How NoSQL Maps to SQL Schema:

- Tables vs. Collections: SQL tables like User, Product, and Order are translated into separate collections in NoSQL.
- Primary/Foreign Keys vs. Object References: Relationships in SQL (e.g., foreign keys) are handled by embedding documents or referencing other document IDs in NoSQL.
- Normalization vs. Denormalization: SQL often normalizes data across multiple tables to reduce redundancy, while NoSQL tends to denormalize by embedding related data directly within documents to optimize read performance.

NoSQL Examples :

User Collection :

```
{
 "_id": ObjectId("..."),
 "Fname": "John",
 "Mname": "D",
 "Lname": "Doe",
 "Email": "john.doe@example.com",
 "Age": 30,
 "Date_of_birth": "1994-05-12",
 "Phone": "12345678901",
 "Password": "hashed_password",
 "Ssn": "12345678901234",
 "Admin": {
  "AdminId": ObjectId("..."),
  "Salary": 50000.00,
  "Date_of_join": "2020-01-15"
 },
 "Customer": {
  "CustomerId": ObjectId("..."),
  "Bank_card": "1234567812345678",
  "Cart": {
    "CartId": ObjectId("..."),
    "Total_cost": 299.99,
    "Items": [
      {
        "ItemId": ObjectId("..."),
        "Quantity": 2,
        "ProductId": ObjectId("...")
      }
    ]
  }
 }
}
```

Product Collection Example

```
{
  "_id": ObjectId("..."),
  "Name": "Laptop",
  "Description": "A powerful laptop",
  "Price": 999.99,
  "Average_rate": 4.5,
  "CategoryId": ObjectId("..."),
  "Images": [
    {
      "ImageId": ObjectId("..."),
      "image": "image1.jpg"
    },
    {
      "ImageId": ObjectId("..."),
      "image": "image2.jpg"
    }
  ],
  "Reviews": [
    {
      "ReviewId": ObjectId("..."),
      "CustomerId": ObjectId("..."),
      "Comment": "Great product!",
      "Rate": 5.0
    }
  ]
}
```

Category Example

```
{
  "_id": ObjectId("..."),
  "name": "Electronics",
  "Description": "Devices and gadgets",
  "ParentCategory": ObjectId("...")
}
```

Order Example

```
{
  "_id": ObjectId("..."),
  "CustomerId": ObjectId("..."),
  "Order_Date": "2024-01-10",
  "OrderDetails": {
    "OrderDetailsId": ObjectId("..."),
    "Taxes": 50.00,
    "Discount": 10.00,
    "Items": [
      {
        "ItemId": ObjectId("..."),
        "Quantity": 2,
        "ProductId": ObjectId("...")
      }
    ]
  },
  "PaymentInformation": {
    "Payment_date": "2024-01-10",
    "Payment_method": "Visa"
  }
}
```