**Procedural Language and Object Oriented Programming Language**

| **BASIS FOR COMPARISON** | **POP** | **OOP** |
| --- | --- | --- |
| Basic | Procedure/Structure oriented . | Object oriented. |
| Approach | Top-down. | Bottom-up. |
| Basis | Main focus is on "how to get the task done" i.e. on the procedure or structure of a program . | Main focus is on 'data security'. Hence, only objects are permitted to access the entities of a class. |
| Division | Large program is divided into units called functions. | Entire program is divided into objects. |
| Entity accessing mode | No access specifier observed. | Access specifier are "public", "private", "protected". |
| Overloading/Polymorphism | Neither it overload functions nor operators. | It overloads functions, constructors, and operators. |
| Inheritance | Their is no provision of inheritance. | Inheritance achieved in three modes public private and protected. |
| Data hiding & security | There is no proper way of hiding the data, so data is insecure | Data is hidden in three modes public, private, and protected. hence data security increases. |
| Data sharing | Global data is shared among the functions in the program. | Data is shared among the objects through the member functions. |
| Friend functions/classes | No concept of friend function. | Classes or function can become a friend of another class with the keyword "friend". Note: "friend" keyword is used only in c++ |
| Virtual classes/ function | No concept of virtual classes . | Concept of virtual function appear during inheritance. |
| Example | C, VB, FORTRAN, Pascal | C++, JAVA, VB.NET, C#.NET. |