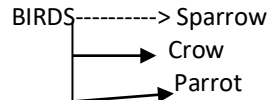


**OBJECT ORIENTED PROGRAMMING(OOP):-** Object based programming is called OOP. It is pronounced as oops. This programming language is different from procedural programming languages. Example:- oops is grouped as “objects”, like inheritance, polymorphism, data abstraction etc. oops language follow following properties.

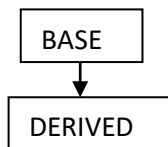
1. **OBJECT:-** Object is a thing which is used to perform certain operations. It has certain properties and behaviors. Eg:- car is an object its properties are color, size and its behavior is driving.
2. **CLASS:-** A group or category of things means objects having some properties and behavior. It is used to specify the form of an object. As for eg:- BIRDS



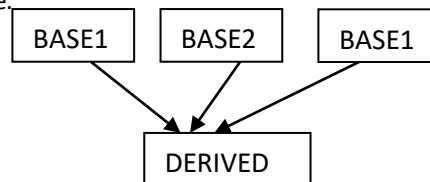
Here, birds is the class which includes parrot, crow, sparrow etc. having different properties and behavior.

3. **ENCAPSULATION:-** It is the mechanism that bind code and the data together we performs encapsulation by putting the data and object in private or public mode. Name can able to access the data and code if we declare data us private.
4. **INHERITANCE:-** It is the properties of oops by which one object can acquire the properties of another object. It is of many types:-

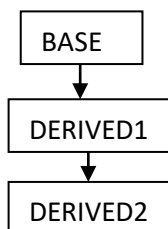
a. **SINGLE INHERITANCE:-** When an object acquire the properties of only one object then it is called single inheritance.



b. **MULTIPLE INHERITANCE:-** When an object acquire the properties of more than one object then it is called multiple inheritance.



c. **MULTILEVEL INHERITANCE:-**When an object acquire the properties of another derived class.



5. **POLYMORPHISM:-** The term “Polymorphism” is made by two words “poly” which means MANY and the second is “morphism” means FROMS. It means an object can take more than one form depending upon situation. This term is taken form greek word.
6. **DATA ABSTRACTION :-** Abstraction is the process by which data and programs are defined with a representation similar in form to its meaning. It is the act of representing essential features without giving the backgraound details. Oop language follow the rule of data abstraction. Data abstraction is defined by their meaning while hiding away the details of background operations . abstraction captures only those details about an object that are relevant to the current perspective. Abstraction can apply to control and also to data.

OR

- Data abstraction is the reduction of a particular body of data to a simplified representation of the whole abstraction. OR
- It is a process of representing the essential features without including implementation details.

❖ **VARIABLE:-** Variable is the name of memory location where we store value the value of variable is not fixed. In VB variable is declared as follow.

Dim A as Integer

**Variable\_ name data type**

Q. if we want to display sum of two number on form. text  
Enter first no.

```
Private sub cmdsum_click()
    Dim a, b as integer
    a=5, b=5
    Print a + b
End sub
```

### **Common dialog Control1**

It provides build in windows dialog boxes. These dialog boxes are built in to the operating system and any application can use them. The built in windows dialog boxes are open, save as, color, font, print, some important properties of dialog control are cancel error dialog tille etc. To add common dialog box control to our project we select project menu then select Microsoft common dialog control 6.0. click on apply and OK button for adding common dialog control to tool box.

**project → components → Microsoft Common dialog control 6.0 →Apply →OK**

❖ **FLEXGRID CONTROL:-** The Microsoft flexgrid control (ms flexgrid) is useful tool for displaying information in tabular form. Ms flexgrid control display read only data. Some important properties of flexgrid control are :- column, rows, datasource and datafields.

❖ **SLIDE CONTROL:-** A slide control is a window containing a slider and optional tickmark. We can move the slider by dragging it or clicking the mouse to either side of the slide. Some important properties of slider control are :- link, max, values, small change, large change, tick style etc. we generally use change and scroll event of slider control.

Example :- Add a label on the form control. Write some text on label control 's caption property. Then write some code in the code window as given below

```
Lblmessage.font size= str &(slider, value)
```

After writing the code run the project we see that we can increase or decrease the font size by drag this scroll bar.

❖ **Some important properties of visual basic controls are:-** some important properties of visual basic control are given bellow:-

1. NAME :- Name is the property of control by which it can be easily identified. By the name property user can access the other property of control. Default name can be changed during design time or runtime.
2. Caption :- Caption property is used to change the text of any component. we can rename the caption. It is used to show the label of the control.
3. Appearance:- Appearance property is used to control the appearance of control.
4. Back color:- A color which is represented on the background is called back color.
5. Fore color:- Text color of component is known as fore color. It is used to display text and graphics in an object.
6. Font:- font is the type of text which is appearing on the component.
7. Text :- number of character can be entered in to textbox.
8. Enable:- Enable is used to check whether a component is active or not.
9. Visible:- Visible means that property by which component can be seen.

❖ **DECLARATION OF VARIABLE:-** A variable is declared with **Dim** statement within a procedure exist only as long as the procedure finishes the value of variable disappears. We can't access a variable of one procedure from another procedure. These characteristics allow us to use the some variable in the different procedures. There are other ways to declare variable. These are given below:-

- i. Declaring variable in the declaration section of the form, standard or class module rather than within the procedure makes the variables available to all the procedure in the module.
- ii. Declaring a variable using **public** keyword makes it available throughout our application.
- iii Declaring the local variable using the **static** keyword preserves its value even a procedure ends.

```
Private sub command1_click()  
Static x as integer  
X= x+1  
Print x  
End sub
```

If we write the keyword static in our program then it print the no. 1,2,3, ..... N because it stores its previous value. But if we write dim then it always print only 1. because it doesn't stores its previous value.