**Database Design**

**Introduction:**

* **Database:** A Database is collection of related data, which can be of any size and complexity. By using the concept of database, we can easily store and retrieve the data. The major purpose of a database is to provide the information, which utilizes it with the information that the system needs according to its own requirements.
* **Database Design:** Database design is done before building it to meet needs of end-users within a given information-system that the database is intended to support. The database design defines the needed data and data structures that such a database comprises.

The database is physically implemented using MySQL.

The database for onlinegiftshop is organized into 11 tables:

* address
* administrator
* billing
* category
* customer
* location
* notification
* products
* purchase
* shopowner
* subcategory

Each entity can be described as follows along with its attributes:

## Table name: address

The address table stores customer address records.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Address\_id** | Integer(10) | Primary key | Address id |
| Cust\_id | Integer(10) | Foreign key | Customer id |
| Address | Text | Not null | Address |
| State | Varchar(25) | Not null | State |
| Country | Varchar(10) | Not null | Country |
| Pincode | Varchar(15) | Not null | Pin code |
| Countact no | Integer(15) | Not null | Countact number |

## Table name: administrator

The administrator table stores item administrator records.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Admin\_id** | Integer(10) | Primary key | Administrator id |
| Name | Varchar(25) | Foreign key | Name |
| Login\_id | Varchar(30) | Foreign Key | Login id |
| A\_password | Varchar(30) | Foreign key | Administrator password |
| Email\_id | Varchar(30) | Not null | Email id |

## Table name: billing

The billing table stores billing details.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Bill\_id** | Integer(10) | Primary key | Billing id |
| Cust\_id | Integer(10) | Foreign key | Customer id |
| Purch\_date | Date | Not Null | Purchase id |
| Deliv\_date | Varchar(25) | Not null | Delivery date |
| Card type | Varchar(20) | Not null | Card type |
| Card no | Varchar(5) | Not null | Card number |
| Cvvno | Varchar(10) | Not null | 3 digit CVV number |
| Expirydate | Date | Not null | Expiry date |
| Comment | Text | Not null | Comment |
| Status | Varchar(10) | Not null | Status |

## Table name: category

The category table stores category details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Cat\_id** | Integer(10) | Primary key | Category id |
| Cat\_name | Varchar(20) | Not Null | Category id |
| Cat\_des | Text | Not Null | Category description |

## Table name: customer

The customer table customer details. It is master table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Cust\_id** | Integer(11) | Primary key | Customer ID |
| CustFname | Varchar(25) | Not null | Customer first name |
| CustLname | Varchar(25) | Not null | Customer last name |
| Dob | Date | Not null | Date of birth |
| Email | Varchar(15) | Not null | Email ID |
| Cpassword | Varchar(15) | Not null | Customer password |
| Status | Varchar(10) | Not null | Status |

## Table name: product

The product table product details.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Index** | **Description** |
| **Purch\_id** | Integer(10) | Primary key | Purchase id |
| Prod\_id | Integer(10) | Foreigen key | Product id |
| Size\_id | Integer(10) | Foreigen key | Size id |
| Cust\_id | Integer(10) | Foreigen key | Customer id |
| Bill\_id | Integer(10) | Foreigen key | Bill id |
| Qty | Float(10,2) | Not null | Total purchased Quantity |
| Price | Double(10,2) | Not null | Price |
| Comments | Text | Not null | Any comments |

## Table name: shopowner

The shopowner table stores showowner records,

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Shop\_id** | Integer(10) | Primary key | Shopowner id |
| Compname | Varchare(25) | Not null | Company name |
| Address | Text | Not null | Address |
| State | Varchare(20) | Not null | State |
| Country | Varchare(20) | Not null | Country |
| Contact-no | Varchare(20) | Not null | Contact number |
| Login\_id | Varchare(20) | Not null | Login id |
| S\_password | Varchare(20) | Not null | Shopowner  Password |
| Last\_login | Datetime | Not null | Last login |
| Created\_at | Datetime | Not null | Created at |
| Imagepath | Varchare(100) | Not null | Image path |
| Status | varchare(100) | Not null | Status |
| Shop\_id | Integer(10) | Primary key | Shopowner id |
| Compname | Varchare(25) | Not null | Company name |
| Address | Text | Not null | Address |
| State | Varchare(20) | Not null | State |
| Country | Varchare(20) | Not null | Country |
| Contact-no | Varchare(20) | Not null | Contact number |
| Login\_id | Varchare(20) | Not null | Login id |
| S\_password | Varchare(20) | Not null | Shopowner  Password |
| Last\_login | Datetime | Not null | Last login |
| Created\_at | Datetime | Not null | Created at |
| Imagepath | Varchare(100) | Not null | Image path |
| Status | varchare(100) | Not null | Status |

## Table name: subcategory

The subcategory table stores sub category details.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data type** | **Index** | **Description** |
| **Subcat\_id** | Integer(10) | Primary key | Subcategory id |
| Cat\_id | Integer(10) | Foreigen key | Category id |
| Subcategory | Varchare(25) | Not null | Subcategory |
| Description | Text | Not null | Description |

**Entity-Relationship Diagram:**

An entity-relationship (ER) diagram is a specialized graphic that illustrates the [relationships between entities in a database](http://databases.about.com/od/specificproducts/a/Database-Relationships-An-Introduction-To-Foreign-Keys-Joins-And-E-R-Diagrams.htm). ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes.

The Symbols are shown in below table:

|  |  |  |
| --- | --- | --- |
| **Name** | **Notation** | **Description** |
| Entity |  | Entity is represented by a box within the ERD. Entities are abstract concepts, each representing one or more instances of the concept in question. An entity might be considered a container that holds all of the instances of a particular thing in a system. Entities are equivalent to database tables in a relational database, with each row of the table representing an instance of that entity. |
| Relationship |  | Relationships are represented by Diamonds. A relationship is a named collection or association between entities or used to relate to two or more entities with some common attributes or meaningful interaction between the objects. |
| Attributes |  | Attributes are represented by Oval. An attribute is a single data item related to a database object. The database schema associates one or more attributes with each database entity. |

**ER Diagram:**

