Formula Translation)
4. C (Structured Programming Language)
5. C++

17. Discuss about different data types of C programming Language>

Ans: **Data Types in C**

Each variable in C has an associated data type. Each data type requires different amounts of memory and has some specific operations which can be performed over it. Let us briefly describe them one by one:
Following are the examples of some very common data types used in C:

- **char:** The most basic data type in C. It stores a single character and requires a single byte of memory in almost all compilers.
- **int:** As the name suggests, an int variable is used to store an integer.
- **float:** It is used to store decimal numbers (numbers with floating point value) with single precision.
- **double:** It is used to store decimal numbers (numbers with floating point value) with double precision.

Different data types also have different ranges upto which they can store numbers. These ranges may vary from compiler to compiler. Below is list of ranges along with the memory requirement and format specifiers on 32 bit gcc compiler.

Data Type	Memory (bytes)	Range	Format Specifier
short int	2	-32,768 to 32,767	%hd
unsigned short int	2	0 to 65,535	%hu
unsigned int	4	0 to 4,294,967,295	%u
int	4	-2,147,483,648 to 2,147,483,647	%d
long int	4	-2,147,483,648 to 2,147,483,647	%ld
unsigned long int	4	0 to 4,294,967,295	%lu
long long int	8	-(2^63) to (2^63)-1	%lld
unsigned long long int	8	0 to 18,446,744,073,709,551,615	%llu

Data Type	Memory (bytes)	Range	Format Specifier
signed char	1	-128 to 127	%c
unsigned char	1	0 to 255	%c
float	4		%f
double	8		%lf
long double	16		%Lf

We can use the sizeof() operator to check the size of a variable. See the following C program for the usage of the various data types:

- C

```
#include <stdio.h>

int main()

{

    int a = 1;

    char b = 'G';

    double c = 3.14;

    printf("Hello World!\n");
```

```
    // printing the variables defined
    // above along with their sizes
    printf("Hello! I am a character. My value is %c and "
           "my size is %lu byte.\n",
           b, sizeof(char));
    // can use sizeof(b) above as well

    printf("Hello! I am an integer. My value is %d and "
           "my size is %lu  bytes.\n",
           a, sizeof(int));
    // can use sizeof(a) above as well

    printf("Hello! I am a double floating point variable."
           " My value is %lf and my size is %lu bytes.\n",
           c, sizeof(double));
    // can use sizeof(c) above as well

    printf("Bye! See you soon. :)\n");

    return 0;
}
```

Output:

Hello World!

Hello! I am a character. My value is G and my size is 1 byte.

Hello! I am an integer. My value is 1 and my size is 4 bytes.

Hello! I am a double floating point variable. My value is 3.140000 and my size i

s 8 bytes.

Bye! See you soon. :)

19. Describe the syntax of the following statements

a) If- else statement b) for loop c) while loop d) do- while loop

If- else statement:

An **if** statement can be followed by an optional **else** statement, which executes when the Boolean expression is false.

Syntax

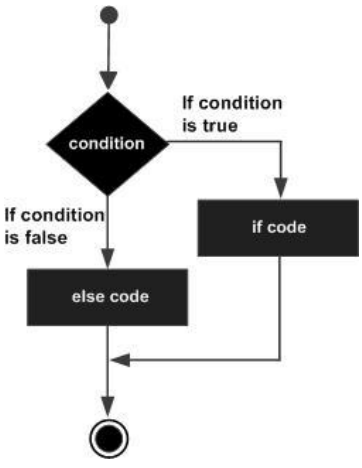
The syntax of an **if...else** statement in C programming language is −

```
if(boolean_expression) {
   /* statement(s) will execute if the boolean expression is true */
} else {
   /* statement(s) will execute if the boolean expression is false */
}
```

If the Boolean expression evaluates to **true**, then the **if block** will be executed, otherwise, the **else block** will be executed.

C programming language assumes any **non-zero** and **non-null** values as **true**, and if it is either **zero** or **null**, then it is assumed as **false** value.

Flow Diagram



Example

```
#include <stdio.h>

int main () {

   /* local variable definition */
   int a = 100;

   /* check the boolean condition */
   if( a < 20 ) {
      /* if condition is true then print the following */
      printf("a is less than 20\n" );
   } else {
      /* if condition is false then print the following */
      printf("a is not less than 20\n" );
   }

   printf("value of a is : %d\n", a);

   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
a is not less than 20;
value of a is : 100
```

If...else if...else Statement

An **if** statement can be followed by an optional **else if...else** statement, which is very useful to test various conditions using single if...else if statement.

When using if...else if..else statements, there are few points to keep in mind −

- An if can have zero or one else's and it must come after any else if's.
- An if can have zero to many else if's and they must come before the else.
- Once an else if succeeds, none of the remaining else if's or else's will be tested.

Syntax

The syntax of an **if...else if...else** statement in C programming language is −

```
if(boolean_expression 1) {
   /* Executes when the boolean expression 1 is true */
} else if( boolean_expression 2) {
   /* Executes when the boolean expression 2 is true */
} else if( boolean_expression 3) {
   /* Executes when the boolean expression 3 is true */
} else {
   /* executes when the none of the above condition is true */
}
```

Example

```
#include <stdio.h>

int main () {

   /* local variable definition */
   int a = 100;

   /* check the boolean condition */
   if( a == 10 ) {
      /* if condition is true then print the following */
      printf("Value of a is 10\n" );
   } else if( a == 20 ) {
      /* if else if condition is true */
      printf("Value of a is 20\n" );
   } else if( a == 30 ) {
      /* if else if condition is true  */
      printf("Value of a is 30\n" );
   } else {
      /* if none of the conditions is true */
      printf("None of the values is matching\n" );
   }

   printf("Exact value of a is: %d\n", a );

   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
None of the values is matching
Exact value of a is: 100
```

For Loop

A **for** loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

Syntax

The syntax of a **for** loop in C programming language is −

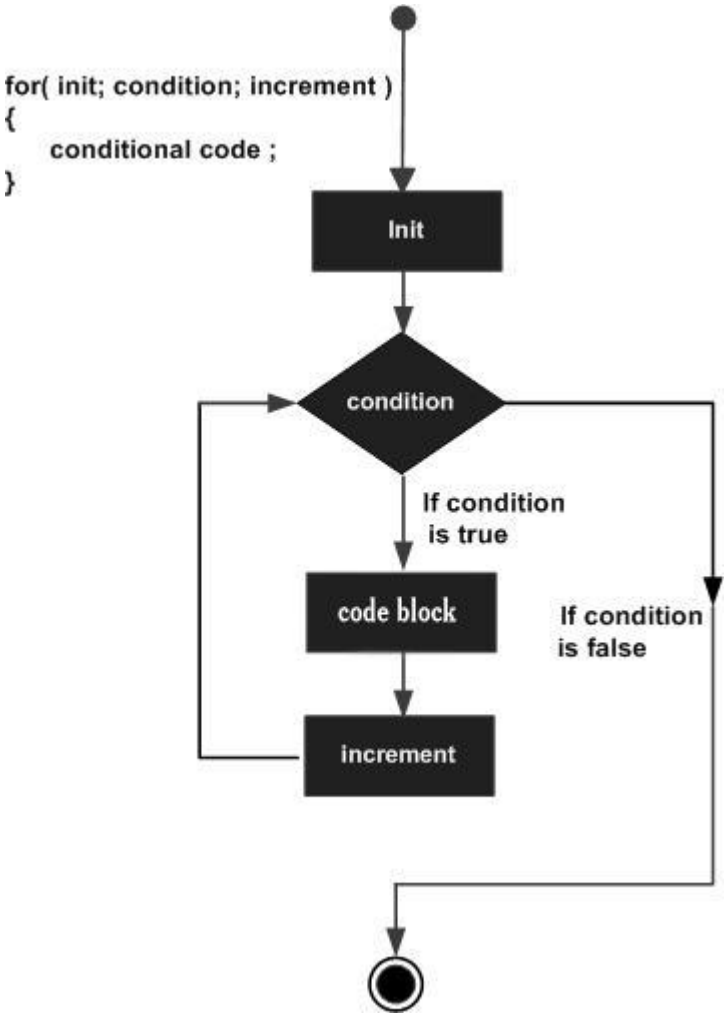
```
for ( init; condition; increment ) {
   statement(s);
}
```

Here is the flow of control in a 'for' loop −

- The **init** step is executed first, and only once. This step allows you to declare and initialize any loop control variables. You are not required to put a statement here, as long as a semicolon appears.
- Next, the **condition** is evaluated. If it is true, the body of the loop is executed. If it is false, the body of the loop does not execute and the flow of control jumps to the next statement just after the 'for' loop.

- After the body of the 'for' loop executes, the flow of control jumps back up to the **increment** statement. This statement allows you to update any loop control variables. This statement can be left blank, as long as a semicolon appears after the condition.
- The condition is now evaluated again. If it is true, the loop executes and the process repeats itself (body of loop, then increment step, and then again condition). After the condition becomes false, the 'for' loop terminates.

Flow Diagram



Example

```
#include <stdio.h>

int main () {

   int a;

   /* for loop execution */
   for( a = 10; a < 20; a = a + 1 ){
      printf("value of a: %d\n", a);
   }

   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```

While Loop :

A **while** loop in C programming repeatedly executes a target statement as long as a given condition is true.

Syntax

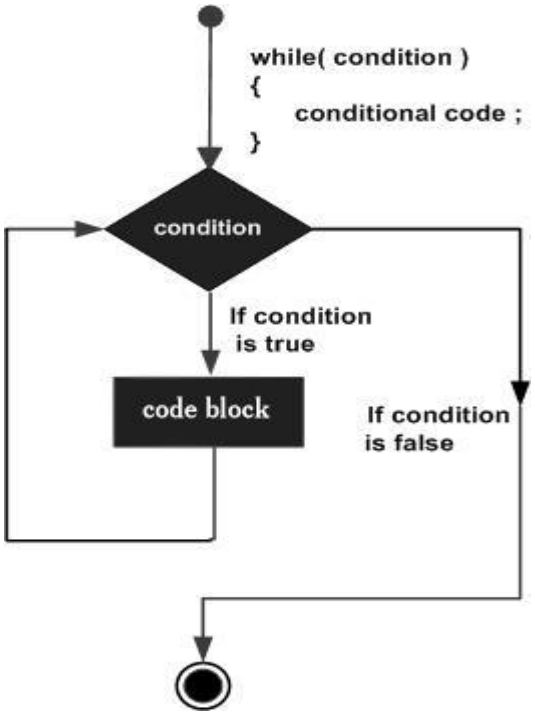
The syntax of a **while** loop in C programming language is −

```
while(condition) {
   statement(s);
}
```

Here, **statement(s)** may be a single statement or a block of statements. The **condition** may be any expression, and true is any nonzero value. The loop iterates while the condition is true.

When the condition becomes false, the program control passes to the line immediately following the loop.

Flow Diagram



Here, the key point to note is that a while loop might not execute at all. When the condition is tested and the result is false, the loop body will be skipped and the first statement after the while loop will be executed.

Example

```
#include <stdio.h>

int main () {

   /* local variable definition */
   int a = 10;

   /* while loop execution */
   while( a < 20 ) {
      printf("value of a: %d\n", a);
      a++;
   }

   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```

Do- While loop:

Unlike **for** and **while** loops, which test the loop condition at the top of the loop, the **do...while** loop in C programming checks its condition at the bottom of the loop.

A **do...while** loop is similar to a while loop, except the fact that it is guaranteed to execute at least one time.

Syntax

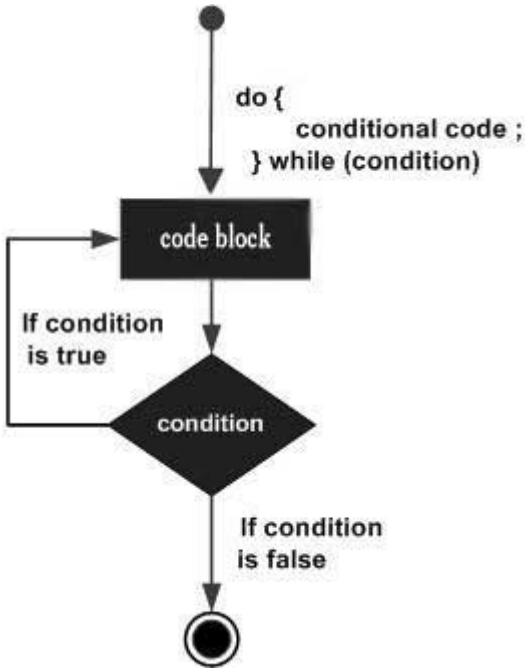
The syntax of a **do...while** loop in C programming language is −

```
do {
   statement(s);
} while( condition );
```

Notice that the conditional expression appears at the end of the loop, so the statement(s) in the loop executes once before the condition is tested.

If the condition is true, the flow of control jumps back up to do, and the statement(s) in the loop executes again. This process repeats until the given condition becomes false.

Flow Diagram



Example

```
#include <stdio.h>

int main () {

   /* local variable definition */
   int a = 10;

   /* do loop execution */
   do {
      printf("value of a: %d\n", a);
      a = a + 1;
   }while( a < 20 );

   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```