**STOCK ANALYZER**

B. VASISTA AVINASH (13471A0567)

CH.VAMSI (13471A0574)

B.SIVA RAMA KRISHNA (13471A05B1)

**Under the guidance of**

Mr. G. Raphi



COMPUTER SCIENCE AND ENGINEERING

**NARASARAOPETA ENGINEERING COLLEGE**

**NARASARAOPETA**

**(Signature of Guide) (Signature of HOD)**

**DATABASE DESIGN**

**Database Design**

**Constraints:**

* NOT NULL
* UNIQUE
* PRIMARY KEY / CANDIDATE KEY
* FOREIGN KEY
* CHECK
* DEFAULT CONSTRAINT

**Not Null:**

* Not Null constraint is used to avoid null values for a specific column.
* Not Null should be defined only at column level.

**Unique:**

* UNIQUE constraint ensures that a field or column will only have unique values.
* A UNIQUE constraint field will not have duplicate data.
* UNIQUE constraint can be applied at column level or table level.

**Key**: A key is an attribute or set of attributes in a relation that identifies a tuple in a relation.

**Primary Key:**

* Primary key constraint uniquely identifies each record in a database.
* Primary key doesn’t allow duplicates and null values (combination of UNIQUE and NOT NULL).
* Usually Primary Key is used to index the data inside the table.
* Only one primary key is allowed per table.

**Candidate Key:**

* A candidate key is a column, or set of columns, in a table that can uniquely identify any dB record without referring to any other data.
* Each table may have one or more candidate keys, but one candidate key is special, and it is called the primary key. This is usually best among the candidate keys.
* When a key is composed of more than one column, it is known as a composite key.

**Foreign Key:**

* Foreign keys are the columns of a table that points to the primary key of another table. They acts as a cross-reference between tables.
* The foreign key constraint provides referential integrity rules (either within table or b/w tables) i.e., we can place a value in TABLE B if the values exist as a primary key in TABLE A.
* We use foreign key if we want to ensure that for every child table record there is a reference in parent table.

**Check:**

* CHECK constraint is used to restrict the value of a column between a range.
* It performs check on the values, before storing them into the database.
* It is like condition checking before saving data into a column.
* The check constraint can be defined at the column level or table level.

**Default Constraint**:

* The DEFAULT constraint is used to insert a default value into a column.
* The default value will be added to all new records, if no other value is specified.
* The default value can be a literal, an expression, or sql function.

**Normalization of Database**

If a database design is not perfect it may contain anomalies, which leads to inconsistence of database itself. Normalization is the process of efficiently organizing data in the DB.

There are two goals of the normalization process:

1. Eliminate redundant data( for example, storing the same data in more than one table ) and
2. Ensure data dependencies make sense (only storing related data in a table).

Both of these worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored.

**First Normal Form (1NF):**

A relation is a said to be in 1NF if it contains no non-atomic values and each row can provide a unique combination of values.

Second Normal Form (2NF):

A relation is said to be in 2NF if it is already in 1NF and each and every attribute fully depends on the primary key of the relation. Speaking inversely, if a table has some attributes which is not dependent on the primary key of that table, then it is not in 2NF.

Full dependency: Non-key attributes are depended on key attributes.

Partial dependency: Non-key attributes are depended on part of key attributes.

**Third Normal Form (3NF):**

A relation is said to be in 3NF, if it is already in 2NF and there exists no transitive dependency in that relation. Speaking inversely, if a table contains transitive dependency, then it is not in 3NF, and the table must be split to bring it in 3NF.

**Advantages of Normalization:**

* Avoid redundant fields or columns.
* Better understanding of data.
* Ensure that distinct tables exits when necessary.

**Disadvantages of normalization:**

* You cannot start building the DB before you know what the user needs.
* It is very time consuming & difficult process in normalizing relations of high degree.

On normalizing the relations to higher normal forms i.e. 4NF, 5NF the performance degrades.

**Database Tables:**

The total number of database tables that were identified to build the system is 7. The major part of the

Database is categorized as

• **Transactional components:** The Transactional components are useful in recording the transactions made by the system. All the inwards, deliveries, returns etc information handled by these components.

• **Data Dictionary components:** These components are used to store the major information like Employee details, Godown details, Customer details, Items information etc.

• **General components:** These components are used to store the general information like login information etc.

**DATABASE DESIGN (TABLES):**

1. **Table** **name**: **godown**

**Description**: To manage godown details.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1. | GodownID | Number(10) | Primary key | Godown identification |
| 2. | Location | Varchar2(15) | Not Null | Godown location |
| 3. | CapacityInQunitals | Number(5) | Not Null | Capacity in quintal |
| 4. | GManager | Varchar2(20) | Not Null | Godown manager |
| 5. | GSDate | Date | Not Null | Godown start date |
| 5. | StockInQunitals | Number(5) | Not Null | Stock in quintals |

2. **Table** **name**: **Emp**

**Description**: user details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1. | EmpName | Varshar2(20) | Not Null | Name of employee |
| 2. | EmpID | Number(10) | P Key ,NN, Unique | Employee ID |
| 3. | GodownID | Number(10) | Foreign Key | Work location |
| 4. | Role | Varchar2(15) | Not Null | Designation |
| 5. | JoiningDate | Date | Not Null | Date of join |
| 6. | Password | Varchar2(45) | Not Null | Password of emp |

3. **Table name**: **Inwards**

**Description**: details of stocks received

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1. | GodownID | Number(10) | Foreign key | Godown id |
| 2. | SID | Number(10) | Foreign key | ID of supplier |
| 3. | ItemID | Number(10) | Foreign Key | Item ID |
| 4. | DateOfSupply | Date | Not Null | Item supply date |
| 5. | Quantity | Number(5) | Not Null | Quantity received |
| 6. | ReceiptNo | Number(10) | Not Null | Receipt generated |
| 7. | REmpID | Number(10) | Foreign Key | Employee who received |
| 8. | Bill | Number(10) | Not Null | Bill generated |
| 9. | CEmpID | Number(10) | Foreign Key | Employee who checked bill |
| 10. | Invoice | Number(10) | Primary key | Inward supply details |

4. **Table name**: **delivery**

**Description**: details of stocks delivered

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1. | GodownID | Number(5) | Foreign key | Godown id |
| 2. | ItemID | Varchar2(20) | Foreign Key | Item id |
| 3. | DateOfSupply | date | Not Null | Item supplied date |
| 4. | Invoice | BIGINT(20) | Primary key | delivery details |
| 5. | DateOfDelivery | Date | Not Null | Item delivery date |
| 6. | Quantity | Number(10) | Not Null | Quantity delivered |
| 7. | ReceiptNo | Number(10) | Not Null | Receipt generated |
| 8. | CID | Number(10) | Foreign Key | Delivered to customer |
| 9. | Bill | Number(10) | Not Null | Bill generated |
| 10. | EmpID | Number(10) | Foreign Key | Bill checked by employee |

5. **Table name**: **return**

**Description**: details of stocks returned

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1. | GodownID | Number(5) | Foreign key | Godown id |
| 2. | ItemID | Number(10) | Foreign key | Item number |
| 3. | DateOfDelivery | Date | Not Null | Item delivered date |
| 4. | Invoice | Number(20) | Primary key | return details |
| 5. | DateOfReturn | Date | Not Null | Item return date |
| 6. | Quantity | Number(10) | Not Null | Quantity returned |
| 7. | Purpose | Varchar2(20) | Not Null | Purpose of return |
| 7. | ReceiptNo | Number(10) | Not Null | Receipt generated |
| 8. | CID | Number(10) | Foreign key | Returned by customer |
| 9. | Bill | Number(10) | Not Null | Bill generated |
| 10. | EmpID | Number(10) | Foreign Key | Bill checked by employee |

6. **Table name**: **Stock**

**Description**: details of stocks in godowns

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1. | GodownID | Number(5) | Primary key | Godown identification |
| 2. | ItemID | Number(10) | Foreign Key | Item ID |
| 3. | InitialStockDate | date | Not Null | Starting stock date |
| 4. | InitialStock | Number(10) | Not Null | Initial stock amount |
| 5. | ReceivedStock | Number(10) | Not Null | Stock received |
| 6. | DeliveredStock | Number(10) | Not Null | Stock delivered |
| 7. | ReturnedStock | Number(10) | Not Null | Stock returned |
| 8. | TotalStock | Number(10) | Not Null | Total stock |

7. **Table name**: **Item**

**Description**: details of items

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1 | ItemID | Number(10) | Primary Key | Item ID |
| 2 | ItemName | Varchar2(20) | NOT NULL | Item name |
| 3 | Itemdetails | Varchar2(20) | NOT NULL | Item Details |

8.**Table name**: **Supplier**

**Description**: details of supplier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1 | SID | Number(10) | Composite PKey | Supplier ID |
| 2 | SName | Varchar2(20) | NOT NULL | Supplier Name |
| 3 | ItemID | Number(10) | Composite PKey | Item of supplier |

9.**Table name**: **Customer**

**Description**: details of Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ATTRIBUTE** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| 1 | CID | Number(10) | Composite PKey | Customer ID |
| 2 | CName | Varchar2(20) | NOT NULL | Customer Name |
| 3 | ItemID | Number(10) | Composite PKey | Item of Customer |