**Practical No. 05**

**Aim:** Modeling Data flow diagram and control flow diagram.

**Data Flow Diagram**

DFD provides the functional overview of a system. The graphical representation easily overcomes any gap between ’user and system analyst’ and ‘analyst and system designer’ in understanding a system. Starting from an overview of the system it explores detailed design of a system through a hierarchy. DFD shows the external entities from which data flows into the process and also the other flows of data

**Graphical notations for Data Flow Diagram**

| **Term** | **Notation** | **Remarks** |
| --- | --- | --- |
| External entity |  | Name of the external entity is written inside the rectangle |
| Process | z | Name of the process is written inside the circle |
| Data store |  | A left-right open rectangle is denoted as data store; name  of the data store is written inside the shape |
| Data flow |  | Data flow is represented by a directed arc with its data name |

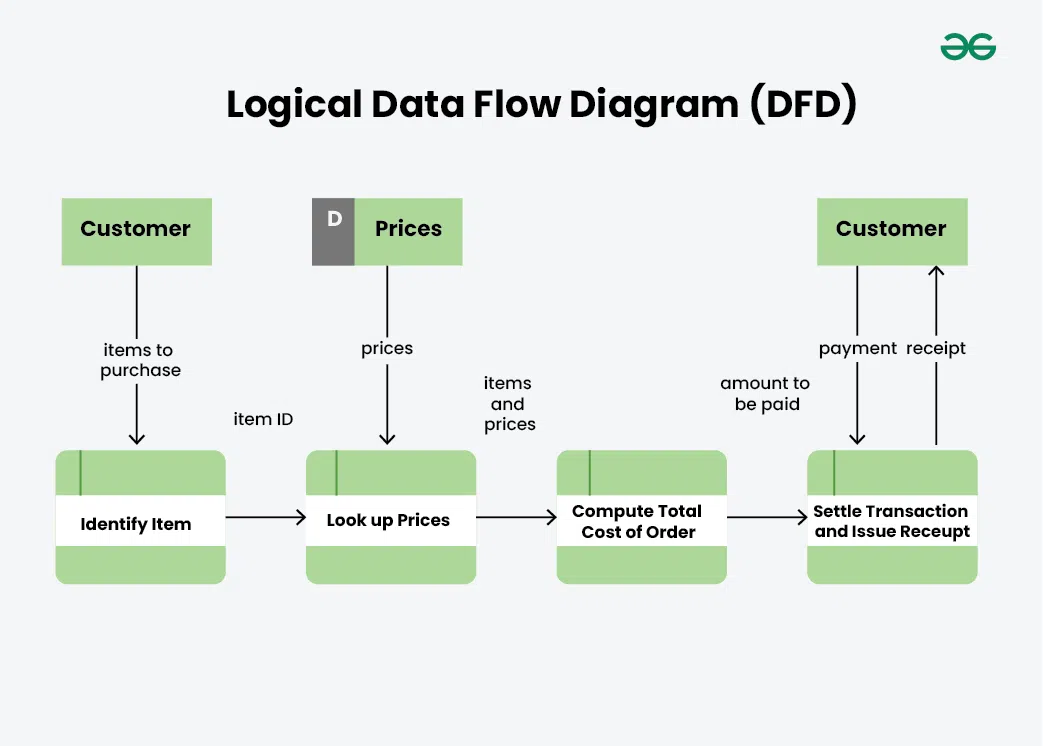
**Types of Data Flow Diagram (DFD)**

**There are two types of Data Flow Diagram (DFD)**

1. Logical Data Flow Diagram
2. Physical Data Flow Diagram

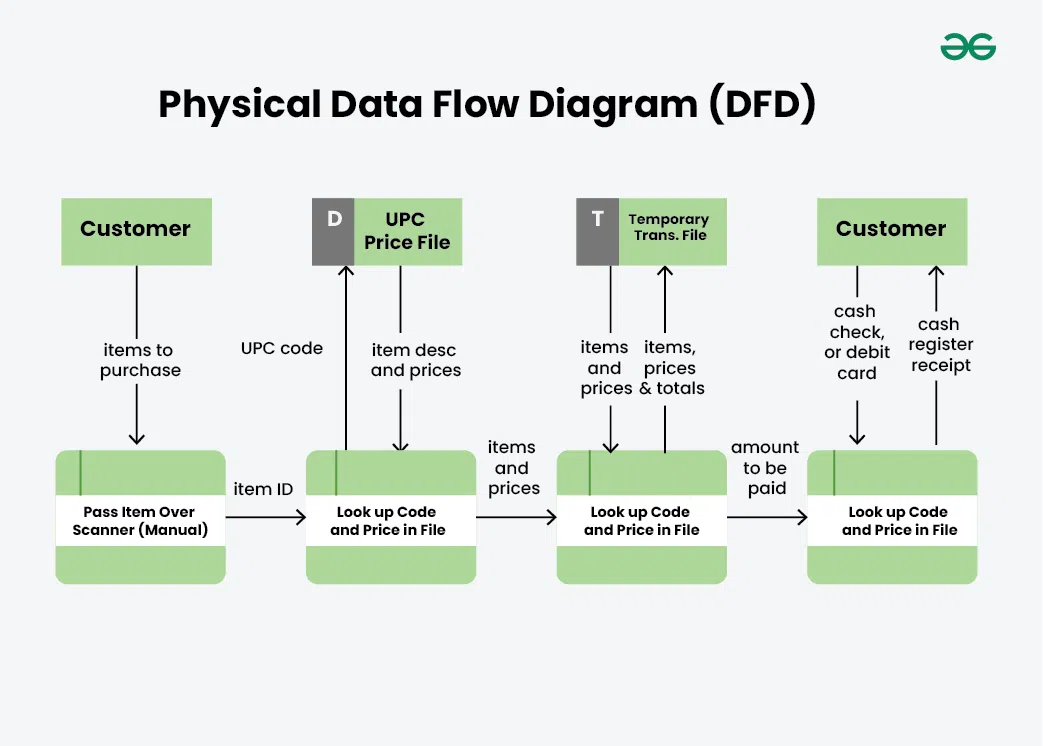
**Logical Data Flow Diagram (DFD)**

Logical data flow diagram mainly focuses on the system process. It illustrates how data flows in the system. Logical Data Flow Diagram (DFD) mainly focuses on high level processes and data flow without diving deep into technical implementation details. Logical DFD is used in various organizations for the smooth running of system. Like in a Banking software system, it is used to describe how data is moved from one entity to another.



**Physical Data Flow Diagram**

Physical data flow diagram shows how the data flow is actually implemented in the system. In the Physical Data Flow Diagram (DFD), we include additional details such as data storage, data transmission, and specific technology or system components. Physical DFD is more specific and close to implementation.



**Control Flow Diagram**

**A Control Flow Graph (CFG) is the graphical representation of control flow or computation during the execution of programs or applications. Control flow graphs are mostly used in static analysis as well as compiler applications, as they can accurately represent the flow inside a program unit. The control flow graph was originally developed by *Frances E. Allen.***

**Characteristics of Control Flow Graph**

* **The control flow graph is process-oriented.**
* **The control flow graph shows all the paths that can be traversed during a program execution.**
* **A control flow graph is a directed graph.**
* **Edges in CFG portray control flow paths and the nodes in CFG portray basic blocks.**

**There exist 2 designated blocks in the Control Flow Graph:**

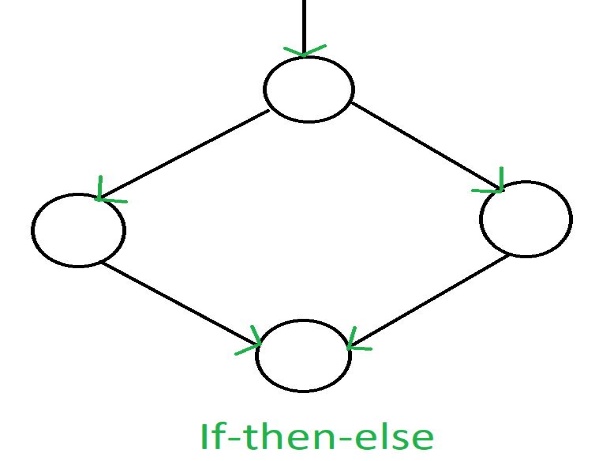
* **Entry Block: The entry block allows the control to enter into the control flow graph.**
* **Exit Block: Control flow leaves through the exit block.**

**Hence, the control flow graph comprises all the building blocks involved in a flow diagram such as the start node, end node, and flows between the nodes.**

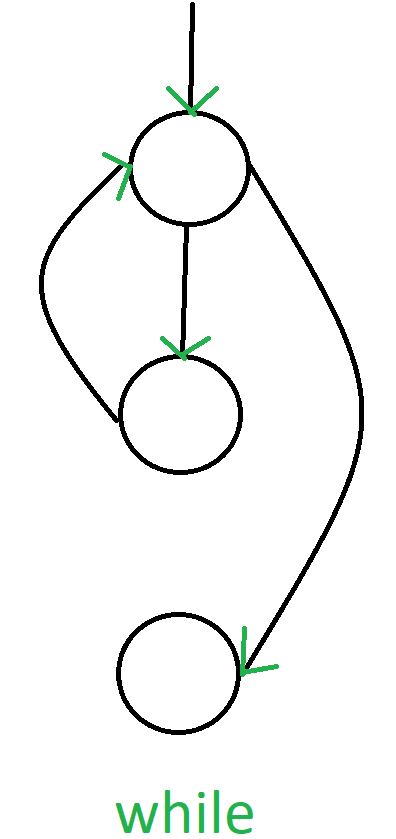
**General Control Flow Graphs**

**Control Flow Graph is represented differently for all statements and loops. The following images describe it:**

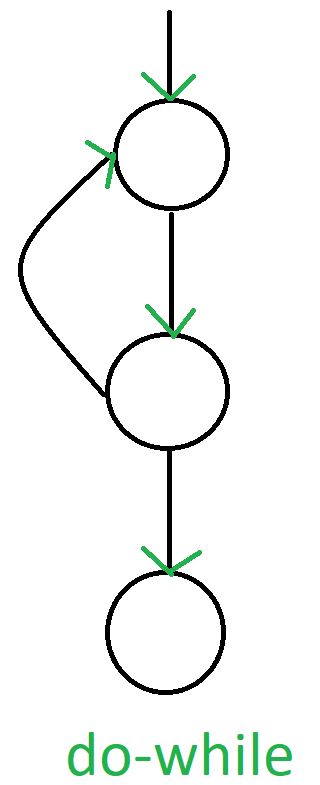
**1. If-else**



**2. While**



**3. do-while**



**4. for**

