**IO Package**

**Input and Output Operation**

**Io package contains lot of classes and interfaces which help to do File handling programming**

**Stream : Flow of data or it is an abstraction between source and destination**

**Stream**

**byte char**

**Input Output Input Output**

**InputStream OutputStream Reader Writer**

**All 4 classes are abstract class part of io package.**

**DataInputStream,DataOutputStream ,InputStreamReader, OutputStreamWriter**

**BufferedInputStream, BufferedOutputStream, BufferedReader, BufferedWriter**

**FileInputStream , FileOutputStream FileReader , FileWriter**

**ObjectInputStream , ObjectOutputStream PrintWriter**

**PrintStream**

**System.in**

**System.out.println(“Welcome to Java”);**

**Buffer is known temporary memory which help to improve the performance while doing file handling or database connectivity.**

**If we use buffer concept it will improve 1000 times read and write operation.**

**File1**

**Buffer**

**Program**

**Bufffer**

**File2**

**InputStream abc = System.in**

**InputStream always refer to standard input device : keyword.**

**PrintStream ps = System.out;**

**ps.println(“Welcome to Java”);**

**Object Serialization : converting object state into byte format or serializable format.**

**Employee class object**

**Property id,name,salary**

**Behaviour readEmp, disEmp**

**Identity emp1**

**We can’t store or transfer in network environment their behaviour or identity.**

**JDBC: Java Database Connectivity**

**Database : it use to store the data in table format.**

**We can store the data permanently**

1. **File system**
2. **Database system**

**Limitation of file system**

1. **Data redundancy: mean we can store duplicate records.**
2. **Data In consistency:** 
   1. **Format the file : .doc, pdf, .txt, csv etc**
   2. **Delimiter :**

**Id,Name,salary**

**1,Ravi,12000**

**1 Ravi 1200**

**1/Ravi/12000**

**1 Ravi 12000**

**3. security : read mode or write mode or execute mode**

**4. CRUD Operation : creating / insert, Read , Update and Delete. Complex task on file system.**

**Database :**

**MySQL : My SQL is a type of RDBMS. which allow to store data in table format.**

**To interact with SQL Database or RDBMS database**

**SQL : Structured Query language**

**Mainly divided into 5 types.**

1. **DRL or DQL (Data Query / Retrieval language)**

**Select clause (View purpose)**

1. **DDL (Data definition language)**

**Create, drop, alter , rename, truncate etc. (Structure of the table)**

1. **DML (Data manipulation language)**

**Insert, Delete and Update (deal with data )**

1. **TCL (Transactional control language)**

**Commit, savepoint and rollback etc**

1. **DCL (Data control language)**

**Create the account, grant the permission and revoke the permission etc.**

**show databases; show all database**

**create database databaseName; create new database**

**use databaseName; switch inside existing database.**

**show tables; it is use to display all tables present in db.**

**Product -🡪 Name**

**Pid(PK) PName price column**

**int varchar(30) float**

**table creation syntax**

**create table product(pid int primary key,pname varchar(30), price float);**

**view the table structure**

**desc product**

**insert the record into table**

**insert into product values(100,'TV',67000);**

**insert into product values(101,'Computer',45000);**

**insert into product values(102,'Laptop',97000);**

**view the records from a table**

**select \* from product;**

**JDBC**

**Java Database Connectivity**

**JDBC is a API (classes / interfaces). Which help to connect any RDBMS or no SQL database like mysql, oracle, db2, postgres or mongo db etc. Using Java program we can insert, delete, update and retrieve records.**

**Steps**

1. **import the package java.sql.\* or javax.sql.\*;**
2. **we need to use exception handling concept try-catch or throws. Because JDBC throw check exception.**
3. **Load the Driver. Driver is a pre defined class which help to connect the database.** 
   1. **4 types of Driver** 
      1. **Type 1 or JDBC ODBC bridge driver : from java8 onward type1 driver removed or not supported.**
      2. **Type 2 or JDBC Native API driver**
      3. **Type 3 or JDBC net protocol driver**
      4. **Type 4 or Jdbc pure or thin driver**

**Class.forName(“DriverName”);**

1. **Establish the connection**

**Connection con = DriverManager.getConnection(url,username,password);**

1. **Create Type Statement or PreparedStatement or CallableStatement**

**Statement stmt = con.createStatement();**

**stmt.executeUpdate(“DML Operation”);**

**or**

**stmt.executeQuery(“select Query”)**

1. **Close the resource like stmt.close(), con.close() etc.**