

# Projet Base de Données : Fashion Week

UVSQ



université PARIS-SACLAY

Abdelhak MERABTENE  
Slimane BOUAROUA  
Louiza MAKHOUKHENE  
Sous la supervision de :  
Mme. Beatrice FINANCE

# Table des matières

<b>Introduction.....</b>	<b>4</b>
<b>Description.....</b>	<b>5</b>
<i>Créateurs :.....</i>	<i>5</i>
<i>Tenues :.....</i>	<i>5</i>
<i>Collection :.....</i>	<i>6</i>
<i>Défilé :.....</i>	<i>7</i>
<i>Mannequin :.....</i>	<i>8</i>
<i>Maison de mode :.....</i>	<i>9</i>
<i>Invité :.....</i>	<i>9</i>
<i>Journalistes :.....</i>	<i>10</i>
<i>Sponsors :.....</i>	<i>10</i>
<b>Modélisation.....</b>	<b>11</b>
<i>Schéma Entité/Association.....</i>	<i>11</i>
<b>Contraintes d'intégrité.....</b>	<b>12</b>
<b>Droits et vues – confidentialité des données.....</b>	<b>14</b>
<b>Requêtes.....</b>	<b>16</b>
<b>Schéma E/A en relationnel.....</b>	<b>18</b>
<b>Conclusion.....</b>	<b>19</b>
<b>Implémentation de BD sous Oracle.....</b>	<b>20</b>
<i>A/ Création du schéma de la base de données.....</i>	<i>20</i>
<i>B/ Jeu de données.....</i>	<i>25</i>
<i>C/ Manipulation des données.....</i>	<i>36</i>
<i>D/ Vues.....</i>	<i>43</i>
<i>E/ Intégrité des données : les triggers.....</i>	<i>48</i>
<i>F/ Méta-données.....</i>	<i>61</i>



## Introduction

La Fashion Week est un événement incontournable dans l'univers de la mode, se déroulant deux fois par an dans des villes emblématiques telles que Paris, Milan, New York et Londres. Ce rendez-vous prestigieux réunit les plus grands créateurs et maisons de mode, qui dévoilent leurs nouvelles collections devant un public composé de journalistes, d'acheteurs et d'influenceurs. La Fashion Week joue un rôle essentiel dans la définition des tendances futures et oriente l'industrie de la mode à l'échelle mondiale.

Chaque édition offre aux créateurs l'opportunité d'exprimer leur vision artistique, tout en influençant directement le marché de la mode. L'organisation d'un tel événement nécessite une gestion rigoureuse des informations, incluant la coordination des invités, la sélection des lieux, la planification des horaires des défilés ainsi que la présentation des collections.

La complexité de cette organisation met en lumière l'importance d'une structure de données efficace pour assurer une coordination fluide et une expérience optimale pour tous les participants. C'est dans cette optique que nous avons choisi d'en faire l'objet de notre base de données.

## Description

### Créateurs :

Les créateurs sont les architectes de l'évènement. Leur objectif est de présenter des collections inédites, chaque pièce étant le reflet de mois de travail acharné. Prenons l'exemple d'une maison comme Dior, où chaque défilé est soigneusement planifié pour captiver l'audience et marquer les esprits. Cependant, un créateur ne peut participer que s'il présente au moins une collection complète.

Il ne peut y avoir de défilés sans le travail des créateurs, ils sont responsables d'imaginer et de dessiner les modèles qui composent leurs collections de haute couture et de prêt-à-porter à présenter, lors des saisons à venir (printemps/été, automne/hiver). Il est important de stocker les informations relatives à chaque créateur, tel que leurs couturiers avec qui ils travaillent, leurs défilés passés et à venir ainsi que diverses informations relatives à ces derniers tel que la nationalité, l'âge, etc.

Chaque créateur peut être responsable de plusieurs collections au fil des saisons, chacune ayant sa propre identité visuelle et son style.

Les créateurs s'occupent également de la création de chaque tenue.

Les créateurs sont affiliés à des maisons de mode. Ils créent des collections pour le compte de la maison de mode, en respectant son image et ses valeurs.

### Tenues :

Les tenues constituent l'élément central des collections présentées lors de la Fashion Week. Elles incarnent le style, la créativité et la signature des créateurs. Chaque tenue est unique et reflète la vision artistique du créateur tout en contribuant à l'identité globale de la collection à laquelle elle appartient.

Les tenues sont organisées en collections, qui suivent un thème ou un style défini.

Les créateurs sont à l'origine de la conception de chaque tenue. Chaque tenue est ainsi directement associée au créateur qui l'a imaginée.

Lorsqu'un créateur est associé à une maison de mode, les tenues qu'il conçoit durant sa présence dans cette maison de mode, peuvent être produites et présentées sous le nom de cette maison.

Les tenues peuvent ne pas avoir de prix si elles ne sont pas à vendre.

Les tenues doivent être à la taille d'au moins un mannequin.

Une tenue ne peut appartenir qu'à une seule collection, garantissant la cohérence de chaque ensemble.

## Collection :

Les collections sont au cœur de la Fashion Week, représentant les ensembles de créations artistiques conçues par les créateurs et présentées lors des défilés. Chaque collection incarne la vision du créateur pour une saison spécifique, illustrant les tendances de la mode à venir. Une collection regroupe un certain nombre de tenues, qui sont les pièces principales exposées lors des événements.

Une collection ne peut être présentée que si elle contient au moins 10 tenues, sinon elle ne pourrait pas être intégrée à un défilé.

Chaque collection est conçue par un créateur spécifique.

Une collection est un ensemble de tenues conçu autour d'un thème et d'une saison spécifique (printemps/été, automne/hiver).

Un créateur peut créer plusieurs collections dans une même année, mais une collection appartient toujours à un seul créateur.

Une maison de mode peut produire plusieurs collections au cours d'une saison, mais chaque collection ne peut être associée qu'à une seule maison de mode.

Chaque collection est composée de plusieurs tenues.

Une collection ne peut être présentée dans plus d'un défilé, afin de maintenir l'exclusivité et l'impact de la présentation.

Les mannequins portent les tenues d'une collection lors des défilés. Chaque mannequin peut être associé une seule ou à plusieurs tenues d'une même collection.

La collection doit être validée par la maison de mode avant le défilé, afin de garantir que la collection reflète bien l'identité de la maison de mode, respecte le thème demandé.

## Défilé :

Les défilés, communément appelés les Show, sont le résultat de longues heures de travaux des créateurs sur les tenues de la collection à présenter, il est de notre devoir de stocker les informations relatives à chaque défilé tel que : le thème du défilé, les horaires, le lieu, ainsi que toutes les informations, qui sont bonnes au bon déroulement de cet événement. Cependant, chaque défilé doit avoir des mannequins pour pouvoir présenter les collections.

Chaque défilé présente une collection, chacune composée de plusieurs tenues qui défilent devant les invités.

Les informations sur les créateurs participant à chaque défilé doivent être bien documentées pour savoir quels créateurs ont présenté leur travail lors de tel ou tel défilé, et pour suivre leur présence à ces événements au fil des saisons.

Une maison de mode présente sa collection lors d'un défilé unique.

Chaque défilé est animé par plusieurs mannequins.

L'accès aux défilés est souvent restreint, des invitations spécifiques sont obligatoires pour chaque invité, afin d'y assister.

Chaque défilé est assisté par plusieurs invités, préalablement conviés.

Chaque défilé doit avoir au moins un événement marquant, qui peut être un lieu emblématique, un invité exceptionnel, une tenue hors du commun, ou autre, sous forme d'une petite description.

Chaque défilé a un nombre limité de places pour invitations.

Le jour du défilé, la tension monte. Chaque présentation est minutieusement orchestrée, des horaires précis aux lieux soigneusement choisis pour refléter l'essence des créations. À Paris, par exemple, le défilé de Dior se déroule dans un lieu emblématique, qui accueille des invités prestigieux. L'ambiance, les lumières, et les performances doivent être parfaitement alignées pour offrir une expérience unique aux spectateurs.

## Mannequin :

Les mannequins jouent un rôle clé dans la présentation des collections lors des défilés de mode. Ils sont les porteurs des créations, mettant en valeur les tenues et transmettant l'essence des collections sur le podium. Les mannequins apportent une dimension vivante aux créations, permettant au public de voir comment les vêtements se comportent lorsqu'ils sont portés, et contribuant à l'ambiance générale d'un défilé.

Un mannequin doit faire au minimum 1.70m de taille afin de garantir sa participation à un défilé.

Afin de garantir le bien-être et la non-surexploitation des mannequins, une limite de participation à 3 défilés par jour est mise en place.

Les mannequins sont directement liés aux tenues, car ce sont eux qui les portent pendant les défilés. Chaque mannequin est assigné à une ou plusieurs tenues lors d'un événement.

Les créateurs ou les directeurs artistiques des défilés choisissent les mannequins en fonction de leur morphologie, de leur allure, et de leur capacité à mettre en valeur les tenues.

Les mannequins sont indispensables pour la réalisation des défilés de mode. Chaque défilé nécessite la présence de plusieurs mannequins.

Les créateurs collaborent étroitement avec les mannequins pour s'assurer que leur vision artistique soit correctement interprétée sur le podium.

Les mannequins peuvent également être choisis par les maisons de mode pour représenter leur image et défiler avec leurs collections. Ils peuvent ainsi devenir les visages emblématiques de certaines marques.

Il est nécessaire que les mannequins soient majeurs pour participer et que leurs identités soient confirmées.

Un mannequin peut porter les tenues d'une collection ou plusieurs de chaque maison de mode.



## Maison de mode :

Les maisons de mode sont des acteurs majeurs dans le monde de la mode, jouant un rôle central dans la création, la promotion, et la diffusion des tendances. Elles représentent bien plus qu'un simple label : ce sont des entités qui possèdent une identité propre et une histoire, souvent associées à un style ou à une philosophie particulière. Les maisons de mode servent de plateforme pour les créateurs, en les soutenant dans la conception de collections et en assurant la promotion de leurs créations lors des défilés et autres événements.

Les maisons de mode collaborent étroitement avec les créateurs. Les créateurs sont employés par une maison de mode.

Les maisons de mode sont responsables de la présentation des collections de mode au public, lors de la Fashion Week.

Chaque collection est associée à une maison de mode, qui la soutient depuis sa conception jusqu'à sa présentation finale. Cela inclut la production, la sélection des tenues, et la mise en scène de la collection lors des défilés.

Les maisons de mode sont souvent les organisatrices des défilés, où elles présentent les collections de leurs créateurs à un public de professionnels et de passionnés de mode.

Elles se chargent de la logistique de l'événement, choisissant le lieu, l'organisation des tenues, et les mannequins qui défileront sur le podium.

La maison de mode a comme responsabilité la création d'un événement marquant et d'attirer l'attention des médias et des acheteurs, en tenant un thème important pour l'art de la mode, moderne ainsi que fidèle à son identité.

Les maisons de mode sélectionnent les mannequins qui défileront avec leurs tenues lors des événements. Elles choisissent des mannequins capables de mettre en valeur les créations et de refléter l'image de la maison.

## Invité :

Les invités de la Fashion Week représentent un public soigneusement sélectionné, souvent composé de personnalités influentes de l'industrie, ou des réseaux sociaux, de journalistes, de célébrités, de critiques de mode, et de partenaires commerciaux. Leur présence est essentielle pour assurer la visibilité et le succès médiatique des collections présentées, car leurs avis et leur influence peuvent contribuer à façonner les tendances et à attirer l'attention sur les créations des maisons de mode.

Les invités reçoivent des invitations pour assister à des défilés précis, selon le type d'événement. Avec une place ou un rang spécifique, tout dépend de l'importance de chaque invité.

Les maisons de mode sélectionnent soigneusement les invités à qui elles envoient des invitations, cherchant à maximiser l'impact médiatique et la réputation de leur marque.

## Journalistes :

Les journalistes mode représentent une branche du journalisme qui couvre les activités liées à la mode, les tendances et les nouveautés du domaine. Ils jouent un rôle fondamental dans la couverture médiatique de la Fashion Week et des autres événements de mode. Ce sont les narrateurs et les critiques qui relayent les créations des créateurs et les tendances des maisons de mode auprès du grand public. À travers leurs articles, reportages, et interviews, ils contribuent à façonner l'image de la mode dans les médias, influençant ainsi la perception des collections et le succès des défilés.

Les journalistes sont souvent invités à assister aux défilés de mode pour couvrir les événements et rapporter les dernières tendances au public.

Les maisons de mode considèrent les journalistes comme des partenaires stratégiques pour la promotion de leurs créations. Elles leur envoient des invitations pour leurs défilés et événements, cherchant à bénéficier d'une couverture médiatique favorable.

Les créateurs comptent sur les journalistes pour faire connaître leur travail et valoriser leur vision artistique. Ils peuvent accorder des interviews aux journalistes pour expliquer l'inspiration derrière leurs collections et les messages qu'ils souhaitent véhiculer.

Les journalistes peuvent interagir avec les autres invités lors des défilés, recueillant leurs impressions sur les collections et prenant le pouls de l'ambiance générale des événements. + les mannequins.

## Sponsors :

Les sponsors jouent un rôle fondamental dans le financement et le succès de la Fashion Week. Leur présence est indispensable pour la viabilité de l'événement.

Il est crucial de pouvoir enregistrer toutes les informations liées aux partenariats et aux contrats, afin de gérer efficacement ces relations.

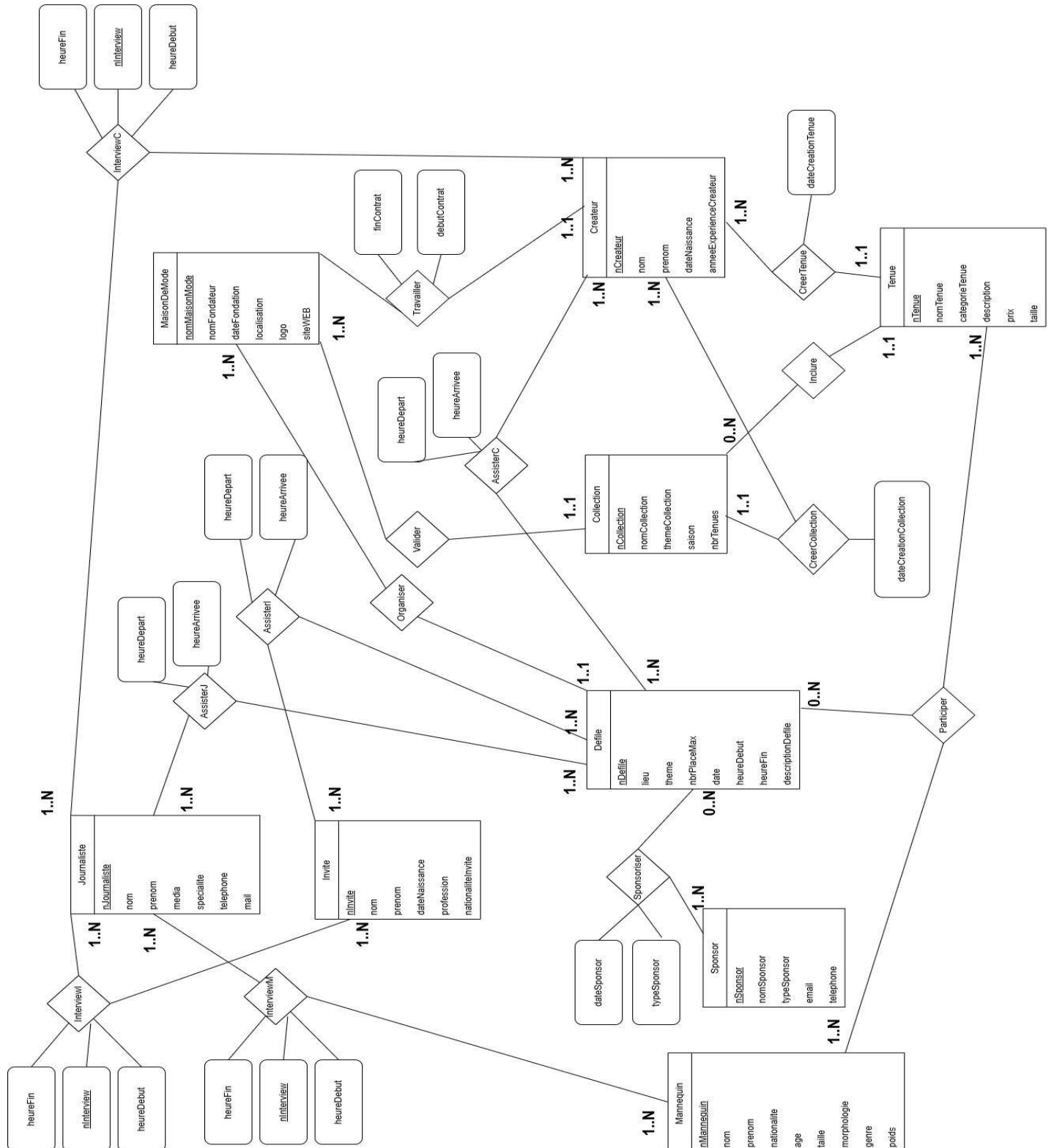
Une bonne gestion permet de garantir que les sponsors sont représentés de manière adéquate lors des événements, et que leurs contributions sont mises en avant.

Les données financières associées aux sponsors doivent être suivies avec soin pour s'assurer que chaque engagement contractuel est respecté.

Les sponsors ne se contentent pas de soutenir financièrement l'événement, ils participent aussi au maintien d'une relation durable et mutuellement bénéfique avec les organisateurs.

# Modélisation

## Schéma Entité/Association



# Contraintes d'intégrité

## Créateurs :

- Chaque créateur doit être associé à une maison de mode.
- Un créateur ne peut pas participer à un défilé s'il n'a pas au moins une collection.
- Un créateur doit avoir minimum 18 ans.

## Tenues :

- Une tenue ne peut pas être présentée dans plus d'un défilé lors d'une saison.
- Chaque tenue doit appartenir à une seule collection.
- Les tenues doivent être à la taille d'au moins un mannequin.
- Les tenues doivent être créées par le créateur qui travaille pour la maison de mode qui organise le défilé.

## Collections :

- Chaque collection doit avoir un thème, un titre et une saison mode.
- Une collection doit contenir au moins 10 tenues pour être considérée comme complète et être présentée lors d'un défilé.
- Une collection ne peut pas être associée à plus d'un défilé dans la même saison.
- Chaque collection doit être associée à un créateur et une maison de mode.

## Défilés :

- Un défilé ne peut pas contenir moins de 10 tenues et peut avoir autant de tenues possibles.
- Un défilé ne peut être associé qu'à une seule maison de mode par date, heure et lieu.
- Les défilés ne peuvent commencer qu'à partir de 14h.
- Un défilé doit avoir au moins 10 mannequins inscrits pour être validé.
- Une seule collection peut être présentée lors d'un défilé.
- Un défilé est assisté par la preuve d'une invitation.

## Mannequins :

- Un mannequin ne peut pas défiler plus de 3 fois par jour.
- Chaque mannequin doit être associé à une tenue lorsqu'il participe à un défilé.
- Un mannequin ne peut pas être affecté à plus de 5 tenues au cours d'un même défilé.
- Un mannequin ne peut pas avoir moins de 1.70 cm de taille.
- Un mannequin doit avoir au minimum 16 ans pour participer à un défilé.

### **Maisons de mode :**

- Une maison de mode ne peut pas avoir plus d'une collection présentée dans une même saison.
- Une maison de mode ne peut avoir plus d'un créateur principal sous contrat.

### **Invités :**

- Le nombre d'invités ne peut pas dépasser le nombre de places disponibles dans le défilé.

### **Journalistes :**

- Un journaliste ne peut pas assister à plus de cinq défilés par jour pendant la Fashion Week.

### **Sponsors :**

- Un sponsor doit sponsoriser au maximum 3 défilés.

# Droits et vues – confidentialité des données

## Administration de la Base de Données

- **Rôle** : Gestion et maintenance de la base de données.
- **Droits** :
  - Droits de création, lecture, modification, et suppression sur toutes les tables (CRUD).
  - Gestion des utilisateurs et des droits d'accès.
  - Création et gestion des vues.
- **Vues** :
  - Accès total à toutes les données pour assurer le bon fonctionnement de la base de données et la correction des anomalies.

## Créateurs

- **Rôle** : Concevoir des collections et superviser leur présentation lors des défilés.
- **Droits** :
  - Lecture et modification sur les informations les concernant, notamment leurs collections et leurs tenues.
  - Lecture sur les défilés auxquels leurs collections sont présentées.
  - Création et suppression de nouvelles collections et tenues.
- **Vues** :
  - Affiche les collections du créateur avec les tenues associées.
  - Affiche les défilés où les collections du créateur sont présentées.

## Maisons de Mode

- **Rôle** : Supporter les créateurs, organiser les défilés et gérer la production des collections.
- **Droits** :
  - Lecture et modification sur les informations concernant les collections et les créateurs affiliés à leur maison de mode.
  - Lecture sur les défilés organisés par la maison de mode.
  - Lecture des mannequins assignés aux défilés de la maison.
  - Création de nouveaux défilés.
  - Création des collections et tenues et les modifier.
- **Vues** :
  - Affiche les collections produites par la maison de mode.
  - Affiche les informations des défilés organisés par la maison.
  - Liste des mannequins participant aux défilés de la maison.

## Mannequins

- **Rôle : Participer aux défilés et porter les tenues des collections.**
- **Droits :**
  - Lecture sur leurs propres informations personnelles, les tenues qu'ils doivent porter, et les défilés auxquels ils participent.
- **Vues :**
  - Affiche les défilés où le mannequin est programmé.
  - Liste les tenues attribuées au mannequin pour chaque défilé.

## Journalistes

- **Rôle : Couvrir les événements de la Fashion Week, rédiger des critiques et analyser les tendances.**
- **Droits :**
  - Lecture sur les défilés, les collections présentées et les créateurs pour préparer des articles et des analyses.
  - Création et modification de leurs propres articles concernant les collections et les défilés couverts.
  - Lecture sur les informations générales des maisons de mode pour les besoins de couverture médiatique.
- **Vues :**
  - Affiche les collections qui seront présentées lors des défilés pour lesquels le journaliste est accrédité.
  - Affiche les informations publiques des maisons de mode pour aider le journaliste dans ses recherches.

## Invités

- **Rôle : Assister aux défilés et découvrir les nouvelles collections.**
- **Droits :**
  - Lecture sur les défilés auxquels ils peuvent assister.
  - Lecture sur les informations générales sur les collections présentées lors des défilés.
- **Vues :**
  - Affiche les défilés auxquels l'invité est autorisé à assister.
  - Liste les collections présentées lors des défilés auxquels ils sont invités.

# Requêtes

## Requêtes sur les Créateurs

1. Donner le nom de tous les créateurs participant à la Fashion Week se déroulant du jj/mm/aaaa à XX par exemple : du 22-déc-2024 à Paris.
2. Lister toutes les collections d'un créateur avec le nombre de tenues par collection.
3. La moyenne des prix des tenues par Créateur.
4. Le créateur avec le plus de tenues créées.

## Requêtes sur les Collections

1. Quelles sont les collections présentées lors de la saison 'automne-hiver 2024' ?
2. Quelles sont les collections créées par le créateur "Y" pour la maison de mode "Z" ?

## Requêtes sur les Tenues

1. Quelle est la moyenne des prix des tenues de collection 'X' qui ont été présentées par la maison de mode "Z" cette année ?
2. Quelles sont les tenues qui coûtent plus cher que la moyenne des prix de leur collection ?
3. Quelle est la tenue la moins chère et la plus chère de chaque maison de mode ?

## Requêtes sur les Défilés

1. Pour chaque défilé, combien d'invités ont été présents ?
2. Quel est le nombre total de défilés organisés par la maison de mode "X" cette année ?

## Requêtes sur les Mannequins

1. Quels mannequins ont participé à plus de trois défilés lors de la saison 'XX' 2024 ?
2. Quels mannequins ont porté des tenues de la collection "X" lors du défilé "yyyy/mm/dd" ?
3. Donner le nom et le prénom des mannequins ont défilé aux deux Fashion Week de Paris 2021.
4. Trouver les mannequins qui n'ont participé à aucun défilé cette année.

## Requêtes sur les Maisons de Mode

1. Quelles maisons de mode ont organisé un défilé pour chaque saison cette année ?
2. Quelles maisons de mode ont organisé le plus grand nombre de défilés cette année ?

## Requêtes sur les Invités

1. Quels sont les invités ayant assisté à plus de cinq défilés cette saison ?
2. Les invités qui n'ont assisté à aucun défilé cette année

## Requêtes sur les Journalistes

1. Quels sont les journalistes qui n'ont couvert aucun défilé cette année ?
2. Combien de défilés chaque journaliste a-t-il couvert cette saison ?
3. Quels sont les journalistes qui ont couvert tous les défilés d'une maison de mode spécifique ?
4. Nombre total d'interviews réalisées par chaque journaliste par type d'interview ?
5. Quels journalistes n'ont réalisé aucune interview cette année ?



## Requêtes sur les sponsors

1. Liste des sponsors et le nombre de défilés sponsorisés.
2. Sponsors ayant sponsorisé des défilés pour une maison de mode spécifique.

## Schéma E/A en relationnel

Les clés primaires sont marquées en vert, tandis que les clés étrangères sont identifiées en rouge.  
modif en cours :

- Createur (nCreateur, prenom, nom, dateNaissance, nationalite, anneeExperienceCreateur, #nomMaisonMonde)
- Tenue (nTenue, taille, prix, nomTenue, Description, categorieTenue, #nCollection, #nCreateur)
- Collection (nCollection, nomCollection, themeCollection, saison, nbrTenues, #nCreateur, #nomMaisonMonde)
- Mannequin (nMannequin, nom, prenom, age, taille, poids, nationalite, genre, morphologie)
- Defile (nDefile, lieu, heureDebut, heureFin, date, theme, descriptionDefile, nbrPlaceMax, #nomMaisonMode)
- MaisonMode(nomMaisonMode, nomFondateur, dateFondation, Localisation, logo, siteWEB)
- Sponsor (nSponsor, nomSponsor, typeSponsor, email, telephone)
- Invite (nInvite, nom, prenom, dateNaissance, nationaliteInvite, profession)
- Journaliste (nJournaliste, nom, prenom, media, specialite, telephone, mail)
- AssisterC (heureDepart, heureArrivee, #nDefile, #nCreateur)
- AssisterJ (heureDepart, heureArrivee, #nDefile, #nJournaliste)
- AssisterI (heureDepart, heureArrivee, #nDefile, #nInvite)
- InterviewI (nInterview, heureDebut, heureFin, #nJournaliste, #nInvite)
- InterviewM (nInterview, heureDebut, heureFin, #nJournaliste, #nMannequin)
- InterviewC (nInterview, heureDebut, heureFin, #nJournaliste, #nCreateur)
- Sponsoriser (dateSponsor, typeSponsor, #nSponsor, #nDefile)
- Participer (#nMannequin, #nDefile, #nTenue)

## Conclusion

La Fashion Week est une véritable démonstration d'organisation et de créativité. Chaque créateur, mannequin, tenue, invité et partenaire joue un rôle crucial dans le succès de cet événement. Derrière les défilés se cache une équipe dédiée, assurant que tout se déroule parfaitement, du choix des lieux aux invitations des VIP. Chaque édition de la Fashion Week repousse les limites, influençant la mode à une échelle mondiale, et c'est grâce à une organisation sans faille que cet événement reste l'un des plus importants dans le monde de la mode.

# Implémentation de BD sous Oracle

**S:** Slimane BOUAROUA  
**L:** Louiza MAKHOUKHE  
**A:** Abdelhak MERABTENE

Ces lettres seront utilisées pour identifier les tâches réalisées par chaque membre dans le projet.

## A/ Création du schéma de la base de données

### Table de Maison de mode

```
CREATE TABLE MaisonMode (
  nomMaisonMode VARCHAR(50) PRIMARY KEY,
  nomFondateur VARCHAR(50),
  dateFondation DATE,
  localisation VARCHAR(100),
  logo VARCHAR(100),
  siteWEB VARCHAR(100));
```

### Table de Créateur

```
• CREATE TABLE Createur (
•   nCreateur INT PRIMARY KEY,
•   prenom VARCHAR(50),
•   nom VARCHAR(50),
•   dateNaissance DATE,
•   nationalite VARCHAR(50),
•   anneeExperienceCreateur INT,
•   nomMaisonMode VARCHAR(50) NOT NULL,
•   FOREIGN KEY (nomMaisonMode) REFERENCES MaisonMode(nomMaisonMode) ON DELETE CASCADE,
•   CONSTRAINT ch_c CHECK (dateNaissance <= TO_DATE('2006-10-30', 'YYYY-MM-DD')));
```

## Table de Mannequin

```
CREATE TABLE Mannequin (
  nMannequin INT PRIMARY KEY,
  nom VARCHAR(50),
  prenom VARCHAR(50),
  age INT CHECK (age >= 16),
  taille NUMBER(5, 2) CHECK (taille >= 170),
  poids NUMBER(5, 2),
  nationalite VARCHAR(50),
  genre VARCHAR(10),
  morphologie VARCHAR(50));
```

## Table de Invité

```
CREATE TABLE Invite (
  nInvite INT PRIMARY KEY,
  nom VARCHAR(50),
  prenom VARCHAR(50),
  dateNaissance DATE,
  nationaliteInvite VARCHAR(50),
  profession VARCHAR(50));
```

## Table de Journaliste

```
CREATE TABLE Journaliste (
  nJournaliste INT PRIMARY KEY,
  nom VARCHAR(50),
  prenom VARCHAR(50),
  media VARCHAR(100),
  specialite VARCHAR(50),
  telephone VARCHAR(20),
  mail VARCHAR(100));
```

## Table de Sponsor

```
CREATE TABLE Sponsor (
  nSponsor INT PRIMARY KEY,
  nomSponsor VARCHAR(100),
  typeSponsor VARCHAR(50),
  email VARCHAR(100),
  telephone VARCHAR(20));
```

## Table de Défile

```

• CREATE TABLE Defile (
•     nDefile INT PRIMARY KEY,
•     lieu VARCHAR(100),
•     heureDebut TIMESTAMP,
•     heureFin TIMESTAMP,
•     dateDefile DATE,
•     theme VARCHAR(50),
•     descriptionDefile VARCHAR(255),
•     nbrPlaceMax INT,
•     nomMaisonMode VARCHAR(50) NOT NULL,
•     FOREIGN KEY (nomMaisonMode) REFERENCES MaisonMode(nomMaisonMode) ON DELETE CASCADE,
•     CONSTRAINT ch_d CHECK (EXTRACT(HOUR FROM heureDebut) >= 14));
  
```

## Table de Collection

```

CREATE TABLE Collection (
    nCollection INT PRIMARY KEY,
    nomCollection VARCHAR(100) NOT NULL,
    themeCollection VARCHAR(50) NOT NULL,
    saison VARCHAR(20) NOT NULL,
    nbrTenues INT DEFAULT 0,
    nCreateur INT NOT NULL,
    nomMaisonMode VARCHAR(50) NOT NULL,
    FOREIGN KEY (nCreateur) REFERENCES Createur(nCreateur) ON DELETE SET NULL,
    FOREIGN KEY (nomMaisonMode) REFERENCES MaisonMode(nomMaisonMode) ON DELETE CASCADE);
  
```

## Table de Tenue

```

CREATE TABLE Tenue (
    nTenue INT PRIMARY KEY,
    taille NUMBER(5, 2),
    prix NUMBER(10, 2),
    nomTenue VARCHAR(100),
    description VARCHAR(255),
    categorieTenue VARCHAR(50),
    nCollection INT NOT NULL,
    nCreateur INT NOT NULL,
    FOREIGN KEY (nCollection) REFERENCES Collection(nCollection) ON DELETE SET NULL,
    FOREIGN KEY (nCreateur) REFERENCES Createur(nCreateur) ON DELETE SET NULL);
  
```

## Table de Assister pour Invité

```

CREATE TABLE AssisterI (
    heureDepart TIMESTAMP,
    heureArrivee TIMESTAMP,
    nDefile INT,
    nInvite INT,
    FOREIGN KEY (nDefile) REFERENCES Defile(nDefile) ON DELETE CASCADE,
    FOREIGN KEY (nInvite) REFERENCES Invite(nInvite) ON DELETE SET NULL);
  
```

## Table de Assister pour Journaliste

```
CREATE TABLE AssisterJ (
    heureDepart TIMESTAMP,
    heureArrivee TIMESTAMP,
    nJournaliste INT,
    nDefile INT,
    FOREIGN KEY (nJournaliste) REFERENCES Journaliste(nJournaliste) ON DELETE SET NULL,
    FOREIGN KEY (nDefile) REFERENCES Defile(nDefile) ON DELETE CASCADE);
```

## Table de Assister pour Créateur

```
CREATE TABLE AssisterC (
    heureDepart TIMESTAMP,
    heureArrivee TIMESTAMP,
    nDefile INT,
    nCreateur INT,
    FOREIGN KEY (nDefile) REFERENCES Defile(nDefile) ON DELETE CASCADE,
    FOREIGN KEY (nCreateur) REFERENCES Createur(nCreateur) ON DELETE SET NULL);
```

## Table de Interview avec Mannequin

```
CREATE TABLE InterviewM (
    nInterview INT PRIMARY KEY,
    heureDebut TIMESTAMP,
    heureFin TIMESTAMP,
    nJournaliste INT,
    nMannequin INT,
    FOREIGN KEY (nJournaliste) REFERENCES Journaliste(nJournaliste) ON DELETE CASCADE,
    FOREIGN KEY (nMannequin) REFERENCES Mannequin(nMannequin) ON DELETE SET NULL
);
```

## Table de Interview avec Créateur

```
CREATE TABLE InterviewC (
    nInterview INT PRIMARY KEY,
    heureDebut TIMESTAMP,
    heureFin TIMESTAMP,
    nJournaliste INT,
    nCreateur INT,
    FOREIGN KEY (nJournaliste) REFERENCES Journaliste(nJournaliste) ON DELETE CASCADE,
    FOREIGN KEY (nCreateur) REFERENCES Createur(nCreateur) ON DELETE SET NULL);
```

## Table de Interview avec Invite

```
CREATE TABLE InterviewI (
  nInterview INT PRIMARY KEY,
  heureDebut TIMESTAMP,
  heureFin TIMESTAMP,
  nJournaliste INT,
  nInvite INT,
  FOREIGN KEY (nJournaliste) REFERENCES Journaliste(nJournaliste) ON DELETE CASCADE,
  FOREIGN KEY (nInvite) REFERENCES Invite(nInvite) ON DELETE SET NULL);
```

## Table de Sponsoriser

```
CREATE TABLE Sponsoriser (
  dateSponsor DATE,
  typeSponsor VARCHAR(50),
  nSponsor INT,
  nDefile INT,
  FOREIGN KEY (nSponsor) REFERENCES Sponsor(nSponsor) ON DELETE CASCADE,
  FOREIGN KEY (nDefile) REFERENCES Defile(nDefile) ON DELETE CASCADE);
```

## Table de Participer

```
• CREATE TABLE Participer (
•   nMannequin INT not null,
•   nDefile INT not null,
•   nTenue INT NOT NULL,
•   FOREIGN KEY (nMannequin) REFERENCES Mannequin(nMannequin) ON DELETE CASCADE,
•   FOREIGN KEY (nDefile) REFERENCES Defile(nDefile) ON DELETE CASCADE,
•   FOREIGN KEY (nTenue) REFERENCES Tenue(nTenue) ON DELETE CASCADE);
```



## B/ Jeu de données

Nous avons construit le jeu de données en respectant soigneusement les relations définies dans le schéma de la base de données. Les données ont été générées en utilisant plusieurs outils afin de varier les sources et garantir une diversité dans les informations. Une partie des données a été générée avec ChatGPT, une autre avec Copilot et Gemini, tandis qu'une partie a été saisie manuellement (notamment pour respecter les triggers mais surtout les clés étrangères). Toutes les données ont ensuite été corrigées et vérifiées pour assurer leur cohérence avec les clés primaires et étrangères ainsi que leur pertinence par rapport au schéma et aux requêtes de la base de données.

### A - Table de Maison de mode (36 tuples)

#### A partir de SQL

```
INSERT INTO MaisonMode (nomMaisonMode, nomFondateur, dateFondation, localisation, logo,
siteWEB) VALUES ('Off-White', 'Virgil Abloh', TO_DATE('2012-01-01', 'YYYY-MM-DD'), 'Milan,
Italy', 'offwhite_logo.png', 'https://www.off--white.com');
```

#### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE MaisonMode

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY '"'

(nomMaisonMode, nomFondateur, dateFondation "TO\_DATE(:dateFondation, 'YYYY-MM-DD')",  
localisation, logo, siteWEB)

BEGINDATA

"UVSQ", "Beatrice Finance", "2012-12-12", "Versailles, France", "uvsq.png", "https://www.uvsq.com"

"ESGI", "Frederic Sanaenes", "2023-09-16", "Nation, France", "esgi.png", "https://www.esgi.com"

"Chanel", "Gabrielle Chanel", "1910-01-01", "Paris, France", "chanel\_logo.png", "https://www.chanel.com"

"Dior", "Christian Dior", "1946-10-08", "Paris, France", "dior\_logo.png", "https://www.dior.com"

## A - Table de Créateur (33 tuples)

### A partir de SQL

```
INSERT INTO Createur (nCreateur, prenom, nom, dateNaissance, nationalite,
anneeExperienceCreateur, nomMaisonMode) VALUES (33, 'Elena', 'Rodriguez', TO_DATE('1990-09-12', 'YYYY-MM-DD'), 'Espagnole', 5, 'Valentino');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Createur

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY '"'

(nCreateur, prenom, nom, dateNaissance "TO\_DATE(:dateNaissance, 'YYYY-MM-DD')", nationalite, anneeExperienceCreateur, nomMaisonMode)

BEGINDATA

1. "Alice", "Dupont", "1985-06-15", "Italienne", 10, "Chanel"
2. "Benjamin", "Martin", "1990-03-22", "Francais", 8, "Dior"
3. "Clara", "Leroy", "1988-11-05", "Anglaise", 12, "Louis Vuitton"
4. "David", "Bernard", "1992-01-30", "Australien", 6, "Gucci"
5. "Emma", "Rousseau", "1980-04-10", "Russe", 15, "Prada"

## A - Table de Mannequin (33 tuples)

### A partir de SQL

```
INSERT INTO Mannequin (nMannequin, nom, prenom, age, taille, poids, nationalite, genre, morphologie) VALUES (33, 'Khan', 'Aaliyah', 27, 174.00, 56.00, 'Indienne', 'Feminin', 'Sablier');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Mannequin

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY '"'

(nMannequin, nom, prenom, age, taille, poids, nationalite, genre, morphologie)

BEGINDATA

1. "Hadid", "Gigi", 29, 178.00, 54.00, "Americaine", "Feminin", "Sablier"
2. "Jenner", "Kendall", 29, 179.00, 58.00, "Americaine", "Feminin", "Sablier"
3. "Hadid", "Bella", 28, 175.00, 55.00, "Americaine", "Feminin", "Sablier"
4. "Lima", "Adriana", 42, 178.00, 60.00, "Bresilienne", "Feminin", "Sablier"
5. "Banks", "Tyra", 50, 180.00, 61.00, "Americaine", "Feminin", "Sablier"

## A - Table de Invité (33 tuples)

### A partir de SQL

```
INSERT INTO Invite (nInvite, nom, prenom, dateNaissance, nationaliteInvite, profession)
VALUES (33, 'Durand', 'Elise', TO_DATE('1992-03-15', 'YYYY-MM-DD'), 'Francaise', 'Peintre');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Invite

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY '"'

(nInvite, nom, prenom, dateNaissance "TO\_DATE(:dateNaissance, 'YYYY-MM-DD')", nationaliteInvite, profession)

BEGINDATA

1. "Dupont", "Alice", "1990-01-15", "Francaise", "Styliste"
2. "Martin", "Julien", "1985-05-20", "Francais", "Designer"
3. "Leroy", "Camille", "1992-07-30", "Francaise", "Journaliste"
4. "Moreau", "Paul", "1988-02-12", "Francais", "Photographe"
5. "Garnier", "Marie", "1987-04-22", "Francaise", "Modele"

## A - Table de Journaliste (32 tuples)

### A partir de SQL

```
INSERT INTO Journaliste (nJournaliste, nom, prenom, media, specialite, telephone, mail)
VALUES (32, 'Klein', 'Marie', 'France Info', 'Sante', '0987654321',
'marie.klein@example.com');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Journaliste

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY '"'

(nJournaliste, nom, prenom, media, specialite, telephone, mail)

BEGINDATA

"Dupont", "Alice", "Le Monde", "Mode", "0123456789", "alice.dupont@example.com"

"Martin", "Julien", "France 24", "Culture", "0234567890", "julien.martin@example.com"

"Leroy", "Camille", "Elle", "Societe", "0345678901", "camille.leroy@example.com"

"Moreau", "Paul", "Le Figaro", "Politique", "0456789012", "paul.moreau@example.com"

"Garnier", "Marie", "Vogue", "Mode", "0567890123", "marie.garnier@example.com"

## A - Table de Sponsor (33 tuples)

### A partir de SQL

```
INSERT INTO Sponsor (nSponsor, nomSponsor, typeSponsor, email, telephone) VALUES (33,  
'ModeVision', 'Mode', 'contact@modevision.com', '0987654321');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Sponsor

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY '"'

(nSponsor, nomSponsor, typeSponsor, email, telephone)

BEGINDATA

1. "LuxeCorp", "Mode", "contact@luxecorp.com", "0123456789"
2. "FashionCo", "Beaute", "info@fashionco.com", "0234567890"
3. "StyleGroup", "Accessoires", "hello@stylegroup.com", "0345678901"
4. "ChicBrand", "Vetements", "support@chicbrand.com", "0456789012"
5. "EliteFashion", "Mode", "contact@elitefashion.com", "0567890123"

## L- Table de Défile (31 tuples)

### A partir de SQL

```
INSERT INTO Defile ( nDefile, lieu, heureDebut, heureFin, dateDefile, theme,
descriptionDefile, nbrPlaceMax, nomMaisonMode) VALUES (
    13,
    'Palace of Versailles, France',
    TO_TIMESTAMP('2025-01-15 14:45:00', 'YYYY-MM-DD HH24:MI:SS'),
    TO_TIMESTAMP('2025-01-15 18:15:00', 'YYYY-MM-DD HH24:MI:SS'),
    TO_DATE('2025-01-15', 'YYYY-MM-DD'),
    'Rêve Baroque',
    'Collection inspirée du baroque français.',
    250,
    'Alberta Ferretti');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Defile

FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'

(nDefile, lieu, heureDebut TIMESTAMP "YYYY-MM-DD HH24:MI:SS", heureFin TIMESTAMP "YYYY-MM-DD HH24:MI:SS", dateDefile DATE "YYYY-MM-DD", theme, descriptionDefile, nbrPlaceMax, nomMaisonMode)

BEGINDATA

14,"Palace of Versailles, France" , "2025-01-15 14:45:00" , "2025-01-15 18:15:00" , "2025-01-15" , "Rêve Baroque" , "Collection inspirée du baroque français" , 250 , "Alberta Ferretti"

15,"Golden Gate Park, San Francisco" , "2025-02-10 16:20:00" , "2025-02-10 15:30:00" , "2025-02-10" , "Vents de Liberté" , "Exploration des styles audacieux et modernes" , 180 , "Alexander McQueen"

16,"Sydney Opera House, Sydney" , "2025-03-05 21:00:00" , "2025-03-05 23:00:00" , "2025-03-05" , "Océans Infinis" , "Collection inspirée des fonds marins et des vagues" , 300 , "Ann Demeulemeester"

17,"Taj Mahal Gardens, India" , "2025-03-20 18:30:00" , "2025-03-20 20:30:00" , "2025-03-20" , "Éclat et Tradition" , "Mariage entre culture indienne et design moderne" , 400 , "Armani"

## S- Table de Collection (31 tuples)

### A partir de SQL

```
INSERT INTO Collection (nCollection, nomCollection, themeCollection, saison, nbrTenues,
nCreateur, nomMaisonMode) VALUES (34, 'Reflets Cristallins', 'Elegance Givree', 'Hiver', 0,
15, 'Versace');
```

### A partir de l'utilitaire SQL\*LOAD

LOAD DATA

INFILE \*

APPEND

INTO TABLE Collection

FIELDS TERMINATED BY "," OPTIONALLY ENCLOSED BY ""

(nCollection, nomCollection, themeCollection, saison, nbrTenues, nCreateur, nomMaisonMode)

BEGINDATA

1. "Echos Sauvages", "Evasion et Nature", "Printemps/Ete", 0, 1, "Lacoste"
2. "Heritage Intemporel", "Elegance et Intemporalite", "Automne/Hiver", 0, 2, "Chanel"
3. "Metropole Brulante", "Moderne et Urbain", "Automne/Hiver", 0, 3, "Dior"
4. "Haute Couture Marine", "Elegance et Mer", "Printemps/Ete", 0, 1, "Lacoste"



## A - Table de Tenue (31 tuples)

### A partir de SQL

```
INSERT INTO Tenue (nTenue, taille, prix, nomTenue, description, categorieTenue, nCollection, nCreateur)
VALUES (1, 178.00, 250.00, 'Robe Printanière', 'Une robe légère pour les journées ensoleillées.', 'Robe', 1, 1);

INSERT INTO Tenue (nTenue, taille, prix, nomTenue, description, categorieTenue, nCollection, nCreateur)
VALUES (2, 179.00, 300.00, 'Costume Intemporel', 'Costume élégant pour toutes les occasions.', 'Costume', 1, 1);

INSERT INTO Tenue (nTenue, taille, prix, nomTenue, description, categorieTenue, nCollection, nCreateur)
VALUES (3, 175.00, 150.00, 'T-shirt Urbain', 'T-shirt moderne et confortable.', 'T-shirt', 1, 1);

INSERT INTO Tenue (nTenue, taille, prix, nomTenue, description, categorieTenue, nCollection, nCreateur)
VALUES (4, 178.00, 450.00, 'Robe Astrale', 'Inspirée des constellations.', 'Robe', 1, 1);

INSERT INTO Tenue (nTenue, taille, prix, nomTenue, description, categorieTenue, nCollection, nCreateur)
VALUES (5, 180.00, 500.00, 'Manteau Vintage', 'Manteau classique avec une touche moderne.', 'Manteau', 1, 1);
```

## S- Table de Assister pour Invité (31 tuples)

### A partir de SQL

```
INSERT INTO AssisterI (heureDepart, heureArrivee, nDefile, nInvite)
VALUES (TO_TIMESTAMP('2024-12-10 14:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-12-10 16:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100, 100);

INSERT INTO AssisterI (heureDepart, heureArrivee, nDefile, nInvite)
VALUES (TO_TIMESTAMP('1987-07-15 15:50:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1987-07-15 18:10:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 32);

INSERT INTO AssisterI (heureDepart, heureArrivee, nDefile, nInvite)
VALUES (TO_TIMESTAMP('1990-09-10 17:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-09-10 19:15:00', 'YYYY-MM-DD HH24:MI:SS'), 3, 1);

INSERT INTO AssisterI (heureDepart, heureArrivee, nDefile, nInvite)
VALUES (TO_TIMESTAMP('1990-03-22 15:35:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-03-22 18:00:00', 'YYYY-MM-DD HH24:MI:SS'), 4, 2);
```

## S- Table de Assister pour Journaliste (30 tuples)

### A partir de SQL

```
INSERT INTO AssisterJ (heureDepart, heureArrivee, nDefile, nJournaliste) VALUES
(TO_TIMESTAMP('2024-12-10 14:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-12-10
16:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100, 100);
INSERT INTO AssisterJ (heureDepart, heureArrivee, nJournaliste, nDefile) VALUES
(TO_DATE('1987-07-15 15:50:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('1987-07-15 18:00:00',
'YYYY-MM-DD HH24:MI:SS'), 2, 2);
INSERT INTO AssisterJ (heureDepart, heureArrivee, nJournaliste, nDefile) VALUES
(TO_DATE('1990-09-10 17:10:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('1990-09-10 19:20:00',
'YYYY-MM-DD HH24:MI:SS'), 3, 3);
INSERT INTO AssisterJ (heureDepart, heureArrivee, nJournaliste, nDefile) VALUES
(TO_DATE('1990-03-22 15:30:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('1990-03-22 18:00:00',
'YYYY-MM-DD HH24:MI:SS'), 4, 4);
```

## S- Table de Assister pour Créateur (30 tuples)

### A partir de SQL

```
INSERT INTO AssisterC (heureDepart, heureArrivee, nDefile, nCreateur)
VALUES ( TO_TIMESTAMP('2024-12-10 14:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-
12-10 16:00:00', 'YYYY-MM-DD HH24:MI:SS'),100, 100);
INSERT INTO AssisterC (heureDepart, heureArrivee, nDefile, nCreateur)
VALUES (TO_TIMESTAMP('1987-07-15 16:10:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1987-07-
15 18:20:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 2);
INSERT INTO AssisterC (heureDepart, heureArrivee, nDefile, nCreateur)
VALUES (TO_TIMESTAMP('1990-09-10 16:40:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-09-
10 18:30:00', 'YYYY-MM-DD HH24:MI:SS'), 3, 31);
INSERT INTO AssisterC (heureDepart, heureArrivee, nDefile, nCreateur)
VALUES (TO_TIMESTAMP('1990-03-22 16:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-03-
22 18:00:00', 'YYYY-MM-DD HH24:MI:SS'), 4, 2);
```

## L- Table de Interview avec Créateur (30 tuples)

### A partir de SQL

```
insert into InterviewC (nInterview, heureDebut, heureFin, nJournaliste, nCreateur)
VALUES (1, TO_TIMESTAMP('1985-06-15 18:50:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1985-
06-15 19:30:00', 'YYYY-MM-DD HH24:MI:SS'), 1, 1);
insert into InterviewC (nInterview, heureDebut, heureFin, nJournaliste, nCreateur)
VALUES (2, TO_TIMESTAMP('1987-07-15 18:15:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1987-
07-15 19:00:00', 'YYYY-MM-DD HH24:MI:SS'), 1, 2);
INSERT INTO InterviewC (nInterview, heureDebut, heureFin, nJournaliste, nCreateur)
VALUES (3, TO_TIMESTAMP('1990-09-10 19:35:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-
09-10 20:15:00', 'YYYY-MM-DD HH24:MI:SS'), 3, 3);
INSERT INTO InterviewC (nInterview, heureDebut, heureFin, nJournaliste, nCreateur)
VALUES (4, TO_TIMESTAMP('1990-03-22 18:05:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-
03-22 19:20:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 4);
```

## L- Table de Interview avec Mannequin (30 tuples)

### A partir de SQL

```

INSERT INTO InterviewM (nInterview, heureDebut, heureFin, nJournaliste, nMannequin)
VALUES (1, TO_TIMESTAMP('1985-06-15 20:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1985-06-15 20:30:00', 'YYYY-MM-DD HH24:MI:SS'), 1, 2);

INSERT INTO InterviewM (nInterview, heureDebut, heureFin, nJournaliste, nMannequin)
VALUES (2, TO_TIMESTAMP('1987-07-15 19:15:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1987-07-15 19:45:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 3);

INSERT INTO InterviewM (nInterview, heureDebut, heureFin, nJournaliste, nMannequin)
VALUES (3, TO_TIMESTAMP('1990-09-10 21:35:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-09-10 22:15:00', 'YYYY-MM-DD HH24:MI:SS'), 3, 4);

INSERT INTO InterviewM (nInterview, heureDebut, heureFin, nJournaliste, nMannequin)
VALUES (4, TO_TIMESTAMP('1990-03-22 19:30:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-03-22 20:00:00', 'YYYY-MM-DD HH24:MI:SS'), 4, 4);

INSERT INTO InterviewM (nInterview, heureDebut, heureFin, nJournaliste, nMannequin)
VALUES (5, TO_TIMESTAMP('1993-01-10 19:10:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1993-01-10 19:50:00', 'YYYY-MM-DD HH24:MI:SS'), 5, 5);
  
```

## L- Table de Interview avec Invité (30 tuples)

### A partir de SQL

```

INSERT INTO InterviewI (nInterview, heureDebut, heureFin, nJournaliste, nInvite)
VALUES (1, TO_TIMESTAMP('1985-06-15 21:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1985-06-15 21:30:00', 'YYYY-MM-DD HH24:MI:SS'), 1, 2);

INSERT INTO InterviewI (nInterview, heureDebut, heureFin, nJournaliste, nInvite)
VALUES (2, TO_TIMESTAMP('1987-07-15 21:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1987-07-15 21:45:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 3);

INSERT INTO InterviewI (nInterview, heureDebut, heureFin, nJournaliste, nInvite)
VALUES (3, TO_TIMESTAMP('1990-09-10 22:35:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-09-10 23:30:00', 'YYYY-MM-DD HH24:MI:SS'), 3, 4);

INSERT INTO InterviewI (nInterview, heureDebut, heureFin, nJournaliste, nInvite)
VALUES (4, TO_TIMESTAMP('1990-03-22 19:30:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1990-03-22 20:00:00', 'YYYY-MM-DD HH24:MI:SS'), 4, 4);

INSERT INTO InterviewI (nInterview, heureDebut, heureFin, nJournaliste, nInvite)
VALUES (5, TO_TIMESTAMP('1993-01-10 19:18:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('1993-01-10 19:57:00', 'YYYY-MM-DD HH24:MI:SS'), 5, 5);
  
```

## S- Table de Sponsoriser (30 tuples)

### A partir de SQL

```
INSERT INTO Sponsoriser (dateSponsor, typeSponsor, nSponsor, nDefile) VALUES
(TO_DATE('1987-07-15', 'YYYY-MM-DD'), 'Beaute', 32, 2);
INSERT INTO Sponsoriser (dateSponsor, typeSponsor, nSponsor, nDefile) VALUES (TO_DATE('1990-
09-10', 'YYYY-MM-DD'), 'Mode', 1, 3);
INSERT INTO Sponsoriser (dateSponsor, typeSponsor, nSponsor, nDefile) VALUES (TO_DATE('1990-
03-22', 'YYYY-MM-DD'), 'Beaute', 2, 4);
INSERT INTO Sponsoriser (dateSponsor, typeSponsor, nSponsor, nDefile) VALUES (TO_DATE('1993-
01-10', 'YYYY-MM-DD'), 'Accessoires', 3, 5);
INSERT INTO Sponsoriser (dateSponsor, typeSponsor, nSponsor, nDefile) VALUES (TO_DATE('1992-
01-30', 'YYYY-MM-DD'), 'Vetements', 4, 6);
```

## A - Table de Participer (30 tuples)

### A partir de SQL

```
INSERT INTO Participer (nMannequin, nDefile, nTenue) VALUES (1, 17, 1);
INSERT INTO Participer (nMannequin, nDefile, nTenue) VALUES (2, 17, 2);
INSERT INTO Participer (nMannequin, nDefile, nTenue) VALUES (3, 17, 3);
INSERT INTO Participer (nMannequin, nDefile, nTenue) VALUES (4, 17, 4);
INSERT INTO Participer (nMannequin, nDefile, nTenue) VALUES (5, 17, 5);
INSERT INTO Participer (nMannequin, nDefile, nTenue) VALUES (6, 17, 6);
```

## C/ Manipulation des données

### Requêtes sur les Créateurs

**S**-Donner le nom de tous les créateurs participant à la Fashion Week se déroulant du jj/mm/aaaa à XX par exemple : du 22-déc-2024 à paris.

```
• select c.nom, c.prenom
• from participer p, tenue t, createur c, defile d
• where p.nTenue = t.nTenue
• and c.nCreateur = t.nCreateur
• and d.nDefile = p.nDefile
• and d.dateDefile = TO_DATE('2024-12-22', 'YYYY-MM-DD')
• and d.lieu = 'Paris';
```

1. **S**-Lister toutes les collections d'un créateur avec le nombre de tenues par collection.

```
• select c.nom , c.prenom, co.nomCollection, co.nbrTenues
• from collection co, createur c
• where co.ncreateur = c.ncreateur;
```

2. **S**-La moyenne des prix des tenues par Créateur.

```
• select c.nom, c.prenom, avg(t.prix)
• from tenue t , createur c
• where t.ncreateur = c.ncreateur
• group by c.nom, c.prenom;
```

3. **S**-Le créateur avec le plus de tenues créées.

```
• SELECT *
• FROM (
•     SELECT c.nom, c.prenom, SUM(co.nbrTenues) AS total_tenues
•     FROM collection co, createur c
•     where co.ncreateur = c.nCreateur
•     GROUP BY c.nom, c.prenom
•     ORDER BY total_tenues DESC
• )
• where rownum = 1;
```

## Requêtes sur les Collections

1. **L**-Quelles sont les collections présentées lors de la saison 'automne-hiver 2024' ?

```
• select co.nomCollection
• from participer p, collection co, tenue t, defile d
• where p.nTenue = t.nTenue
• and t.nCollection = co.nCollection
• and p.nDefile = d.nDefile
• and co.saison = 'Automne/Hiver'
• and TO_CHAR(d.dateDefile, 'YYYY') = '2024';
```

2. **L**-Quelles sont les collections créées par le créateur "Y" pour la maison de mode "Z" ?

```
• select co.ncollection, co.nomCollection
• from collection co, createur c, MaisonMode mm
• where co.ncreateur = c.ncreateur
• and c.nomMaisonMode = mm.nomMaisonMode
• and c.nom = 'Renaud'
• and c.prenom = 'Dorian'
• and mm.nomMaisonMode = 'Chanel';
```

## Requêtes sur les Tenues

1. A-Quelle est la moyenne des prix des tenues de collection 'Echo Sauvages' qui ont été présentées par la maison de mode "Lacoste" cette année ?

```

• select avg(t.prix) as moyenne_prix
• from collection co, participer p, tenue t, defile d
• where t.nTenue = p.nTenue
• and d.nDefile = p.nDefile
• and co.nCollection = t.nCollection
• and co.nomCollection = 'Echos Sauvages'
• and co.nomMaisonMode = 'Lacoste'
• and TO_CHAR(d.dateDefile, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY');

```

2. A-Quelles sont les tenues qui coûtent plus cher que la moyenne des prix de leur collection ?

```

• SELECT t.nTenue, t.nomTenue, t.prix, c.nomCollection
• FROM Tenue t
• JOIN Collection c ON t.nCollection = c.nCollection
• WHERE t.prix > (
•     SELECT AVG(t2.prix)
•     FROM Tenue t2
•     WHERE t2.nCollection = t.nCollection
• );

```

3. A-Quelle est la tenue la moins chère et la plus chère de chaque maison de mode ?

```

• SELECT co.nomMaisonMode,
•     MIN(t.prix) AS tenue_moins_chere,
•     MAX(t.prix) AS tenue_plus_chere
• FROM Tenue t, collection co
• where t.nCollection = co.nCollection
• GROUP BY co.nomMaisonMode;

```

## Requêtes sur les Défilés

1. L-Pour chaque défilé, combien d'invités ont été présents ?

```

• select a.nDefile, count(nInvite) as nombre_invites
• from AssisterI a
• group by a.nDefile;

```

2. L-Quel est le nombre total de défilés organisés par la maison de mode "X" cette année ?

```

• select count(d.nDefile) as nombre_total_defiles
• from defile d
• where TO_CHAR(d.dateDefile, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
• and d.nomMaisonMode = 'Chanel';

```

## Requêtes sur les Mannequins

1. S-Quels mannequins ont participé à plus de trois défilés lors de la saison 'XX' 2024 ?

```

• select m.nom, m.prenom, count(p.ndefile) as nombre_defiles
• from participer p, mannequin m, collection co, defile d, tenue t
• where p.nMannequin = m.nMannequin
• and p.ntenue = t.ntenue
• and t.ncollection = co.ncollection
• and p.ndefile = d.ndefile
• and co.saison = 'Printemps/Ete'
• and TO_CHAR(d.dateDefile, 'YYYY') = '2024'
• group by m.nom, m.prenom
• having count(p.ndefile) > 3;

```

2. S-Quels mannequins ont porté des tenues de la collection "X" lors du défilé "yyyy/mm/dd" ?

```

• select m.nom, m.prenom
• from mannequin m, participer p, tenue t, collection co, defile d
• where p.nMannequin = m.nMannequin
• and p.ntenue = t.ntenue
• and t.ncollection = co.ncollection
• and p.ndefile = d.ndefile
• and co.nomCollection = 'Echos Sauvages'
• and d.dateDefile = TO_DATE('1997-05-23', 'YYYY-MM-DD');

```

3. S-Donner le nom et le prénom des mannequins ont défilé aux deux Fashion Week de Paris 2021.

```

• select m.nom, m.prenom
• from participer p, mannequin m, defile d
• where p.nMannequin = m.nMannequin
• and p.ndefile = d.ndefile
• and d.lieu = 'Paris'
• and TO_CHAR(d.dateDefile, 'YYYY') = '2021'
• group by m.nom, m.prenom
• having count(distinct p.ndefile) = 2;

```

4. S-Trouver les mannequins qui n'ont participé à aucun défilé cette année.

```

• select m.nom, m.prenom
• from mannequin m
• where not exists( select *
•                  from participer p, defile d
•                  where p.ndefile = d.ndefile
•                  and TO_CHAR(d.dateDefile, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY'));

```

## Requêtes sur les Maisons de Mode

1. A-Quelles maisons de mode ont organisé un défilé pour chaque saison cette année ?

```

• select d.nomMaisonMode
• from Defile d, collection c
• where d.nomMaisonMode = c.nomMaisonMode
• and TO_CHAR(d.dateDefile, 'YYYY') = '2024'
• group by d.nomMaisonMode
• having COUNT(DISTINCT c.saison) = (SELECT COUNT(DISTINCT saison) FROM Collection);

```

2. A-Quelles maisons de mode ont organisé le plus grand nombre de défilés cette année ?

```

• select d.nomMaisonMode, count(d.ndefile) as nombre_defiles
• from defile d
• group by d.nomMaisonMode
• order by nombre_defiles desc;

```

## Requêtes sur les Invités

1. A-Quels sont les invités ayant assisté à plus de cinq défilés cette saison ?

```

• SELECT i.nom, i.prenom
• FROM invite i, assisteri a, defile d, collection c
• where i.nInvite = a.nInvite
• and a.nDefile = d.nDefile
• and d.nomMaisonMode = c.nomMaisonMode
• and TO_CHAR(d.dateDefile, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
• AND c.saison = 'Automne/Hiver'
• GROUP BY i.nom, i.prenom
• HAVING COUNT(a.nDefile) > 2

```

2. A-Les invités qui n'ont assisté à aucun défilé cette année.

```

• SELECT i.nom, i.prenom
• FROM Invite i
• WHERE NOT EXISTS (
•     SELECT *
•     FROM AssisterI ai
•     WHERE ai.nInvite = i.nInvite
•     AND TO_CHAR(ai.heureDepart, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
• );

```



## Requêtes sur les Journalistes

1. **S**-Quels sont les journalistes qui n'ont couvert aucun défilé cette année ?

```

• select j.nom, j.prenom
• from journaliste j
• where not exists(
•
•         select *
•         from AssisterJ a, defile d
•         where a.ndefile = d.ndefile
•         and a.njournaliste = j.njournaliste
•         and TO_CHAR(d.dateDefile, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY');

```

2. **S**-Combien de défilés chaque journaliste a-t-il couvert cette saison ?

```

• select j.nom, j.prenom, count(a.ndefile) as nombre_defiles
• from journaliste j, AssisterJ a, defile d
• where j.njournaliste = a.njournaliste
• and a.ndefile = d.ndefile
• and TO_CHAR(d.dateDefile, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
• group by j.nom, j.prenom;

```

3. **S**-Quels sont les journalistes qui ont couvert tous les défilés d'une maison de mode spécifique ?

```

• select j.nom, j.prenom, d.nomMaisonMode,
• from journaliste j, AssisterJ a, defile d
• where a.njournaliste = j.njournaliste
• and a.ndefile = d.ndefile
• and d.nomMaisonMode = 'Armani'
• group by j.nom, j.prenom, d.nomMaisonMode
• having count(distinct d.ndefile) = (
•
•         select count(*)
•         from defile
•         where nomMaisonMode = 'Armani');

```

4. **S**-Nombre total d'interviews réalisées par chaque journaliste par type d'interview ?

```

• select j.nom, j.prenom,
•         count(distinct im.nInterview) as interviews_mannequins,
•         count(distinct ii.nInterview) as interviews_invites,
•         count(distinct ic.nInterview) as interviews_createurs
• from journaliste j
• left join InterviewC ic on ic.njournaliste = j.njournaliste
• left join InterviewM im on im.njournaliste = j.njournaliste
• left join InterviewI ii on ii.njournaliste = j.njournaliste
• group by j.nom, j.prenom;

```

5. **S**-Quels journalistes n'ont réalisé aucune interview cette année ?

```

• select j.nom, j.prenom
• from journaliste j
• where not exists(
•     select * from InterviewI ii where j.njournaliste = ii.njournaliste and
    TO_CHAR(ii.heureDebut, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
•     union all
•     select * from InterviewC ic where j.njournaliste = ic.njournaliste and
    TO_CHAR(ic.heureDebut, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
•     union all
•     select * from InterviewM im where j.njournaliste = im.njournaliste and
    TO_CHAR(im.heureDebut, 'YYYY') = TO_CHAR(SYSDATE, 'YYYY')
• );
  
```

## Requêtes sur les sponsors

1. **L**-Liste des sponsors et le nombre de défilés sponsorisés.

```

• select s.nomSponsor, count(se.ndefile) as nombre_defiles_sponsoriser
• from sponsor s, sponsoriser se
• where s.nSponsor = se.nSponsor
• group by s.nomSponsor
• order by nombre_defiles_sponsoriser desc;
  
```

2. **L**-Sponsors ayant sponsorisé des défilés pour une maison de mode spécifique

```

• select s.nomSponsor, count(se.ndefile) as nombre_defiles_sponsorises
• from sponsor s, sponsoriser se, defile d
• where s.nSponsor = se.nSponsor
• and se.ndefile = d.ndefile
• and d.nomMaisonMode = 'Chanel';
• group by s.nomSponsor;
  
```

## D/ Vues

Admin :

Dans ce projet c'est l'utilisateur Etudiant sachant que c'est l'administrateur oracle qui lui a attribué pratiquement tous les droits.

## A-Mannequin :

Vue des défilés où le mannequin est programmé

```
• CREATE OR REPLACE VIEW Vue_Mannequin_Defiles AS
• SELECT
•     m.nMannequin,
•     m.nom AS NomMannequin,
•     m.prenom AS PrenomMannequin,
•     m.genre,
•     m.morphologie,
•     d.nDefile,
•     d.lieu,
•     d.dateDefile,
•     d.heureDebut,
•     d.heureFin,
•     d.theme,
•     d.descriptionDefile
• FROM
•     Mannequin m, Defile d, Participer p
• WHERE
•     m.nMannequin = p.nMannequin
•     AND p.nDefile = d.nDefile;
```

Vue des tenues attribuées au mannequin pour chaque défilé

```
• CREATE OR REPLACE VIEW Vue_Mannequin_Tenues AS
• SELECT
•     m.nMannequin,
•     m.nom AS NomMannequin,
•     m.prenom AS PrenomMannequin,
•     t.nTenue,
•     t.nomTenue,
•     t.description AS DescriptionTenue,
•     t.categorieTenue,
•     t.taille,
•     t.prix,
•     p.nDefile,
•     d.dateDefile
• FROM
•     Mannequin m, Tenue t, Defile d, Participer p
• WHERE
•     m.nMannequin = p.nMannequin
•     AND p.nTenue = t.nTenue
•     AND p.nDefile = d.nDefile;
```

## S-Maison de mode :

### Liste des collections produites par une maison de mode

```

• CREATE OR REPLACE VIEW V_Maison_Col AS
• SELECT
•     mm.nomMaisonMode,
•     c.nCollection,
•     c.nomCollection,
•     c.themeCollection,
•     c.saison,
•     c.nbrTenues
• FROM
•     MaisonMode mm, Collection c
• WHERE
•     mm.nomMaisonMode = c.nomMaisonMode;

```

### Informations des défilés organisés par une maison de mode

```

• CREATE OR REPLACE VIEW V_Maison_Def AS
• SELECT
•     mm.nomMaisonMode,
•     d.nDefile,
•     d.lieu,
•     d.dateDefile,
•     d.heureDebut,
•     d.heureFin,
•     d.theme,
•     d.descriptionDefile,
•     d.nbrPlaceMax
• FROM
•     MaisonMode mm, Defile d
• WHERE
•     mm.nomMaisonMode = d.nomMaisonMode;
•
•

```

## Liste des mannequins participant aux défilés d'une maison de mode

```

• CREATE OR REPLACE VIEW V_Maison_Def_Mannequins AS
• SELECT
•     mm.nomMaisonMode,
•     d.nDefile,
•     m.nMannequin,
•     m.nom AS NomMannequin,
•     m.prenom AS PrenomMannequin,
•     m.genre,
•     m.morphologie
• FROM
•     MaisonMode mm, Defile d, Participer p, Mannequin m
• WHERE
•     mm.nomMaisonMode = d.nomMaisonMode
•     AND d.nDefile = p.nDefile
•     AND p.nMannequin = m.nMannequin;
•
• GRANT SELECT ON V_Maison_Col TO MaisonsDeModes;
• GRANT SELECT ON V_Maison_Def TO MaisonsDeModes;
• GRANT SELECT, INSERT, UPDATE, DELETE ON Collection TO MaisonsDeModes;
• GRANT SELECT, INSERT, UPDATE, DELETE ON Defile TO MaisonsDeModes;
•

```

## I -Créateurs :

Cette vue va associer les créateurs à leurs collections et les tenues associés

```
• CREATE OR REPLACE VIEW V_Creat_Tenue_Col AS
• SELECT
•     cr.nCreateur,
•     cr.nom AS NomCreateur,
•     cr.prenom AS PrenomCreateur,
•     c.nCollection,
•     c.nomCollection,
•     c.themeCollection,
•     c.saison,
•     t.nTenue,
•     t.nomTenue,
•     t.description,
•     t.categorieTenue,
•     t.taille,
•     t.prix
• FROM
•     Createur cr, Collection c, Tenue t
• WHERE
•     cr.nCreateur = c.nCreateur
•     AND c.nCollection = t.nCollection;
•
```

Affiche les défilés où les collections du créateur sont présentées

```
• CREATE OR REPLACE VIEW V_Creat_Col_Def AS
• SELECT
•     cr.nCreateur,
•     cr.nom AS NomCreateur,
•     cr.prenom AS PrenomCreateur,
•     c.nCollection,
•     c.nomCollection,
•     d.nDefile,
•     d.lieu,
•     d.dateDefile,
•     d.heureDebut,
•     d.heureFin,
•     d.theme,
•     d.descriptionDefile
• FROM
•     Createur cr, Collection c, Defile d
• WHERE
•     cr.nCreateur = c.nCreateur
•     AND c.nomMaisonMode = d.nomMaisonMode;
•
• -- Droits pour les utilisateurs
• GRANT SELECT ON V_Creat_Tenue_Col TO Createurs;
• GRANT SELECT ON V_Creat_Col_Def TO Createurs;
• GRANT SELECT, INSERT, UPDATE, DELETE ON Tenue TO Createurs;
• GRANT SELECT, INSERT, UPDATE, DELETE ON Collection TO Createurs;
```

## S -Invités :

Liste des défilés auxquels un invité peut assister.

```
• CREATE OR REPLACE VIEW V_Inv_Def AS
• SELECT
•     i.nInvite,
•     i.nom AS NomInvite,
•     i.prenom AS PrenomInvite,
•     d.nDefile,
•     d.lieu,
•     d.dateDefile,
•     d.heureDebut,
•     d.heureFin,
•     d.theme,
•     d.descriptionDefile
• FROM
•     AssisterI a, defile d, invite i
• where a.nDefile = d.nDefile
• and i.nInvite = a.nInvite;
```

Liste les collections présentées lors des défilés auxquels ils sont invités.

```
• CREATE OR REPLACE VIEW V_Inv_Col AS
• SELECT
•     i.nInvite,
•     i.nom AS NomInvite,
•     i.prenom AS PrenomInvite,
•     c.nCollection,
•     c.nomCollection,
•     c.themeCollection,
•     c.saison,
•     c.nomMaisonMode
• FROM
•     AssisterI a, invite i, defile d, collection c
• where a.nDefile = d.nDefile
• and a.nInvite = i.nInvite
• and c.nomMaisonMode = d.nomMaisonMode;
• GRANT SELECT ON V_Inv_Def TO Invites;
• GRANT SELECT ON V_Inv_Col TO Invites;
```

## E/ Intégrité des données : les triggers

### Créateurs :

1. **A**- Chaque créateur doit être associé à une maison de mode.

- Nous avons ajouté la contrainte (NOT NULL) à l'attribut nomMaisonMode.
- nomMaisonMode **VARCHAR(50) NOT NULL**,

3. **S**- Un créateur ne peut pas participer à un défilé s'il n'a pas au moins une collection.

```

• CREATE OR REPLACE TRIGGER Verifier_Collection_Createur
• BEFORE INSERT ON Assister
• FOR EACH ROW
• DECLARE
•     v_count INT;
• BEGIN
•     -- Compter le nombre de collections pour le createur
•     SELECT COUNT(*)
•     INTO v_count
•     FROM Collection
•     WHERE nCreateur = :NEW.nCreateur;
•
•     IF v_count = 0 THEN
•         RAISE_APPLICATION_ERROR(-20001, 'Le créateur doit avoir au moins une
collection pour participer à un défilé.');

```

4. **S**- Un créateur doit avoir minimum 18 ans.

- Nous avons ajouté la contrainte dans la table Créateur.
- **CONSTRAINT ch\_c CHECK (dateNaissance <= TO\_DATE('2006-10-30', 'YYYY-MM-DD'))**



## Tenues :

1. **S**- Chaque tenue doit appartenir à une seule collection.

- Nous avons ajouté la contrainte (NOT NULL) à l'attribut nCollection avec nTenue est clé primaire.
- nCollection **INT NOT NULL**,

2. **A**- Les tenues doivent être à la taille d'au moins un mannequin.

```

• CREATE OR REPLACE TRIGGER Check_Mannequin_Tenue_Taille
• BEFORE INSERT OR UPDATE ON Participer
• FOR EACH ROW
• DECLARE
•     mannequin_taille NUMBER(5, 2);
•     tenue_taille NUMBER(5, 2);
• BEGIN
•     -- Récupérer la taille du mannequin
•     SELECT taille INTO mannequin_taille
•     FROM Mannequin
•     WHERE nMannequin = :NEW.nMannequin;
•
•     -- Récupérer la taille de la tenue
•     SELECT taille INTO tenue_taille
•     FROM Tenue
•     WHERE nTenue = :NEW.nTenue;
•
•     IF mannequin_taille != tenue_taille THEN
•         RAISE_APPLICATION_ERROR(-20001, 'La taille de la tenue doit correspondre à la
taille du mannequin.');

```

3. **S**- Le créateur de la tenue doit correspondre à celui de la collection lors de l'insertion.

```

• CREATE OR REPLACE TRIGGER trg_verif_createur_tenue
• BEFORE INSERT ON Tenue
• FOR EACH ROW
• DECLARE
•     ncreat_coll INT;

```

```

• BEGIN
•     -- Récupérer le nCreateur associé à la collection spécifiée dans l'insertion
•     SELECT nCreateur INTO ncreat_coll
•     FROM Collection
•     WHERE nCollection = :NEW.nCollection;
•
•     -- Comparer si le nCreateur de la table Tenue correspond à celui de Collection
•     IF :NEW.nCreateur != ncreat_coll THEN
•         RAISE_APPLICATION_ERROR(-20001, 'Le nCreateur ne correspond pas à celui de la
collection.');

```

4. A- Les tenues doivent être créées par le créateur qui travaille pour la maison de mode qui organise le défilé.

```

• CREATE OR REPLACE TRIGGER chk_createur_maisonmode
• BEFORE INSERT OR UPDATE ON Participer
• FOR EACH ROW
• DECLARE
•     v_nomMaisonModeDefile VARCHAR(50);
•     v_nomMaisonModeCreateur VARCHAR(50);
• BEGIN
•     -- Récupérer la maison de mode du défilé
•     SELECT nomMaisonMode
•     INTO v_nomMaisonModeDefile
•     FROM Defile
•     WHERE nDefile = :NEW.nDefile;
•
•     -- Récupérer la maison de mode du créateur
•     SELECT nomMaisonMode
•     INTO v_nomMaisonModeCreateur
•     FROM Createur
•     WHERE nCreateur = (SELECT nCreateur
•                        FROM Tenue
•                        WHERE nTenue = :NEW.nTenue);
•
•     IF v_nomMaisonModeDefile != v_nomMaisonModeCreateur THEN
•         RAISE_APPLICATION_ERROR(-20001, 'Le créateur de la tenue doit appartenir à la
maison de mode organisant le défilé.');
```

## Collections :

1. **S**- Chaque collection doit avoir un thème, un titre et une saison mode.

```

• nomCollection VARCHAR(100) NOT NULL,
• themeCollection VARCHAR(50) NOT NULL,
• saison VARCHAR(20) NOT NULL,

```

2. **S**- Une collection doit contenir au moins 10 tenues pour être considérée comme complète et être présentée lors d'un défilé.

```

• CREATE OR REPLACE TRIGGER check_collection_tenues
• BEFORE INSERT ON Participer
• FOR EACH ROW
• DECLARE
•     nbr_tenues INT;
• BEGIN
•     -- Récupérer le nombre de tenues dans la collection associée à la tenue
•     SELECT nbrTenues
•     INTO nbr_tenues
•     FROM Collection
•     WHERE nCollection = (SELECT nCollection
•                           FROM Tenue
•                           WHERE nTenue = :NEW.nTenue);
•
•     IF nbr_tenues < 10 THEN
•         RAISE_APPLICATION_ERROR(-20001, 'La collection associée à cette tenue doit
•         contenir au moins 10 tenues.');
```

3. **A**- Chaque collection doit être associée à un créateur et une maison de mode.

```

• nCreateur INT NOT NULL,
• nomMaisonMode VARCHAR(50) NOT NULL,

```

4. **A**- Le nom de la maison de mode doit correspondre à celui du créateur dans la table Collection.

```

• CREATE OR REPLACE TRIGGER trg_verif_createur_maisonmode
• BEFORE INSERT ON Collection
• FOR EACH ROW
• DECLARE
•     nommais_creat VARCHAR(50);
• BEGIN
•     -- Récupérer le nomMaisonMode correspondant au nCreateur donné
•     SELECT nomMaisonMode INTO nommais_creat
•     FROM Createur
•     WHERE nCreateur = :NEW.nCreateur;
•
•     IF :NEW.nomMaisonMode != nommais_creat THEN
•         RAISE_APPLICATION_ERROR(-20001, 'Le nCreateur ne correspond pas à celui de la
•         collection.');
```

5. **S**- Mise à jour automatique du nombre de tenues associées à une collection lors d'insertion ou de suppression.

```
• CREATE OR REPLACE TRIGGER T
• AFTER INSERT OR DELETE ON Tenue
• for each row
• begin
•     if inserting then
•         update Collection set
•         nbrTenues = nbrTenues + 1
•         where nCollection = :new.nCollection;
•     end if;
•
•     if deleting then
•         update Collection set
•         nbrTenues = nbrTenues - 1
•         where nCollection = :old.nCollection;
•     end if;
• end;
• /
```

## **L** Défilés :

1. Un défilé ne peut pas contenir moins de 10 tenues et peut avoir autant de tenues possibles.

```
• CREATE OR REPLACE TRIGGER Check_Lmt_tneue
• BEFORE INSERT ON Participer
• FOR EACH ROW
• DECLARE
•     v_total_tenues INT;
• BEGIN
•
•     SELECT COUNT(*)
•     INTO v_total_tenues
•     FROM Participer
•     WHERE nDefile = :NEW.nDefile;
•
•     -- Vérifier si le nombre de tenues est inférieur à 10
•     IF v_total_tenues >= 10 THEN
•
•         RETURN;
•     ELSE
•         -- Si le nombre de tenues est inférieur à 10, lever une erreur
•         RAISE_APPLICATION_ERROR(-20003, 'Un défilé doit contenir au moins 10
tenues. ');
•     END IF;
• END;
```

2. Un défilé ne peut être associé qu'à une seule maison de mode par date, heure et lieu.

- `ALTER TABLE Defile`
- `ADD CONSTRAINT unique_defile`
- `UNIQUE (nomMaisonMode, dateDefile, heureDebut, lieu);`

3. Les défilés ne peuvent commencer qu'à partir de 14h.

- Nous avons ajouté la contrainte dans la table Défilé.
- `CONSTRAINT ch_d CHECK (EXTRACT(HOUR FROM heureDebut) >= 14)`

4. Un défilé doit avoir au moins 10 mannequins inscrits pour être validé.

```

• CREATE OR REPLACE TRIGGER Check_lmit_mnq1
• BEFORE INSERT ON Participer
• FOR EACH ROW
• DECLARE
•     v_total_mannequins INT;
• BEGIN
•
•     SELECT COUNT(*)
•     INTO v_total_mannequins
•     FROM Participer
•     WHERE nDefile = :NEW.nDefile;
•
•     IF v_total_mannequins >= 10 THEN
•
•         RETURN;
•     ELSE
•
•         RAISE_APPLICATION_ERROR(-20004, 'Un défilé doit avoir au moins 10 mannequins
inscrits.');
```

## Mannequins :

1. **S**- Un mannequin ne peut pas défiler plus de 3 fois par jour.

```

• CREATE OR REPLACE TRIGGER Verif_Nb_Defiles_Mannequin
• BEFORE INSERT ON Participer
• FOR EACH ROW
• DECLARE
•     v_count INT;
• BEGIN
•     SELECT COUNT(*)
•     INTO v_count
•     FROM Participer p
•     JOIN Defile d ON p.nDefile = d.nDefile
•     WHERE p.nMannequin = :NEW.nMannequin
•     AND TRUNC(d.heureDebut) = TRUNC(SYSDATE); -- Compare la date (sans l'heure)
•
•     IF v_count >= 3 THEN
•         RAISE_APPLICATION_ERROR(-20003, 'Un mannequin ne peut pas défiler plus de 3
fois par jour.');
```

2. **S**- Chaque mannequin doit être associé à une tenue lorsqu'il participe à un défilé.

- nMannequin **INT** not null,
- nDefile **INT** not null,
- nTenue **INT** NOT NULL,

3. **S**- Un mannequin ne peut pas être affecté à plus de 5 tenues au cours d'un même défilé.

```
• CREATE OR REPLACE TRIGGER Verif_Nb_Tenues_Mannequin
• BEFORE INSERT ON Participer
• FOR EACH ROW
• DECLARE
•     v_count INT;
• BEGIN
•     SELECT COUNT(*)
•     INTO v_count
•     FROM Participer
•     WHERE nMannequin = :NEW.nMannequin
•     AND nDefile = :NEW.nDefile;
•
•     IF v_count >= 5 THEN
•         RAISE_APPLICATION_ERROR(-20005, 'Un mannequin ne peut pas être affecté à plus
de 5 tenues au cours d'un même défilé.');
```

4. **S**- Un mannequin ne peut pas avoir moins de 1.70 cm de taille.

- Nous avons ajouté la contrainte dans la table Défilé.
- taille **NUMBER**(5, 2) **CHECK** (taille >= 170),

5. **S**- Un mannequin doit avoir au minimum 16 ans pour participer à un défilé.

- Nous avons ajouté la contrainte dans la table Défilé.
- age **INT** **CHECK** (age >= 16),

## A-Maisons de mode :

- Une maison de mode ne peut avoir plus d'un créateur principal sous contrat.

```
• ALTER TABLE Createur  
• ADD CONSTRAINT unique_maison_createur UNIQUE (nomMaisonMode);
```



## S- Invités :

- Le nombre d'invités ne peut pas dépasser le nombre de places disponibles dans le défilé.

```

• CREATE OR REPLACE TRIGGER Check_Limit_dePlaces
• BEFORE INSERT ON AssisterI
• FOR EACH ROW
• DECLARE
•     total_invites INT;
•     places_disponibles INT;
• BEGIN
•     -- je compte le nombre d'invités qui assiste avec table ASSISTER I
•     SELECT COUNT(*)
•     INTO total_invites
•     FROM AssisterI
•     WHERE nDefile = :NEW.nDefile;
•
•     -- Obtenir le nombre de places disponibles pour ce défilé
•     SELECT nbrPlaceMax
•     INTO places_disponibles
•     FROM Defile
•     WHERE nDefile = :NEW.nDefile;
•
•     -- Vérifier si le nombre d'invités dépasse la capacité maximale
•     IF total_invites >= places_disponibles THEN
•         RAISE_APPLICATION_ERROR(-20002, 'Le nombre d''invités pour ce défilé dépasse
la capacité maximale.');
```

## L - Journalistes :

- Un journaliste ne peut pas assister à plus de cinq défilés par jour pendant la Fashion Week

```

• CREATE OR REPLACE TRIGGER Check_Journaliste
• BEFORE INSERT ON AssisterJ
• FOR EACH ROW
• DECLARE
•     v_total_defiles INT;
• BEGIN
•     -- Compter le nombre de défilés auxquels le journaliste est déjà inscrit ce jour-
    là
•     SELECT COUNT(*)
•     INTO v_total_defiles
•     FROM AssisterJ a
•     JOIN Defile d ON a.nDefile = d.nDefile
•     WHERE a.nJournaliste = :NEW.nJournaliste
•           AND TRUNC(d.dateDefile) = TRUNC(:NEW.heureDepart); -- Comparer la date sans
    l'heure
•
•     -- Vérifier si le journaliste est déjà inscrit à 5 défilés ce jour-là
•     IF v_total_defiles >= 5 THEN
•         RAISE_APPLICATION_ERROR(-20001, 'Un journaliste ne peut pas assister à plus
    de cinq défilés par jour.');
```

## S- Sponsors :

- Un sponsor doit sponsoriser au maximum 3 défilés.

```

• CREATE OR REPLACE TRIGGER Verif_Nb_Defiles_Sponsor
• BEFORE INSERT ON Sponsoriser
• FOR EACH ROW
• DECLARE
•     v_count INT;
• BEGIN
•     SELECT COUNT(*)
•     INTO v_count
•     FROM Sponsoriser
•     WHERE nSponsor = :NEW.nSponsor;
•
•     IF v_count >= 3 THEN
•         RAISE_APPLICATION_ERROR(-20002, 'Un sponsor ne peut pas sponsoriser plus de 3
    défilés.');
```

## A-Suppression des tables existantes.

Nous avons utilisé cette procédure pour éviter les erreurs de type ORA-00942 : table ou vue inexistante, qui pourraient survenir après chaque lancement du projet.

```

• CREATE OR REPLACE PROCEDURE drop_table_if_exists (table_name VARCHAR2) IS
• BEGIN
•     EXECUTE IMMEDIATE 'DROP TABLE ' || table_name || ' CASCADE CONSTRAINTS';
• EXCEPTION
•     WHEN OTHERS THEN
•         IF SQLCODE != -942 THEN
•             RAISE;
•         END IF;
• END;
• /
• BEGIN
•     drop_table_if_exists('Participer');
•     drop_table_if_exists('Sponsoriser');
•     drop_table_if_exists('InterviewI');
•     drop_table_if_exists('InterviewC');
•     drop_table_if_exists('InterviewM');
•     drop_table_if_exists('AssisterJ');
•     drop_table_if_exists('AssisterC');
•     drop_table_if_exists('AssisterI');
•     drop_table_if_exists('Tenue');
•     drop_table_if_exists('Collection');
•     drop_table_if_exists('Defile');
•     drop_table_if_exists('Createur');
•     drop_table_if_exists('Mannequin');
•     drop_table_if_exists('Sponsor');
•     drop_table_if_exists('Invite');
•     drop_table_if_exists('Journaliste');
•     drop_table_if_exists('MaisonMode');
• END;
• /

```

## S- Création et Vérification Automatique des Utilisateurs Oracle.

Ce script crée automatiquement des utilisateurs en vérifiant s'ils existent déjà.

```

• BEGIN
•   -- Vérification et création des utilisateurs
•   FOR user_info IN (
•       SELECT 'Invites' AS username, '2024' AS password FROM DUAL
•       UNION ALL
•       SELECT 'Mannequins', '2024' FROM DUAL
•       UNION ALL
•       SELECT 'Createurs', '2024' FROM DUAL
•       UNION ALL
•       SELECT 'MaisonsDeModes', '2024' FROM DUAL
•   ) LOOP
•       BEGIN
•           EXECUTE IMMEDIATE 'CREATE USER ' || user_info.username || ' IDENTIFIED BY
• ' || user_info.password;
•           DBMS_OUTPUT.PUT_LINE('Utilisateur "' || user_info.username || '" créé
avec succès.');
```

```

•       EXCEPTION
•           WHEN OTHERS THEN
•               IF SQLCODE = -1920 THEN
•                   DBMS_OUTPUT.PUT_LINE('Utilisateur "' || user_info.username || '"
existe déjà.');
```

```

•               ELSE
•                   RAISE;
•               END IF;
•           END;
•       END LOOP;
•   END;
• /
•

```

## F/ Méta-données

Script SQL qui donne la liste de toutes les contraintes d'intégrité définies sur notre BD, classer par table, par type de contrainte et par le corps de chaque contrainte.

A- Script SQL qui donne la liste de tous les triggers, classés par nom de table.

```
• SELECT
•     TABLE_NAME,
•     TRIGGER_NAME,
•     TRIGGER_TYPE,
•     STATUS,
•     DESCRIPTION
• FROM
•     user_triggers
• ORDER BY
•     TABLE_NAME, TRIGGER_NAME;
```

S- Script SQL qui donne la Liste des tables sans clé primaire, classés par nom de table. « La liste des tables sans clé primaire est importante pour analyser et garantir l'intégrité et la gestion des données dans une base de données relationnelle. »

```
• SELECT
•     TABLE_NAME
• FROM
•     user_tables
• WHERE
•     TABLE_NAME NOT IN (
•         SELECT TABLE_NAME
•         FROM user_constraints
•         WHERE CONSTRAINT_TYPE = 'P'
•     )
• ORDER BY
•     TABLE_NAME;
```

A- Ce script donne toutes les contraintes d'intégrité, classées par table, par type, avec le corps de chaque contrainte

```
• SELECT
•     TABLE_NAME,
•     CONSTRAINT_NAME,
•     CONSTRAINT_TYPE,
•     SEARCH_CONDITION
• FROM
•     USER_CONSTRAINTS
• ORDER BY
•     TABLE_NAME,
•     CONSTRAINT_TYPE,
•     CONSTRAINT_NAME;
```

**S**- Ce script affiche la liste des colonnes et leurs types pour chaque table

```
• SELECT
•     TABLE_NAME,
•     COLUMN_NAME,
•     DATA_TYPE,
•     DATA_LENGTH
• FROM
•     USER_TAB_COLUMNS
• ORDER BY
•     TABLE_NAME, COLUMN_NAME;
```