

|  |
| --- |
| Business Template  **La Galerie Dior MUSEUM:**  **daTA MODEL** |
|  |

  
(https://www.deluxefrance.com/details-discovering+dior+an+exclusive+museum+tour+and+shopping+experience+in+paris-527)

Contents

[1 Business Description 3](#_Toc62212630)

[1.1 Business background 3](#_Toc62212631)

[1.2 Problems. Current Situation 3](#_Toc62212632)

[1.3 The benefits of implementing a database. Project Vision 3](#_Toc62212633)

[2 Model description 3](#_Toc62212634)

[2.1 Definitions & Acronyms 3](#_Toc62212635)

[2.2 Logical Scheme 3](#_Toc62212636)

[2.3 Objects 3](#_Toc62212637)

# 

# Business Description

## Business background

La Galerie Dior Museum showcases and preserves items related to the Dior brand, like clothing, artworks, and historical objects. The museum hosts exhibitions and manages a large collection of items stored across different locations. Keeping track of everything and planning exhibitions can be a challenge without a proper system.

## Problems. Current Situation

Manual management of artifacts and inventory often leads to errors, misplaced items, and inefficiency. Exhibition planning lacks a centralized system for scheduling and artifact allocation. Employees data are scattered, making analytics and operations tedious and reporting time-consuming.

## the Benefits of implementing a database. Project Vision

The database could centralize all museum operations, making it easier to manage collections, exhibitions and staff data.

* keep track of all artifacts and storage efficiently,
* plan exhibitions smoothly by connecting them to artifacts,
* access and analyze employee information easily,
* save time by automating reports and managing storage capacity

# Model description

## Definitions & Acronyms

* PK – primary key, unique
* FK - foreign key
* INT - integer type, simple number
* VARCHAR – string type
* DECIMAL – floating point number
* GENERATED ALWAYS AS IDENTITY - auto-incrementing column
* NOT NULL – can’t have empty value
* CHECK – constraint to put a certain condition
* DATE – a date, DD-MM-YYYY

## Logical Scheme

Obraz zawierający tekst, zrzut ekranu, Czcionka, design

Opis wygenerowany automatycznie

*Fig. LA GALERIE DIOR MUSEUM: LOGICAL MODEL*

## Objects

* **Storage:** Places where artifacts are kept when not on display

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Storage | **storage\_id** | Unique identifier for storage location, PK | INT, GENERATED ALWAYS AS IDENTITY |
| **location** | Storage location | VARCHAR(100), NOT NULL |
| **capacity** | Maximum number of items the storage can hold | INT, NOT NULL, CHECK (capacity > 0) |
| **space\_taken** | Number of items currently stored | INT, CHECK (space\_taken >= 0 AND space\_taken <= capacity) |

**Comments on table relationships:**

* **Storage & Artifacts**: many artifacts are stored in a single location (1-to-many), each artifact can physically be only in one location;

**Example with data**

|  |  |  |  |
| --- | --- | --- | --- |
| **storage\_id** | **location** | **capacity** | **space\_taken** |
| *1* | *"Paris Main Gallery"* | *200* | *NULL* |

* **Artifacts**: Items in the collection (like dresses or accessories),

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Artifacts | **artifact\_id** | Unique identifier for each artifact, PK | INT, GENERATED ALWAYS AS IDENTITY |
| **storage\_id** | Storage location reference, FK | INT, NOT NULL REFERENCES Storage(storage\_id) |
| **name** | Name of the artifact | VARCHAR(100), NOT NULL DEFAULT 'Anonymous' |
| **style** | Style or category of the artifact | VARCHAR(100), NOT NULL |
| **creation\_date** | Date the artifact was created | DATE, CHECK (creation\_date <= CURRENT\_DATE) |
| **artist** | Creator or designer of the artifact | VARCHAR(100), NOT NULL |

**Comments on table relationships:**

* **Storage & Artifacts**: many artifacts are stored in a single location (1-to-many), each artifact can physically be only in one location;
* **Artifacts & Maintenance**: each artifact can have multiple maintenance tasks associated with it over time (1-to-many), but each maintenance task is linked to only one artifact;
* **Artifacts & Exhibitions:** many artifacts can be part of many exhibitions (many-to-many, added bridge);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **artifact\_id** | **storage\_id** | **name** | **style** | **creation\_date** | **artist** |
| *1* | *1* | *"Lady Dior Bag"* | *"Accessories"* | *"2020-01-15"* | *"Christian Dior"* |

* **Exhibition\_Artifacts** - Bridge table representing the many-to-many relationship between artifacts and exhibitions;

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Exhibition\_Artifacts | **exhibition\_id** | Identifier for exhibition, FK | INT REFERENCES Exhibition(exhibition\_id) |
| **artifact\_id** | Identifier for artifact, FK | INT REFERENCES Artifacts(artifact\_id) |
| **primary Key** | Combination of (exhibition\_id, artifact\_id) | Composite Key |

|  |  |
| --- | --- |
| **exhibition\_id** | **artifact\_id** |
| *3* | *5* |

* **Exhibition:** Events that display the artifacts for visitors,

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Exhibition | **exhibition\_id** | Unique identifier for each exhibition, PK | INT, GENERATED ALWAYS AS IDENTITY |
| **title** | Title of the exhibition | VARCHAR(200), UNIQUE, NOT NULL |
| **start\_date** | Start date of the exhibition | DATE, CHECK (start\_date > '2024-07-01') |
| **end\_date** | End date of the exhibition | DATE, CHECK (end\_date IS NULL OR end\_date > start\_date) |

**Comments on table relationships:**

* **Exhibition &Tickets:** tickets are associated with a single exhibition (1-to-many), one ticket is one exhibition;
* **Artifacts & Exhibitions:** many artifacts can be part of many exhibitions (many-to-many, added bridge);

|  |  |  |  |
| --- | --- | --- | --- |
| **exhibition\_id** | **title** | **start\_date** | **end\_date** |
| *1* | *"Christian Dior: Designer of Dreams"* | *"2024-08-01"* | *"2024-12-01"* |

* **Tickets:** Proof of paying a fee to come to the museum,

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Tickets | **ticket\_id** | Unique identifier for each ticket, PK | INT, GENERATED ALWAYS AS IDENTITY |
| **exhibition\_id** | Exhibition the ticket belongs to, FK | INT REFERENCES Exhibition(exhibition\_id) |
| **price** | Price of the ticket | DECIMAL(4, 2), NOT NULL CHECK (price > 0) |
| **ticket\_type** | Type of ticket | VARCHAR(15), NOT NULL CHECK (ticket\_type IN ('Normal', 'Discounted', 'VIP')) |
| **purchase\_date** | Date when the ticket was purchased | DATE NOT NULL DEFAULT CURRENT\_DATE |

**Comments on table relationships:**

* **Exhibition &Tickets:** tickets are associated with a single exhibition (1-to-many), one ticket is one exhibition;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ticket\_id** | **exhibition\_id** | **price** | **ticket\_type** | **purchase\_date** |
| *1* | *1* | *20.00* | *"Normal"* | *"2024-09-15"* |

* **Employees:** Museum staff members,

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Employees | **employee\_id** | Unique identifier for each employee, PK | INT, GENERATED ALWAYS AS IDENTITY |
| **name** | First name of the employee | VARCHAR(100), NOT NULL |
| **surname** | Last name of the employee | VARCHAR(100), NOT NULL |
| **job** | Job role within the museum | VARCHAR(30), NOT NULL |

**Comments on table relationships:**

* **Employees & Maintenance:** An employee can perform multiple maintenance tasks (1-to-many), but each maintenance task is assigned to a single employee;

|  |  |  |  |
| --- | --- | --- | --- |
| **employee\_id** | **name** | **surname** | **job** |
| *2* | *"Marie"* | *"Curie"* | *"Director"* |

* **Maintenance:** Necessary works of restoration, preservation, cleaning

|  |  |  |  |
| --- | --- | --- | --- |
| * Table Name | Field name | Field Description | Data Type |
| Maintenance | **maintenance\_id** | Unique identifier for maintenance task, PK | INT, GENERATED ALWAYS AS IDENTITY |
| **artifact\_id** | Artifact being maintained, FK | INT REFERENCES Artifacts(artifact\_id) |
| **employee\_id** | Employee responsible, FK | INT REFERENCES Employees(employee\_id) |
| **date\_performed** | Date when maintenance was performed | DATE |
| **next\_due\_date** | Date for the next maintenance check | DATE, CHECK (next\_due\_date IS NULL OR next\_due\_date > date\_performed) |

**Comments on table relationships:**

* **Employees & Maintenance:** An employee can perform multiple maintenance tasks (1-to-many), but each maintenance task is assigned to a single employee;
* **Artifacts & Maintenance**: Each artifact can have multiple maintenance tasks associated with it over time (1-to-many), but each maintenance task is linked to only one artifact;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **maintenance\_id** | **artifact\_id** | **employee\_id** | **date\_performed** | **next\_due\_date** |
| *60* | *8* | *2* | *"2024-09-10"* | *"2024-12-10"* |