

IUT de Fontainebleau

# Traduction en modèle relationnel

Sarah EL MCHANTEF, Maylee PEIRO-TOMAS

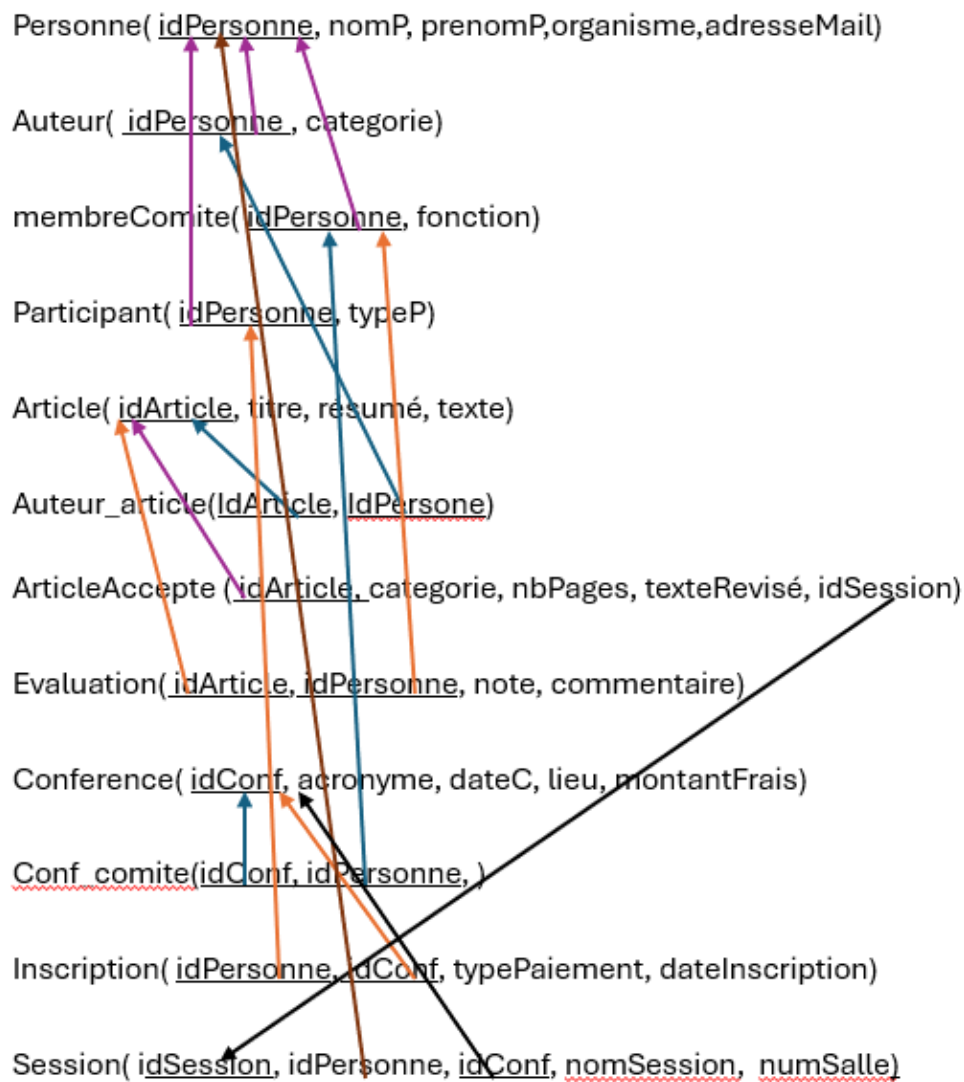
## Table des matières

Schéma de la base de données .....	2
Copie d'écran des requêtes saisies sur Oracle.....	6
Requêtes CREATE TABLE : .....	12

## Schéma de la base de données

Lorsque chaque classe devient une relation. Et l'identifiant devient la clé primaire de la relation, nous appliquons la règle R1.

Nous avons utilisé un code couleur pour faciliter la l'explications entre le texte et les flèches sur le schéma ci-dessous.



### Règle R5-3 :

Nous avons un héritage « Personne », on garde les 4 (Personne, Auteur, membreComité, et Participant). On ne garde pas seulement Auteur, membreComité et Participant car il y

a totalité, c'est-à-dire une personne peut être soit un Auteur, soit un membreComité, soit un Participant ou bien combiner ces rôles. On ne garde pas seulement Personne car cela nécessiterait de rajouter des contraintes complexes pour traduire les associations. (Règle 5.3)

La clé primaire de la super-classe (Personne) migre dans les sous-classes (Auteur, membreComité et Participant) comme clé primaire et étrangère. Cette solution est adaptée car la super-classe et les sous-classes ont des attributs et des associations propres.

On a aussi un héritage entre Article et ArticleAccepté, on utilisera aussi la règle 5-3, car la super-classe (Article) et la sous-classe (ArticleAccepté) ont des attributs et des associations propres. Ainsi la clé primaire de la classe Article (idArticle) migre dans la sous-classe ArticleAccepté comme clé primaire et étrangère.

## **Règle R2-2**

Nous avons ensuite deux associations binaires plusieurs-à-plusieurs, entre Auteur et Article et entre membreComité et Conférence.

Pour Auteur et Article : Cette association est une relation binaire plusieurs-à-plusieurs entre les entités Auteur et Article, où un article peut avoir plusieurs auteurs, et un auteur peut écrire plusieurs articles. Ainsi l'association donne lieu à une relation nommée : Auteur\_article. La clé primaire de cette relation est composée par la concaténation des clés primaires des deux entités connectées à l'association : idPersonne (provenant de Auteur) et idArticle (provenant de Article). Ces clés deviennent également clés étrangères dans la relation Auteur\_article afin de garantir le lien avec les entités Auteur et Article.

Pour Conférence et membreComité : On constate le même principe, cette association est également une relation binaire plusieurs-à-plusieurs entre Conférence et membreComité, car une conférence peut avoir plusieurs membres du comité, et un membre du comité peut être attaché à aucune, ou plusieurs conférences. Ainsi, l'association donne lieu à une relation nommée : Conf\_comite. La clé primaire de cette relation est composée des clés primaires des deux entités connectées : idConf (provenant de Conférence) et idPersonne (provenant de membreComité). Ces clés deviennent également clés étrangères dans la relation Conf\_comite.

## **Règle R4-1**

Nous avons ensuite deux Association binaire avec classe-association : entre Participant et Conférence (avec une classe-association Inscription), et entre membreComité et Article (avec une classe-association Evaluation).

Pour la relation entre Participant et Conférence : Il s'agit d'une association binaire entre Participant et Conférence, enrichie par une classe-association Inscription. La clé primaire de cette relation est composée par la concaténation des clés primaires des deux classes connectées : idPersonne (provenant de Participant) et idConf (provenant de Conférence). Ces clés deviennent également clés étrangères dans la relation Inscription, pointant vers les relations Participant et Conférence. Les attributs de la classe association inscription (dateInscription et typePaiement) deviennent des attributs de la relation.

Pour la relation entre membreComité et Article : Il s'agit cette fois-ci d'une association binaire entre membreComité et Article, enrichie par une classe-association Evaluation. La clé primaire de cette relation est composée par la concaténation des clés primaires des deux classes connectées : idPersonne (provenant de membreComité) et idArticle (provenant de Article). Ces clés deviennent également clés étrangères dans la relation Evaluation, pointant vers les relations membreComité et Article. Les attributs de la classe association Evaluation (note et commentaire) deviennent des attributs de la relation.

#### **Règle R2-4 :**

L'association entre Session et inscription est décrite par la règle R2-4 : Association avec multiplicités maximales à 1. Étant donné qu'une session a un seul président qui figure parmi les inscrits, la classe session va recevoir une clé étrangère : idPersonne, qui sera l'identifiant du président de la session.

#### **Règle R2-1 :**

Nous avons une association binaire, 1 à plusieurs, avec identifiant relatif entre Conférence et Session. Ainsi, Il faut rajouter un attribut de type clé étrangère dans la relation « fils »( ici Session) de l'association. L'attribut porte le nom de la clé primaire de la relation « père », ici idConf de la relation Conférence.

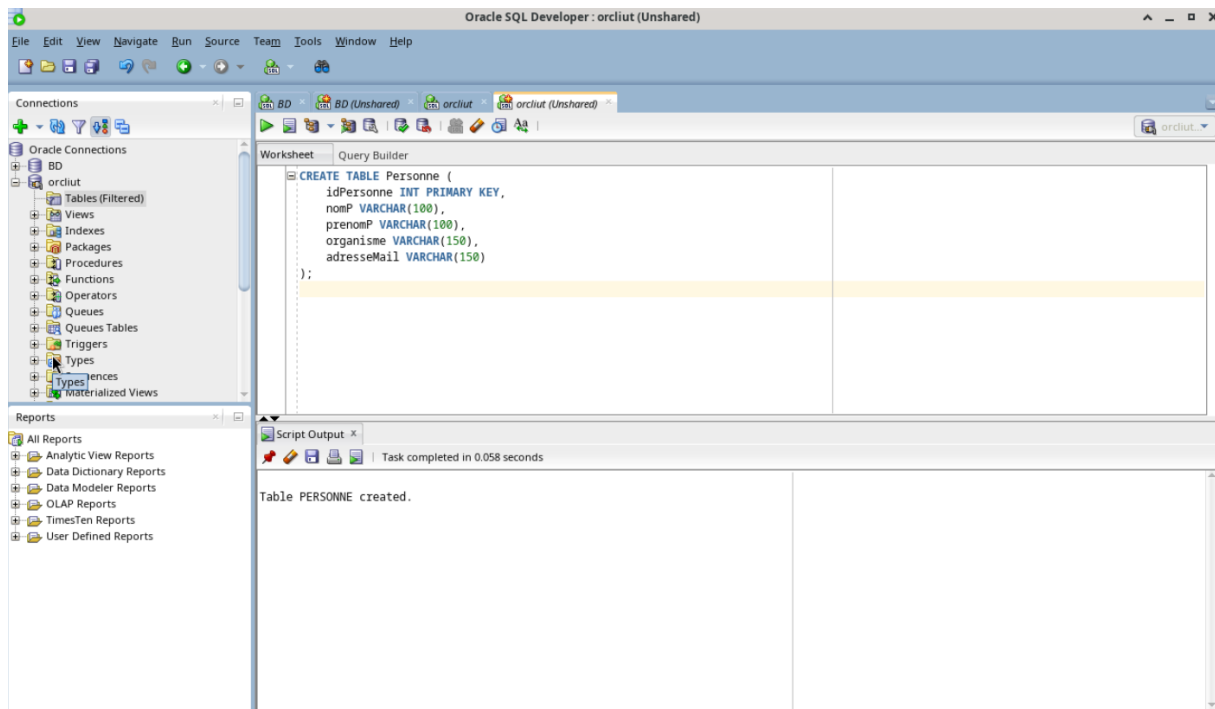
Pour faciliter l'identification de la classe Session, nous avons décider de rajouter un identifiant de la Session, appelé idSession. Néanmoins l'attribut idSession ne permet pas de distinguer entre les objets de la classe Session. Une Session dépend fortement de la Conférence à laquelle elle est rattachée. On part du principe que les conférences ont plusieurs sessions (par exemple : pour la conférence sur le climat, il y a la session numéro 1, session numéro 2, etc., et pour une conférence sur l'économie international,

il y a aussi la session numéro 1, session numéro 2, etc). Ainsi l'identifiant de la classe Session est composé des attributs idSession + idConf. On l'appelle identifiant relatif.

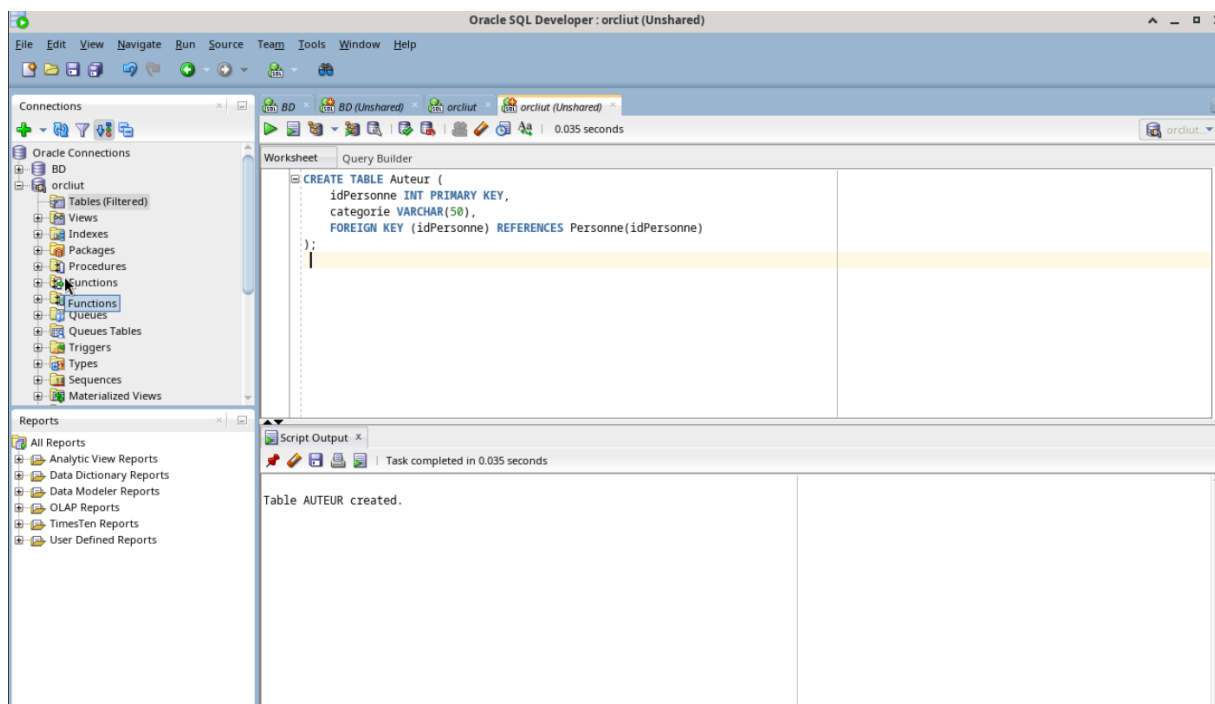
Pour la relation entre Session et ArticleAccepté, c'est aussi une association binaire 1 à plusieurs (R2-1), étant donné qu'un article accepté ne peut être présenté qu'à une seule session, la clé primaire de la classe Session (idSession) va migrer dans la classe ArticleAccepté.

# Copie d'écran des requêtes saisies sur Oracle

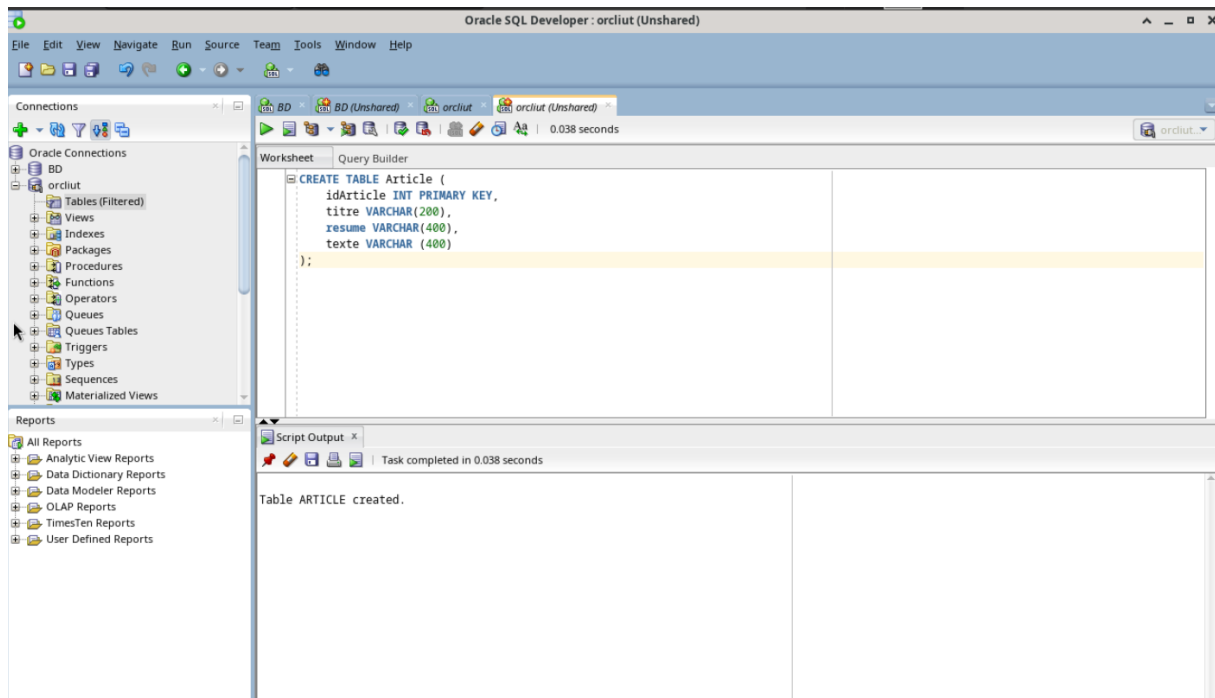
## Table Personne :



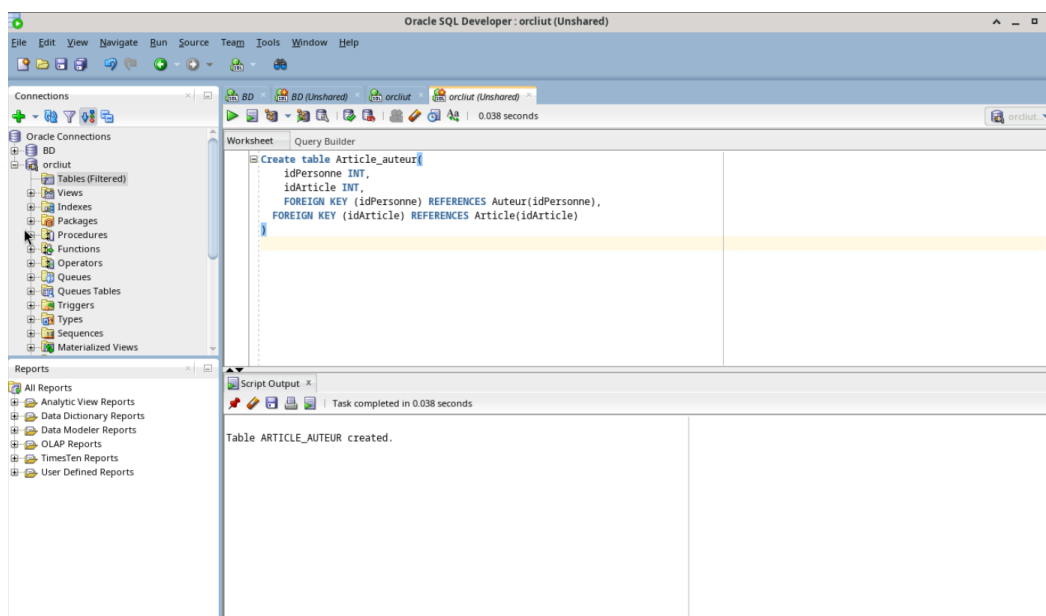
## Table Auteur :



## Table Article :

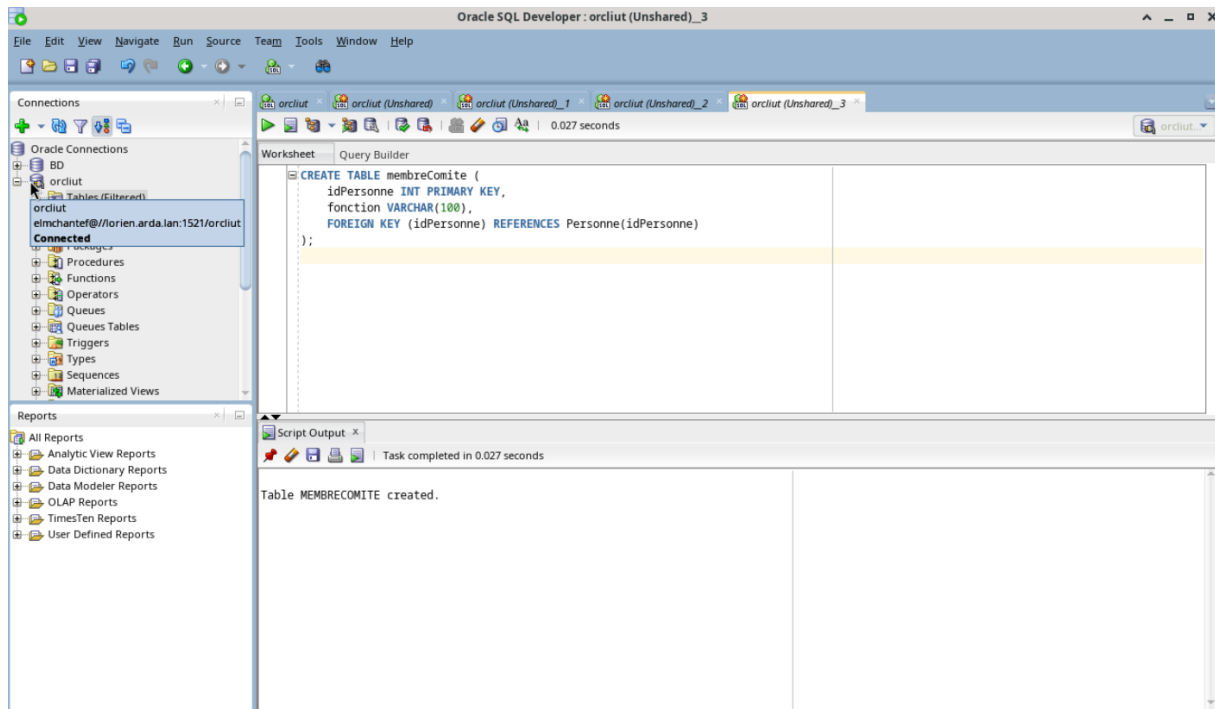


## Table Article auteur :

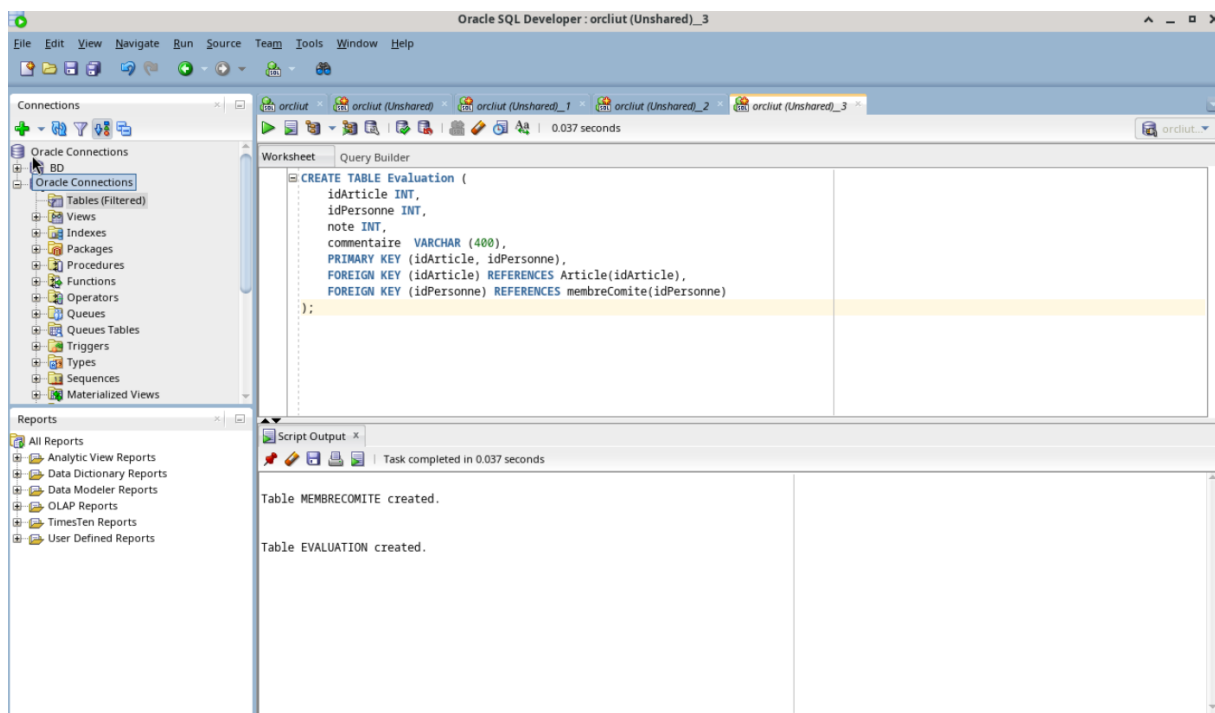




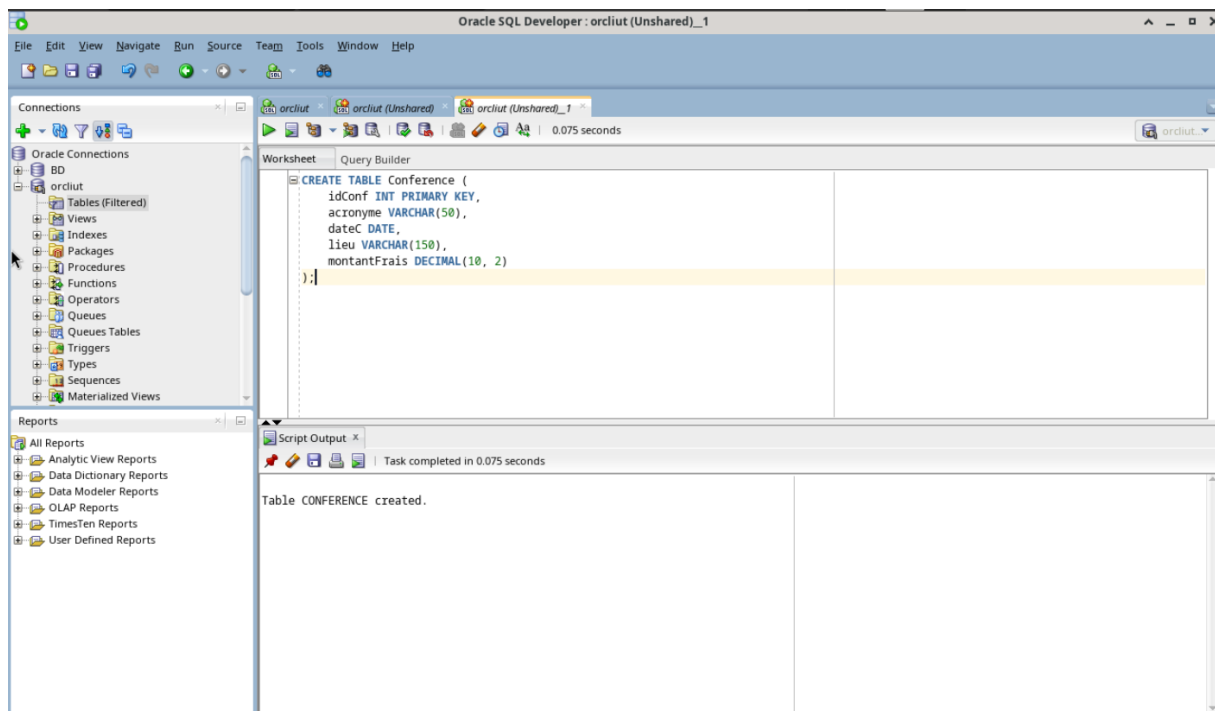
## Table membreComite :



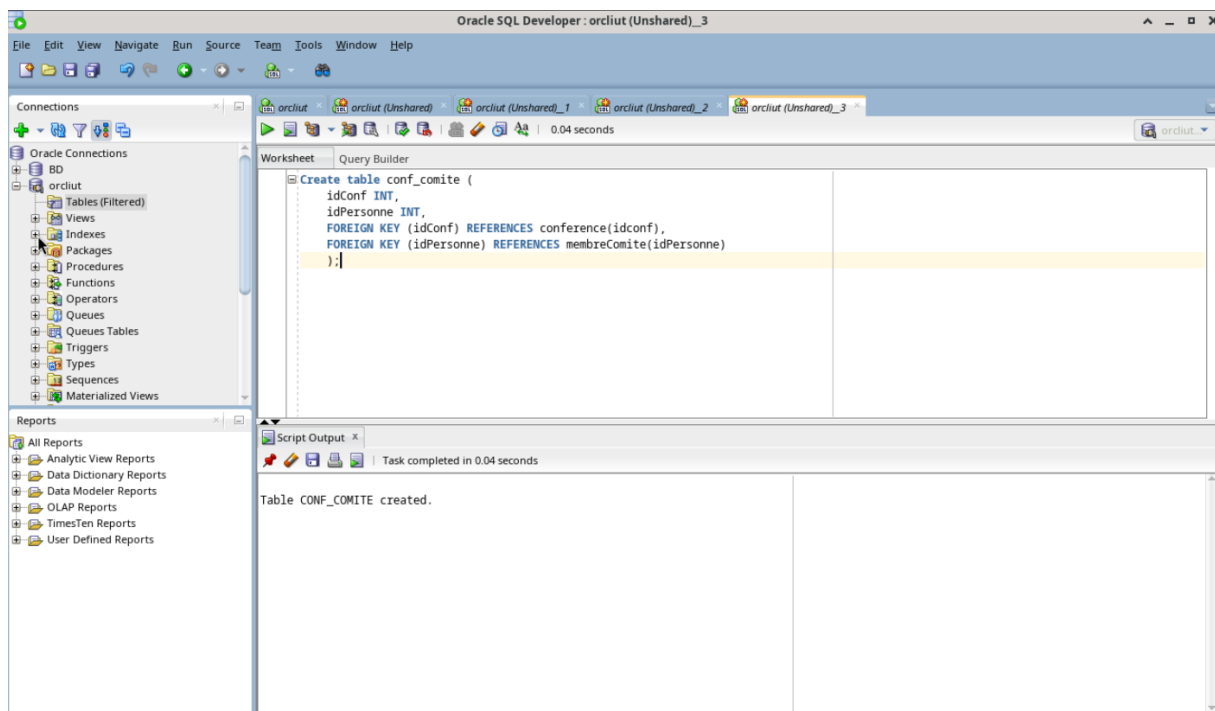
## Table Evaluation :



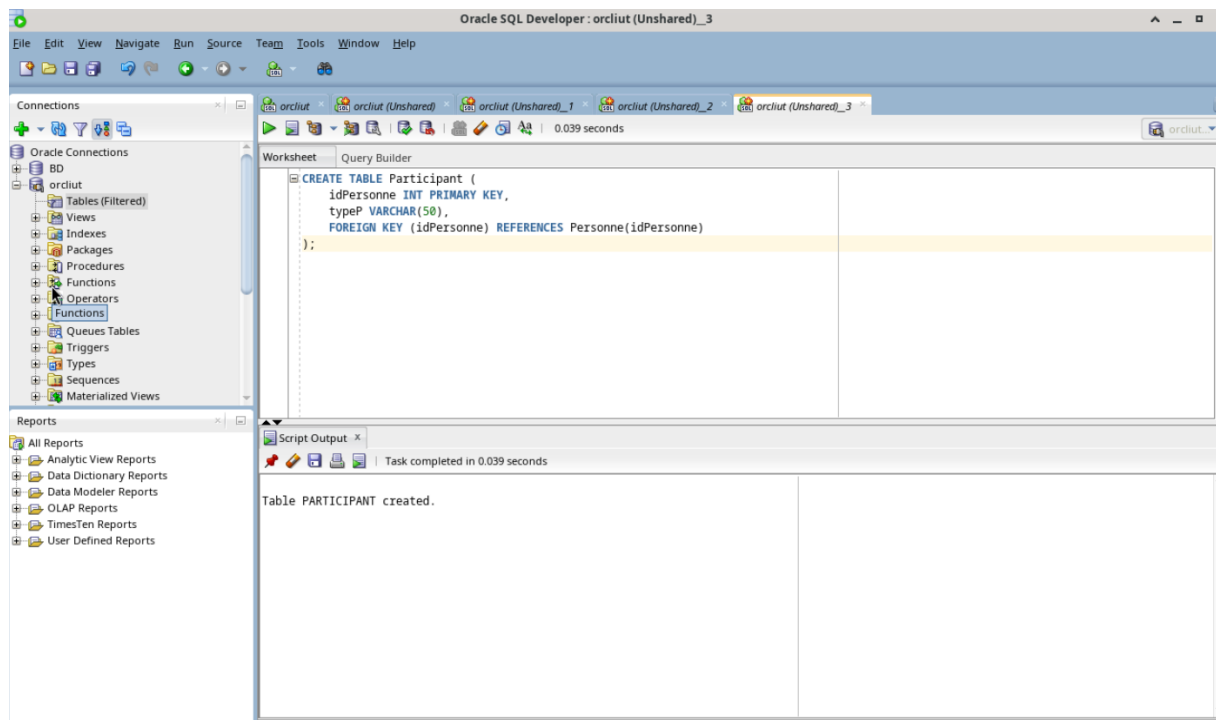
## Table Conference :



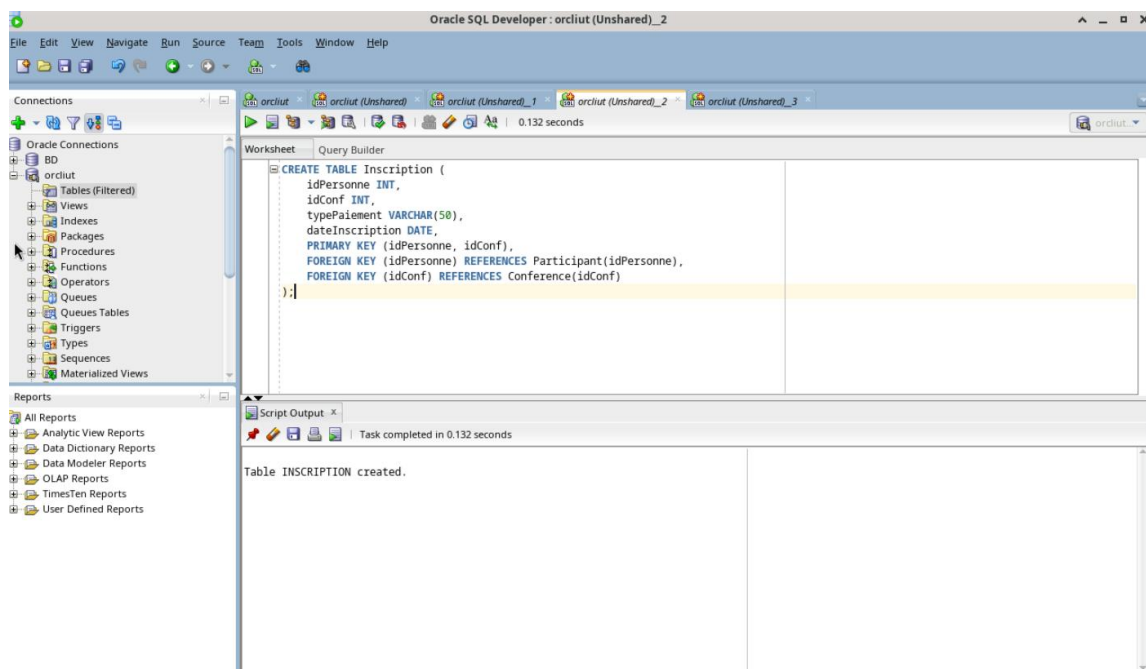
## Table conf\_comite:



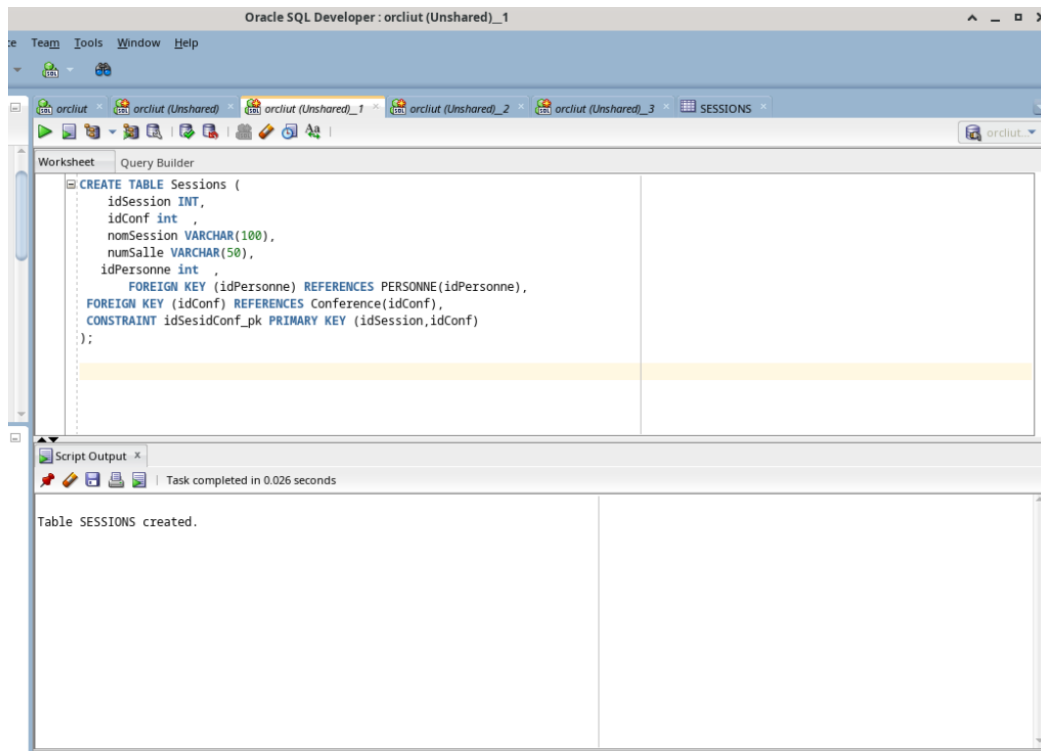
## Table Participant :



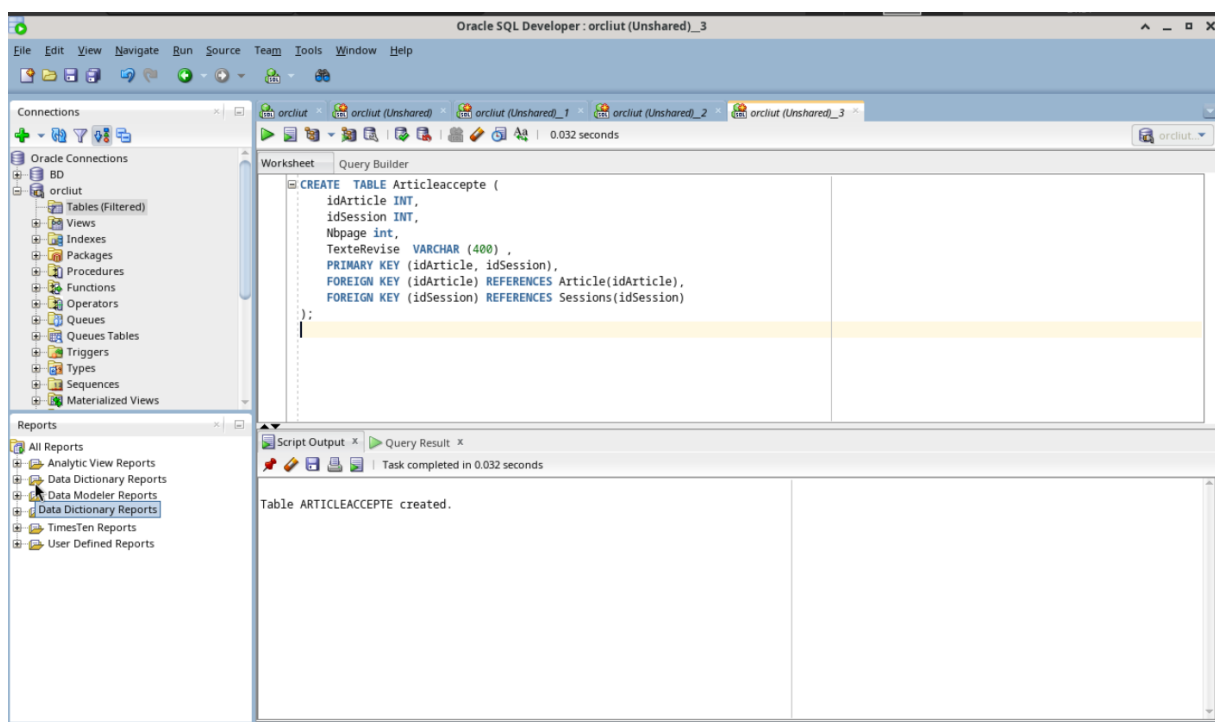
## Table Inscription :



**Table Sessions :** (Le nom de la table Session a été changé en Sessions car le mot Session est un mot réservé à Oracle).



**Table Articleaccepte:**



## Requêtes CREATE TABLE :

--Table : Personne

```
CREATE TABLE Personne (  
    idPersonne INT PRIMARY KEY,  
    nomP VARCHAR(100),  
    prenomP VARCHAR(100),  
    organisme VARCHAR(150),  
    adresseMail VARCHAR(150)  
);
```

-- Table : Auteur

```
CREATE TABLE Auteur (  
    idPersonne INT PRIMARY KEY,  
    categorie VARCHAR(50),  
    FOREIGN KEY (idPersonne) REFERENCES Personne(idPersonne)  
);
```

-- Table : Article

```
CREATE TABLE Article (  
    idArticle INT PRIMARY KEY,  
    titre VARCHAR(200),  
    resume VARCHAR(400),  
    texte VARCHAR (400)  
);
```

---Table Article\_Auteur

```
Create table Article_auteur(  
    idPersonne INT,  
    idArticle INT,  
    FOREIGN KEY (idPersonne) REFERENCES Auteur(idPersonne),  
    FOREIGN KEY (idArticle) REFERENCES Article(idArticle)  
)
```

-- Table : membreComite

```
CREATE TABLE membreComite (  
    idPersonne INT PRIMARY KEY,
```

```
fonction VARCHAR(100),  
FOREIGN KEY (idPersonne) REFERENCES Personne(idPersonne)  
);
```

-- Table : Evaluation

```
CREATE TABLE Evaluation (  
    idArticle INT,  
    idPersonne INT,  
    note INT,  
    commentaire VARCHAR (400),  
    PRIMARY KEY (idArticle, idPersonne),  
    FOREIGN KEY (idArticle) REFERENCES Article(idArticle),  
    FOREIGN KEY (idPersonne) REFERENCES membreComite(idPersonne)  
);
```

-- Table : Conference

```
CREATE TABLE Conference (  
    idConf INT PRIMARY KEY,  
    acronyme VARCHAR(50),  
    dateC DATE,  
    lieu VARCHAR(150),  
    montantFrais DECIMAL(10, 2)  
);
```

-- Table conf \_comite

```
Create table conf _comite (  
    idConf INT,  
    idPersonne INT,  
    FOREIGN KEY (idConf) REFERENCES conference(idconf),  
    FOREIGN KEY (idPersonne) REFERENCES membreComite(idPersonne))
```

-- Table : Participant

```
CREATE TABLE Participant (  
    idPersonne INT PRIMARY KEY,  
    typeP VARCHAR(50),  
    FOREIGN KEY (idPersonne) REFERENCES Personne(idPersonne)  
);
```

-- Table : Inscription

```
CREATE TABLE Inscription (  
    idPersonne INT,  
    idConf INT,  
    typePaiement VARCHAR(50),  
    dateInscription DATE,  
    PRIMARY KEY (idPersonne, idConf),  
    FOREIGN KEY (idPersonne) REFERENCES Participant(idPersonne),  
    FOREIGN KEY (idConf) REFERENCES Conference(idConf)  
);
```

-- Table : Session

```
CREATE TABLE Sessions (  
    idSession INT,  
    nomSession VARCHAR(100),  
    numSalle VARCHAR(50),  
    idConf INT,  
    idPersonne INT,  
    FOREIGN KEY (idPersonne) REFERENCES Personne(idPersonne),  
    FOREIGN KEY (idConf) REFERENCES Conference(idConf),  
    CONSTRAINT idSession_pk PRIMARY KEY (idSession, idConf)  
);
```

-- Table : Articleaccepte

```
CREATE TABLE Articleaccepte (  
    idArticle INT,  
    idSession INT,  
    Nbpage int,  
    Text_revise VARCHAR (400) ,  
    PRIMARY KEY (idArticle, idSession),  
    FOREIGN KEY (idArticle) REFERENCES Article(idArticle),  
    FOREIGN KEY (idSession) REFERENCES Sessions(idSession)  
);
```