

Instituto Tecnológico de Costa Rica

Campus Tecnológico Local San José

Laboratorio 01: Creacion de tablas

Profesor:

Adriana Álvarez Figueroa

Estudiantes:

González Prendas Mauricio: 2024143009

Hidalgo Paz Carmen: 2020030538

Meza Marin Dylan: 2021577352

Fecha de entrega:

31/03/25

Semestre I 2025

Tabla de contenidos.

1. Laboratorio de Base de Datos	2
1.1. Creación de los tablespaces	2
1.2. Creación del esquema GE	3
1.3. Creación de las tablas en el esquema GE	4
1.4. Creación de llaves primarias	8
1.5. Creación de constraints	10
1.6. Creación del Esquema AD	12
1.7. Creación de tablas en AD según el requerimiento	14
1.8. Renombrar columnas en una tabla	18
1.9. Ampliación del tamaño de nombre en People	19

1. Laboratorio de Base de Datos

1.1. Creación de los tablespaces

Código:

```
CREATE TABLESPACE GE_Data
DATAFILE '/u01/app/oracle/oradata/XE/gedata01.dbf'
SIZE 10M
REUSE
AUTOEXTEND ON
NEXT 512k
MAXSIZE 200M;

CREATE TABLESPACE GE_Index
DATAFILE '/u01/app/oracle/oradata/XE/geindex01.dbf'
SIZE 10M
REUSE
AUTOEXTEND ON
NEXT 512k
MAXSIZE 200M;
```

Evidencia:

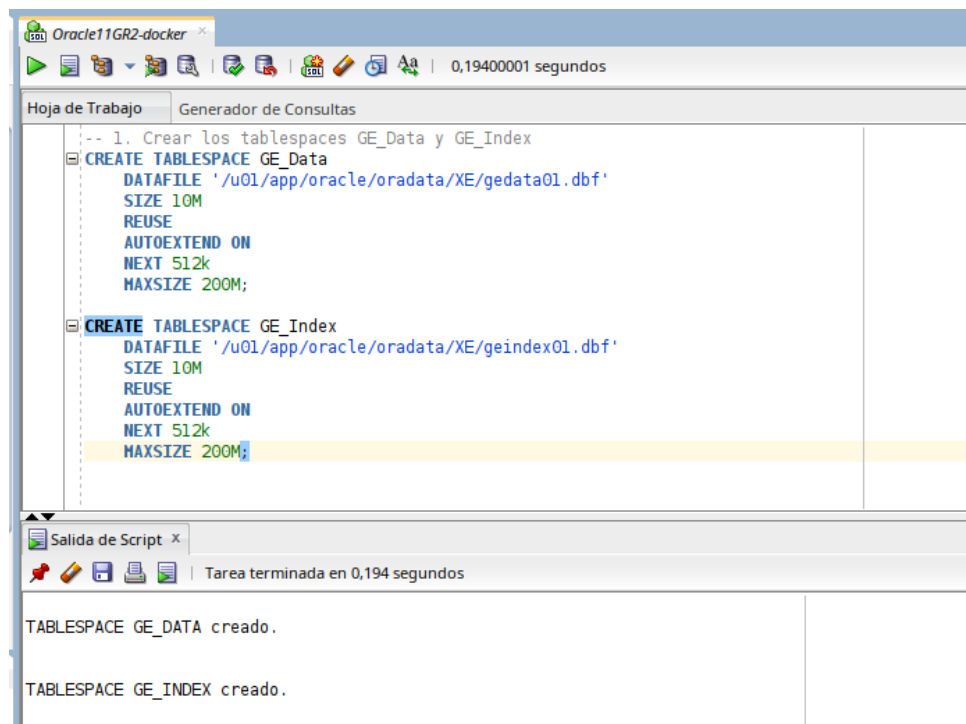


Figura 1: Evidencia de creación de tablespaces

1.2. Creación del esquema GE

Código:

```
CREATE USER GE
IDENTIFIED BY ge
DEFAULT TABLESPACE GE_Data
QUOTA 10M ON GE_Data
TEMPORARY TABLESPACE temp
QUOTA 5M ON SYSTEM
QUOTA 10M ON GE_Index;

GRANT CONNECT to GE;
GRANT CREATE SESSION to GE;
GRANT CREATE TABLE to GE;
```

Respuestas:

- ¿Se pueden crear varios esquemas en la misma BD? Sí, en Oracle un esquema es equivalente a un usuario con objetos propios.
- ¿Cuál es el propósito de crear más de un esquema en la misma BD? Separar datos y mejorar la organización y seguridad de la información.

Evidencia:

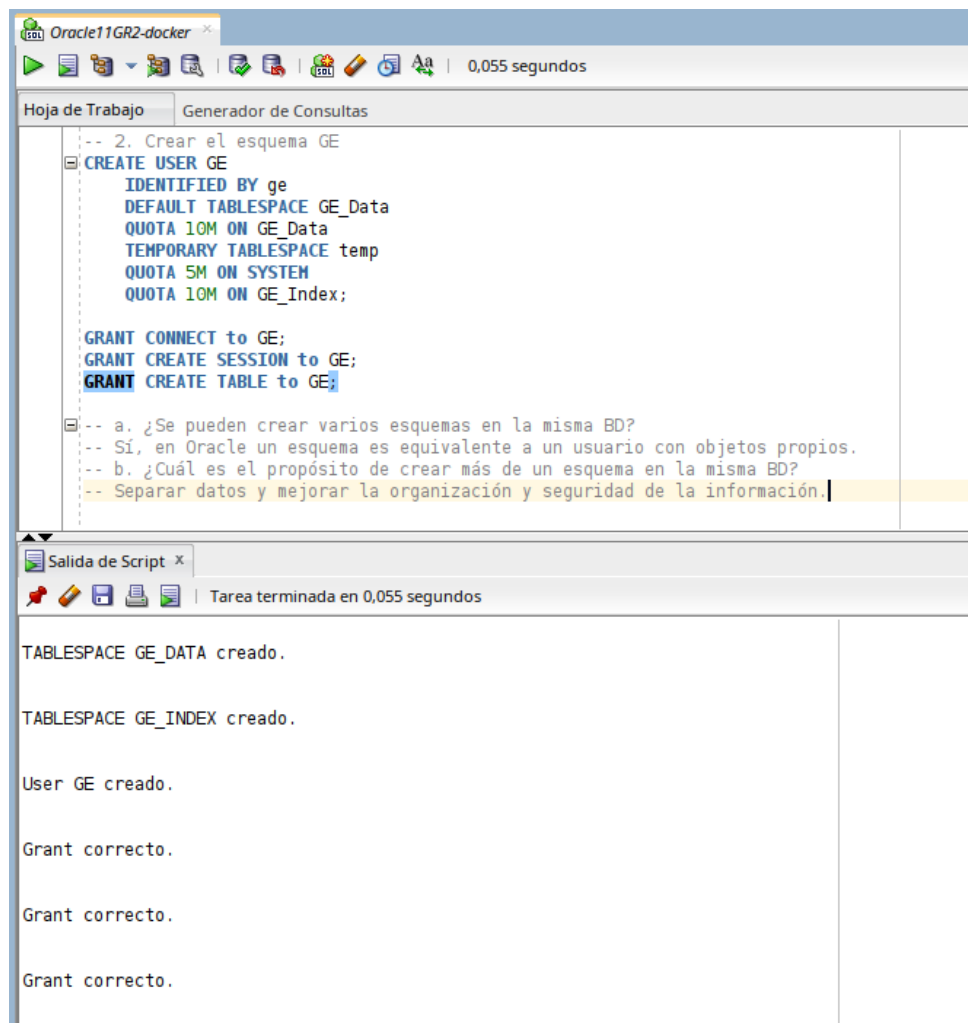


Figura 2: Evidencia de creación del esquema GE

1.3. Creación de las tablas en el esquema GE

Modelo Conceptual:

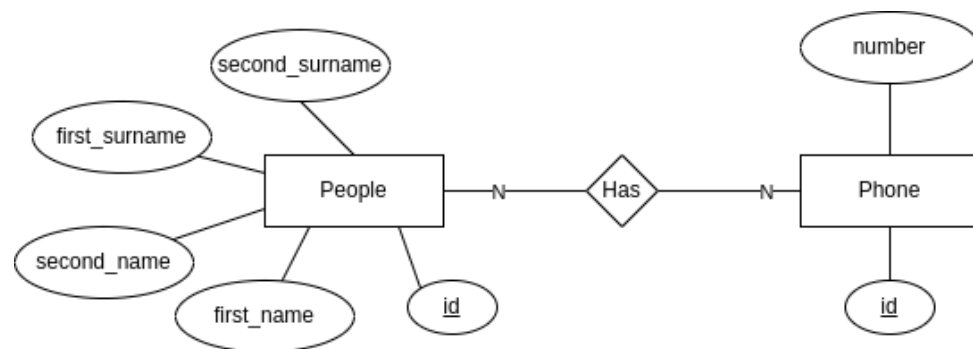


Figura 3: Modelo conceptual de las tablas en el esquema GE

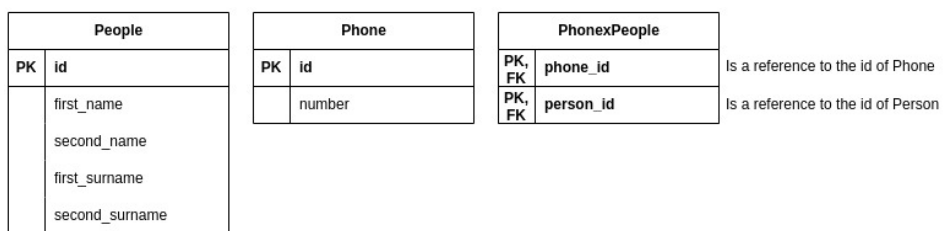
Modelo Lógico:

Figura 4: Modelo lógico de las tablas en el esquema GE

Código:

```
CREATE TABLE GE.People (  
    id_people NUMBER,  
    first_name VARCHAR2(50),  
    second_name VARCHAR2(50),  
    first_surname VARCHAR2(50),  
    second_surname VARCHAR2(50)  
)  
TABLESPACE GE_Data  
STORAGE (  
    INITIAL 6144  
    NEXT 6144  
    MINEXTENTS 1  
    MAXEXTENTS 5  
);  
  
CREATE TABLE GE.Phone (  
    id_phone NUMBER,  
    phone_number VARCHAR2(20)  
)  
TABLESPACE GE_Data  
STORAGE (  
    INITIAL 6144  
    NEXT 6144  
    MINEXTENTS 1  
    MAXEXTENTS 5  
);  
  
CREATE TABLE GE.PhonexPeople (  
    phone_id NUMBER,  
    person_id NUMBER  
)  
TABLESPACE GE_Data  
STORAGE (  
    INITIAL 6144  
    NEXT 6144  
    MINEXTENTS 1  
    MAXEXTENTS 5  
);
```

Evidencia:

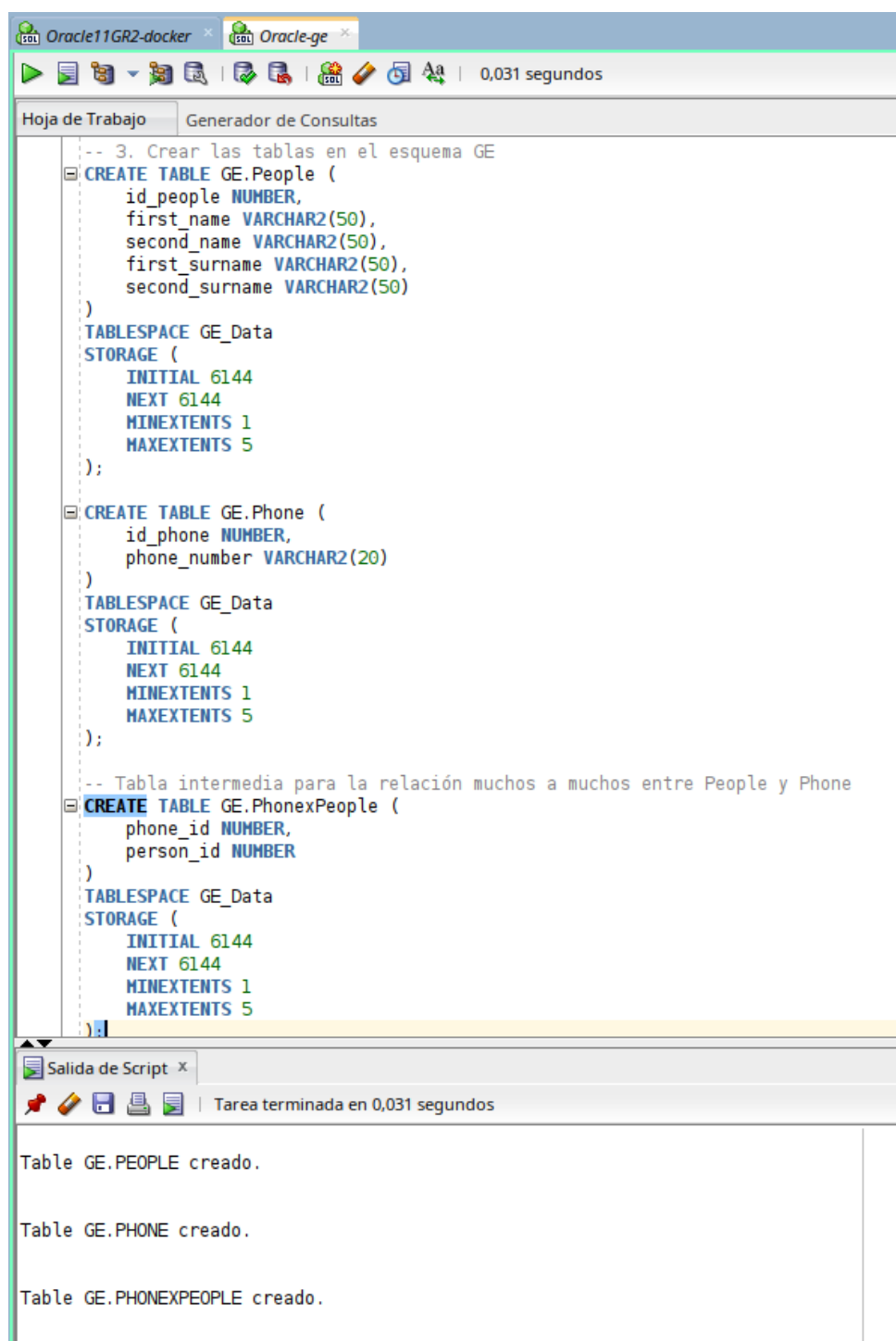


Figura 5: Evidencia de creación de tablas en GE

1.4. Creación de llaves primarias

Código:

```
ALTER TABLE GE.People ADD CONSTRAINT pk_people PRIMARY KEY (id_people)
USING INDEX
TABLESPACE GE_Index PCTFREE 20
STORAGE (
    INITIAL 10K
    NEXT 10K
    MINEXTENTS 1
    MAXEXTENTS UNLIMITED
    PCTINCREASE 0
);

ALTER TABLE GE.Phone ADD CONSTRAINT pk_phone PRIMARY KEY (id_phone)
USING INDEX
TABLESPACE GE_Index PCTFREE 20
STORAGE (
    INITIAL 10K
    NEXT 10K
    MINEXTENTS 1
    MAXEXTENTS UNLIMITED
    PCTINCREASE 0
);

ALTER TABLE GE.PhonexPeople ADD CONSTRAINT pk_phonexpeople PRIMARY KEY (phone_id,
    person_id);
```

Evidencia:

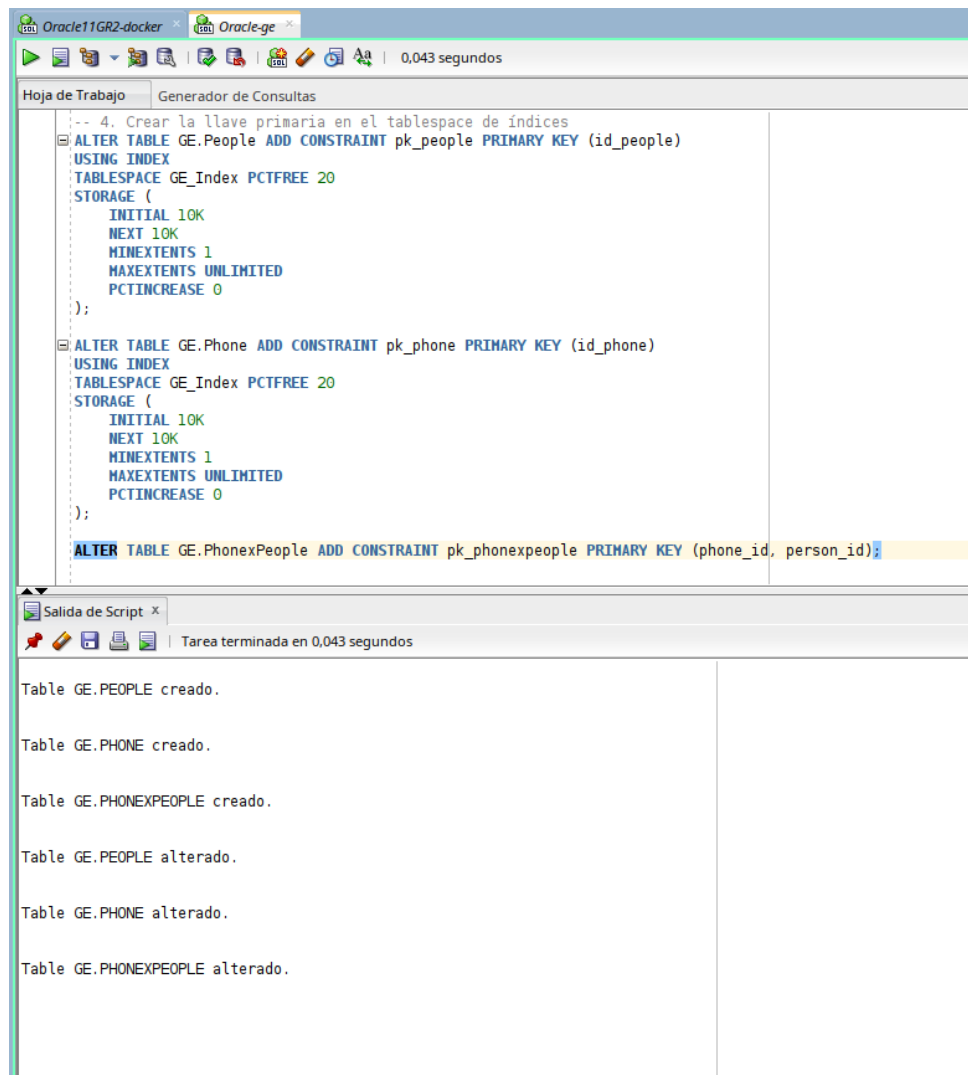


Figura 6: Evidencia de creación de llaves primarias

1.5. Creación de constraints

Código:

```
ALTER TABLE GE.People
  MODIFY first_name CONSTRAINT people_firstName_nn NOT NULL;

ALTER TABLE GE.Phone
  MODIFY phone_number CONSTRAINT phone_number_nn NOT NULL;

ALTER TABLE GE.PhonexPeople
  ADD CONSTRAINT fk_phonexpeople_phone
  FOREIGN KEY (phone_id)
  REFERENCES GE.Phone(id_phone);

ALTER TABLE GE.PhonexPeople
  ADD CONSTRAINT fk_phonexpeople_person
  FOREIGN KEY (person_id)
  REFERENCES GE.People(id_people);
```

Evidencia:

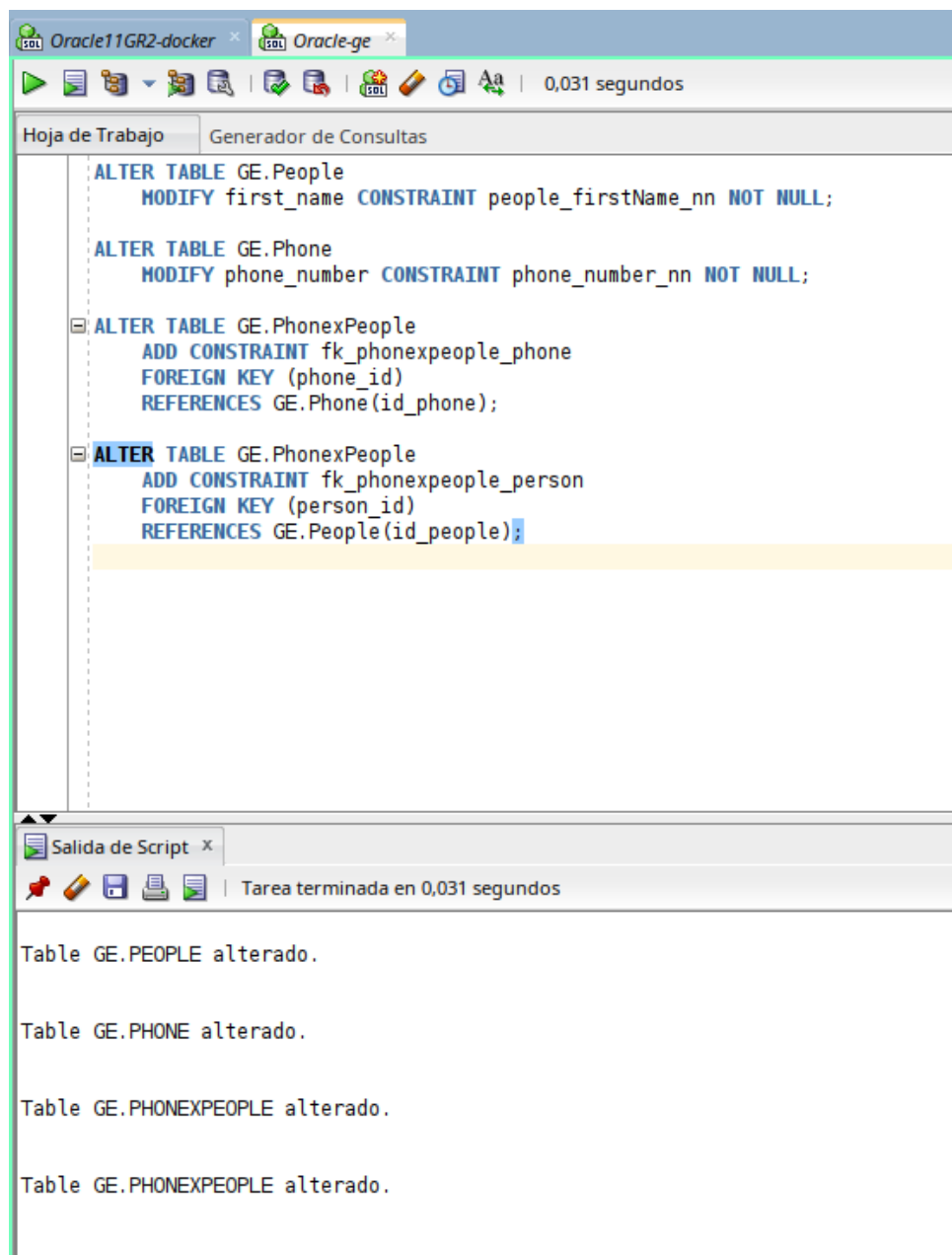


Figura 7: Evidencia de creación de constraints

1.6. Creación del Esquema AD

Código:

```
CREATE TABLESPACE AD_Data
  DATAFILE '/u01/app/oracle/oradata/XE/addata01.dbf'
  SIZE 10M
  REUSE
  AUTOEXTEND ON
  NEXT 512k
  MAXSIZE 200M;

CREATE TABLESPACE AD_Index
  DATAFILE '/u01/app/oracle/oradata/XE/adindex01.dbf'
  SIZE 10M
  REUSE
  AUTOEXTEND ON
  NEXT 512k
  MAXSIZE 200M;

CREATE USER AD
  IDENTIFIED BY ad
  DEFAULT TABLESPACE AD_Data
  QUOTA 10M ON AD_Data
  TEMPORARY TABLESPACE temp
  QUOTA 5M ON SYSTEM
  QUOTA 10M ON AD_Index;

GRANT CONNECT to AD;
GRANT CREATE SESSION to AD;
GRANT CREATE TABLE to AD;
```

Evidencia:

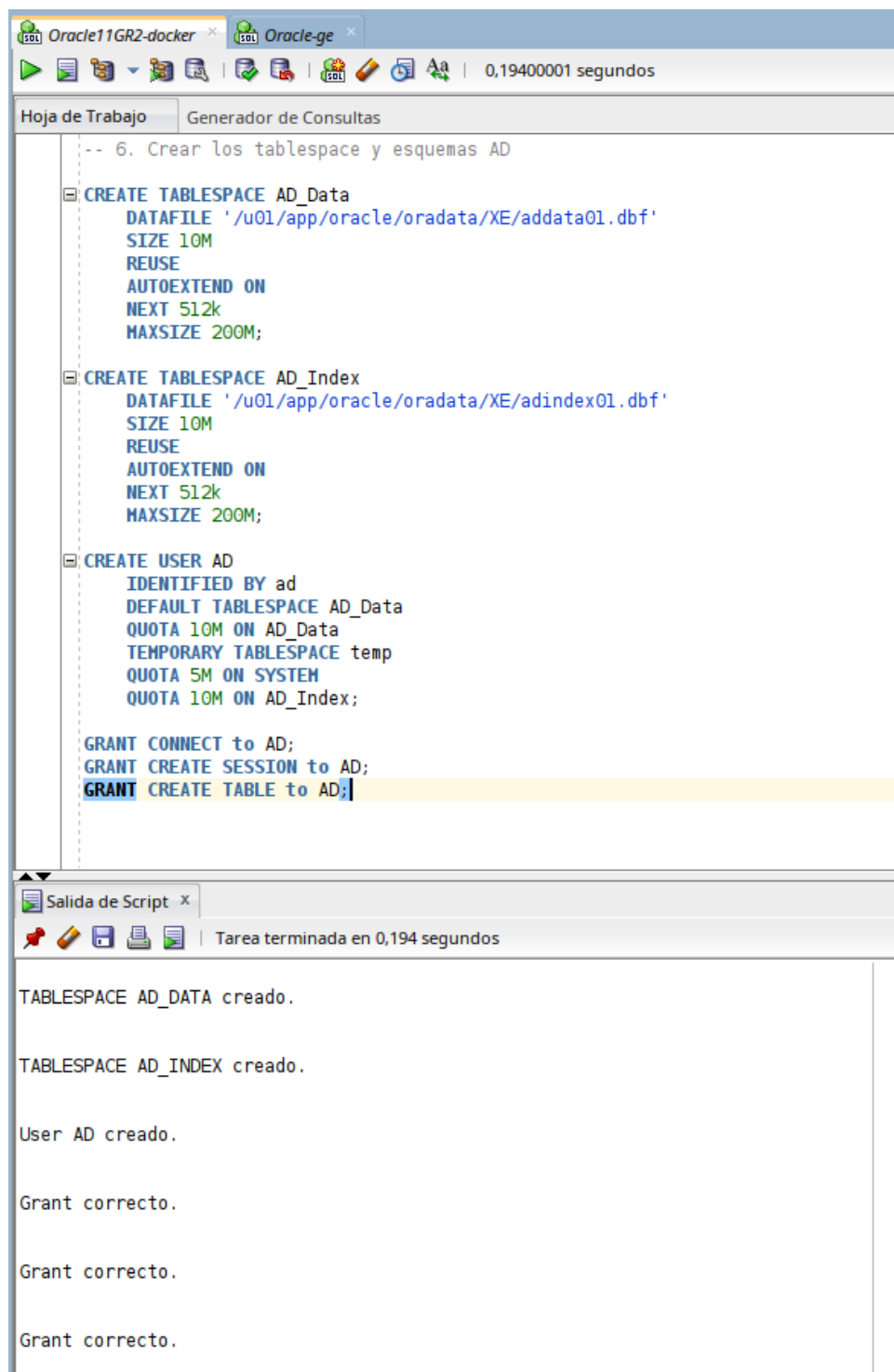


Figura 8: Evidencia de creación del esquema AD

1.7. Creación de tablas en AD según el requerimiento

Modelo Conceptual:

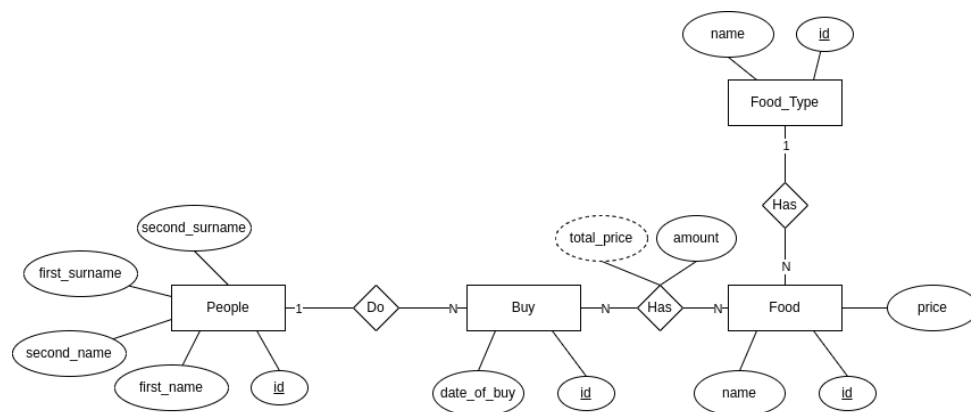


Figura 9: Modelo conceptual de las tablas en el esquema AD

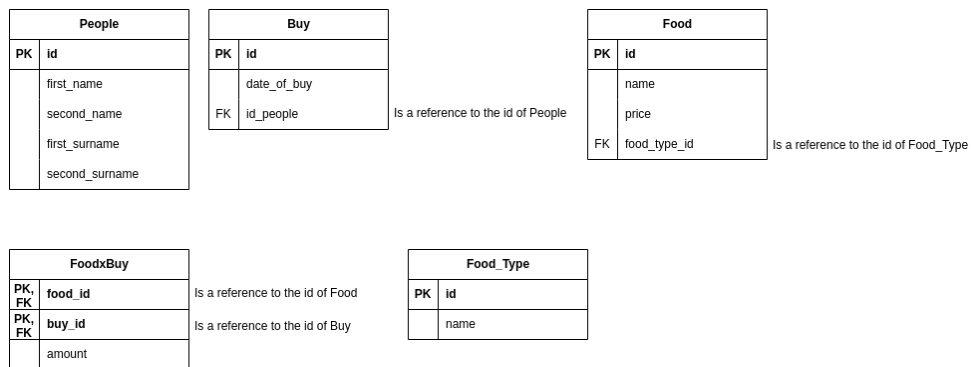
Modelo Lógico:

Figura 10: Modelo lógico de las tablas en el esquema AD

Código:

```

CREATE TABLE AD.Food_Type (
    food_type_id    NUMBER CONSTRAINT pk_Food_Type PRIMARY KEY
                   USING INDEX TABLESPACE AD_Index
                   STORAGE (
                       INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE
                       0
                   ),
    name            VARCHAR2(50) CONSTRAINT Food_Type_name_nn NOT NULL
) TABLESPACE AD_Data
  STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
  );

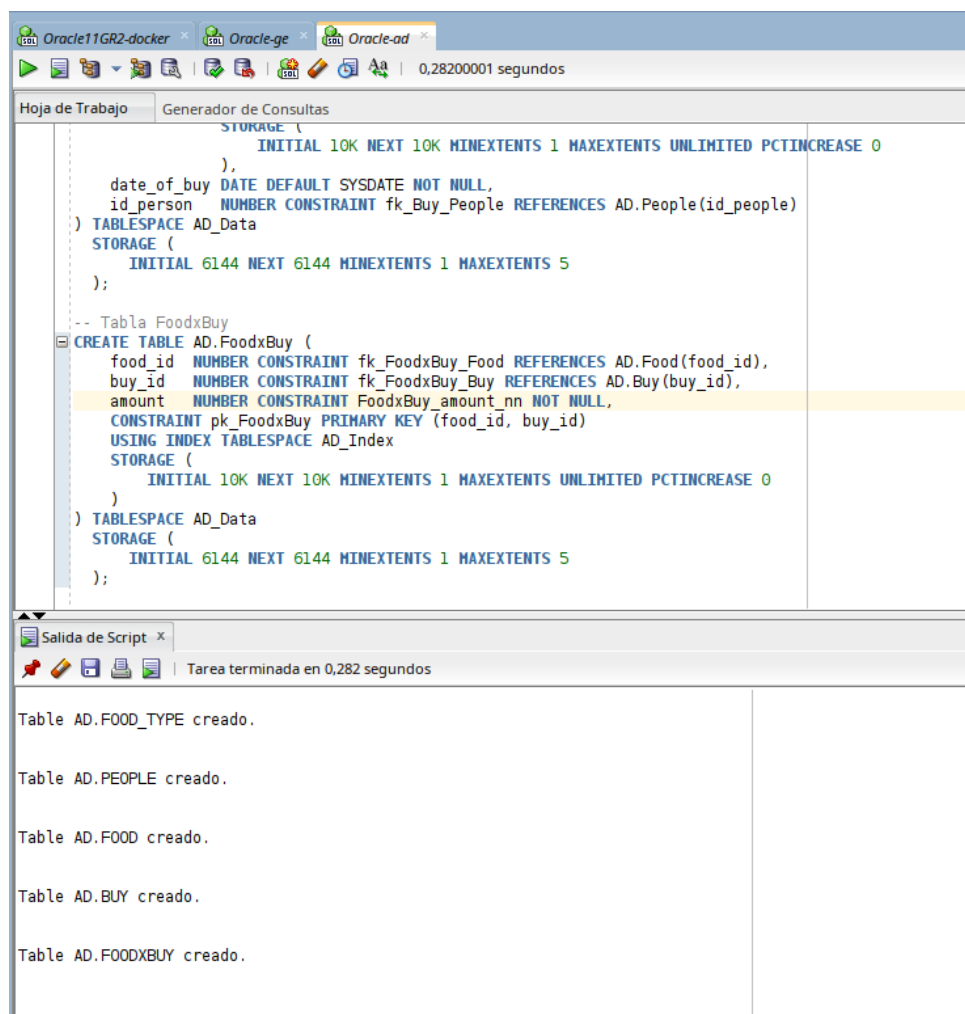
CREATE TABLE AD.People (
    id_people       NUMBER CONSTRAINT pk_People PRIMARY KEY
                   USING INDEX TABLESPACE AD_Index
                   STORAGE (
                       INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE
                       0
                   ),
    first_name      VARCHAR2(50) CONSTRAINT people_firstName_nn NOT NULL,
    second_name     VARCHAR2(50),
    first_surname   VARCHAR2(50) CONSTRAINT people_firstSurname_nn NOT NULL,
    second_surname  VARCHAR2(50)
) TABLESPACE AD_Data
  STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
  );

CREATE TABLE AD.Food (
    food_id         NUMBER CONSTRAINT pk_Food PRIMARY KEY
                   USING INDEX TABLESPACE AD_Index
                   STORAGE (
                       INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE
                       0
                   ),
    name            VARCHAR2(100) CONSTRAINT Food_name_nn NOT NULL,
    price           NUMBER(10,2) CONSTRAINT Food_price_chk CHECK (price > 0) NOT NULL,
    food_type_id    NUMBER CONSTRAINT fk_Food_Food_Type REFERENCES AD.Food_Type(
        food_type_id)
) TABLESPACE AD_Data
  STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
  );

CREATE TABLE AD.Buy (
    buy_id          NUMBER CONSTRAINT pk_Buy PRIMARY KEY
                   USING INDEX TABLESPACE AD_Index
                   STORAGE (
                       INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE 0
                   ),
    date_of_buy     DATE DEFAULT SYSDATE NOT NULL,
    id_person       NUMBER CONSTRAINT fk_Buy_People REFERENCES AD.People(id_people)
) TABLESPACE AD_Data
  STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
  );

CREATE TABLE AD.FoodxBuy (
    food_id         NUMBER CONSTRAINT fk_FoodxBuy_Food REFERENCES AD.Food(food_id),
    buy_id          NUMBER CONSTRAINT fk_FoodxBuy_Buy REFERENCES AD.Buy(buy_id),
    amount          NUMBER CONSTRAINT FoodxBuy_amount_nn NOT NULL,
    CONSTRAINT pk_FoodxBuy PRIMARY KEY (food_id, buy_id)
    USING INDEX TABLESPACE AD_Index
    STORAGE (
        INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE 0
    )
) TABLESPACE AD_Data
  STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
  );

```

Evidencia:

```
STORAGE (
    INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE 0
),
date_of_buy DATE DEFAULT SYSDATE NOT NULL,
id_person NUMBER CONSTRAINT fk_Buy_People REFERENCES AD.People(id_people)
) TABLESPACE AD_Data
STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
);

-- Tabla FoodxBuy
CREATE TABLE AD.FoodxBuy (
    food_id NUMBER CONSTRAINT fk_FoodxBuy_Food REFERENCES AD.Food(food_id),
    buy_id NUMBER CONSTRAINT fk_FoodxBuy_Buy REFERENCES AD.Buy(buy_id),
    amount NUMBER CONSTRAINT FoodxBuy_amount_nn NOT NULL,
    CONSTRAINT pk_FoodxBuy PRIMARY KEY (food_id, buy_id)
    USING INDEX TABLESPACE AD_Index
    STORAGE (
        INITIAL 10K NEXT 10K MINEXTENTS 1 MAXEXTENTS UNLIMITED PCTINCREASE 0
    )
) TABLESPACE AD_Data
STORAGE (
    INITIAL 6144 NEXT 6144 MINEXTENTS 1 MAXEXTENTS 5
);
```

Salida de Script x

Tarea terminada en 0,282 segundos

Table AD.FOOD_TYPE creado.

Table AD.PEOPLE creado.

Table AD.FOOD creado.

Table AD.BUY creado.

Table AD.FOODXBUY creado.

Figura 11: Evidencia de creación de tablas en AD

1.8. Renombrar columnas en una tabla

Código:

```
ALTER TABLE AD.Food RENAME COLUMN name TO food_name;  
ALTER TABLE AD.Food RENAME COLUMN price TO food_price;
```

Evidencia:

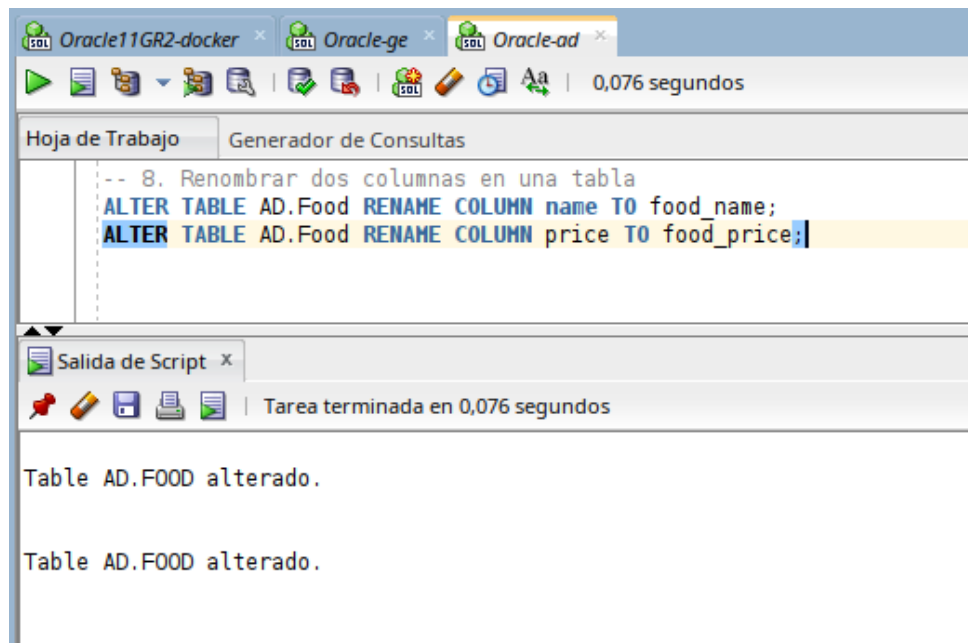


Figura 12: Evidencia de renombrado de columnas

1.9. Ampliación del tamaño de nombre en People

Código:

```
ALTER TABLE AD.People MODIFY first_name VARCHAR2(100);
```

Evidencia:

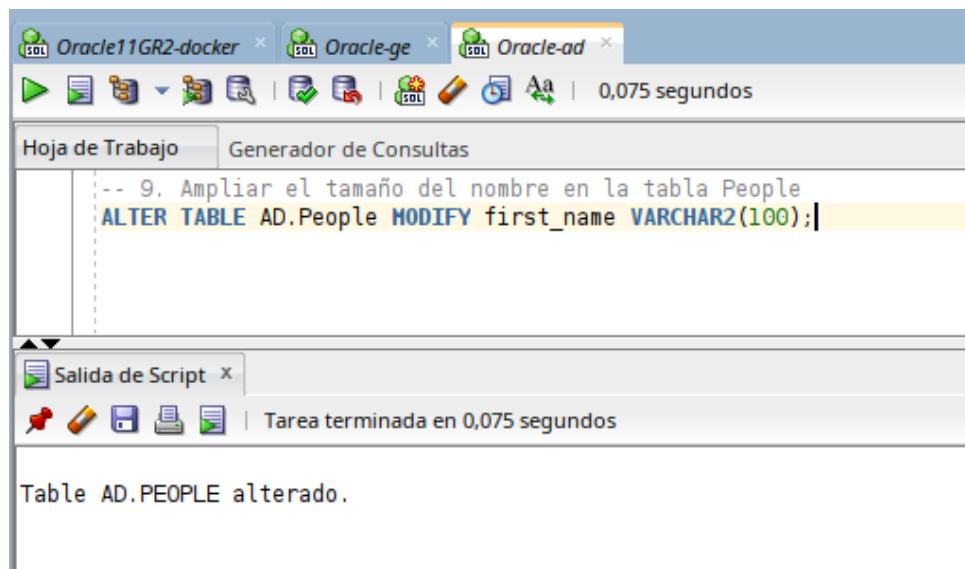


Figura 13: Evidencia de ampliación de tamaño de nombre en People