



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université des Sciences et de la Technologie Houari Boumediene

Faculté d'Electronique et d'Informatique

Département Informatique

Spécialité : Systèmes d'Information Intelligent.

TP ABDD N°4

NOM: AISSAT

Prénom: Abdelhak

Matricule: 171733032123

2.

1. Ajouter la contrainte

```
ALTER TABLE Sportifs ADD CONSTRAINT ck_Sportifs_age_20_45 CHECK( (age > 20) AND  
(age < 45) );
```

2. Procédure qui augmente l'âge de chaque sportif de 5 ans

```
CREATE OR REPLACE Procedure Maj_age_plus5 AS  
CURSOR cr is SELECT idsportif, nom, prenom, age FROM Sportifs;  
newage sportifs.age%TYPE;  
BEGIN  
FOR item in cr  
LOOP  
newage := item.age + 5 ;  
UPDATE Sportifs SET age = newage  
WHERE idsportif = item.idsportif;  
dbms_output.put_line('l'age du sportif '||item.nom||' est passer de '||item.age||'  
a '||newage);  
END LOOP;  
END ;  
/
```

3. Désactiver contrainte et effectuer Mise a jour

```
ALTER TABLE Sportifs DISABLE CONSTRAINT ck_Sportifs_age_20_45 ;
```

```
l'age du sportif IGHOLI est passer de 30 a 35
l'age du sportif GUEMEZ est passer de 30 a 35
l'age du sportif LECOM est passer de 30 a 35
l'age du sportif HOUAT est passer de 30 a 35
l'age du sportif BEQUETA est passer de 30 a 35
l'age du sportif RATENI est passer de 30 a 35
l'age du sportif TOUAT est passer de 30 a 35
l'age du sportif JALONI est passer de 30 a 35
l'age du sportif DEBOUBA est passer de 30 a 35
l'age du sportif GASTAB est passer de 30 a 35
l'age du sportif GIRONI est passer de 30 a 35
l'age du sportif DABONI est passer de 30 a 35
l'age du sportif LACHOUBI est passer de 30 a 35
l'age du sportif GALLOI est passer de 30 a 35
l'age du sportif DORONI est passer de 30 a 35
l'age du sportif LENOUCHI est passer de 30 a 35
l'age du sportif LERICHE est passer de 30 a 35
l'age du sportif MANSOUR est passer de 30 a 35
l'age du sportif LABOULAIS est passer de 26 a 31
l'age du sportif DOUDOU est passer de 26 a 31
l'age du sportif MAALEM est passer de 26 a 31
l'age du sportif BESNARD est passer de 26 a 31
l'age du sportif BELHAMID est passer de 26 a 31
l'age du sportif BOUAAZA est passer de 26 a 31
l'age du sportif CORCHI est passer de 26 a 31
l'age du sportif BELAID est passer de 26 a 31
l'age du sportif GASMI est passer de 26 a 31
l'age du sportif LAAMARA est passer de 25 a 30
l'age du sportif DABOUB est passer de 25 a 30
l'age du sportif HASSINI est passer de 25 a 30
l'age du sportif KALOUNE est passer de 27 a 32
l'age du sportif BELHAOUA est passer de 27 a 32
l'age du sportif BELAID est passer de 25 a 30
l'age du sportif HENDI est passer de 25 a 30

PL/SQL procedure successfully completed.
```

3. Pour chaque gymnase donner par jour d'ouverture les horaires des premières et dernières séances pour les gymnases de superficie > 400 m²:

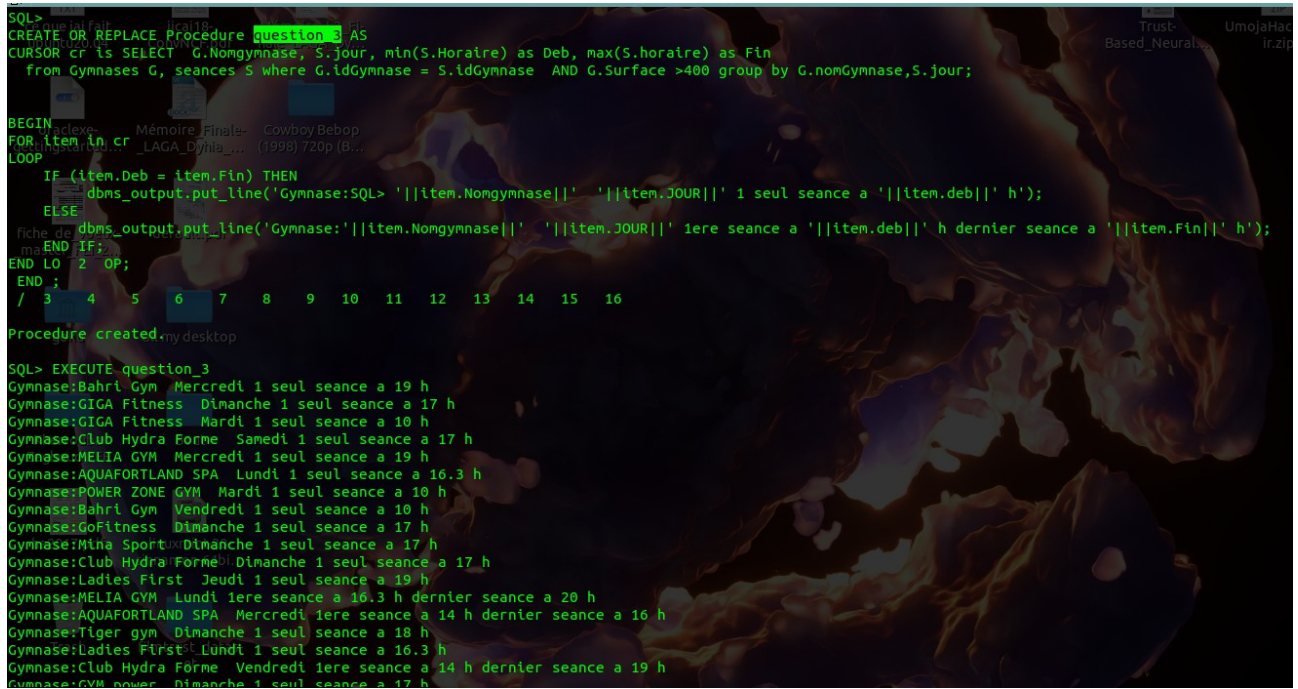
```
CREATE OR REPLACE Procedure question_3 AS
CURSOR cr is SELECT G.Nomgymnase, S.jour, min(S.Horaire) as Deb,
max(S.horaire) as Fin
from Gymnases G, seances S where G.idGymnase = S.idGymnase AND
G.Surface >400 group by G.nomGymnase,S.jour;

BEGIN
FOR item in cr
LOOP
IF (item.Deb = item.Fin) THEN
dbms_output.put_line('Gymnase:'||item.Nomgymnase||' '||item.JOUR||' 1 seul
seance a '||item.deb||' h');
ELSE
```

```

dbms_output.put_line('Gymnase:'||item.Nomgymnase||' '||item.JOUR||' 1ere
seance a '||item.deb||' h dernier seance a '||item.Fin||' h');
END IF;
END LOOP;
END ;
/

```



```

SQL>
CREATE OR REPLACE Procedure question_3 AS
CURSOR cr is SELECT G.Nomgymnase, S.jour, min(S.Horaire) as Deb, max(S.horaire) as Fin
from Gymnases G, seances S where G.idGymnase = S.idGymnase AND G.Surface >400 group by G.nomGymnase,S.jour;

BEGIN
FOR item IN cr
LOOP
IF (item.Deb = item.Fin) THEN
dbms_output.put_line('Gymnase:SQL> '||item.Nomgymnase||' '||item.JOUR||' 1 seul seance a '||item.deb||' h');
ELSE
dbms_output.put_line('Gymnase:SQL> '||item.Nomgymnase||' '||item.JOUR||' 1ere seance a '||item.deb||' h dernier seance a '||item.Fin||' h');
END IF;
END LOOP;
END ;
/

```

Procedure created.

```

SQL> EXECUTE question_3
Gymnase:Bahri Gym Mercredi 1 seul seance a 19 h
Gymnase:GIGA Fitness Dimanche 1 seul seance a 17 h
Gymnase:GIGA Fitness Mardi 1 seul seance a 10 h
Gymnase:Club Hydra Forme Samedi 1 seul seance a 17 h
Gymnase:MELIA GYM Mercredi 1 seul seance a 19 h
Gymnase:AQUAFORTLAND SPA Lundi 1 seul seance a 16.3 h
Gymnase:POWER ZONE GYM Mardi 1 seul seance a 10 h
Gymnase:Bahri Gym Vendredi 1 seul seance a 10 h
Gymnase:GoFitness Dimanche 1 seul seance a 17 h
Gymnase:Mtna Sport Dimanche 1 seul seance a 17 h
Gymnase:Club Hydra Forme Dimanche 1 seul seance a 17 h
Gymnase:Ladies First Jeudi 1 seul seance a 19 h
Gymnase:MELIA GYM Lundi 1ere seance a 16.3 h dernier seance a 20 h
Gymnase:AQUAFORTLAND SPA Mercredi 1ere seance a 14 h dernier seance a 16 h
Gymnase:Tiger gym Dimanche 1 seul seance a 18 h
Gymnase:Ladies First Lundi 1 seul seance a 16.3 h
Gymnase:Club Hydra Forme Vendredi 1ere seance a 14 h dernier seance a 19 h
Gymnase:GYM power Dimanche 1 seul seance a 17 h

```

4. Fonction qui retourne pour chaque sportif donné, le nombre de sport qu'il entraîne:

```

CREATE or REPLACE FUNCTION question4(id in number) return number IS
n number ;
BEGIN
SELECT COUNT(E.idSport) INTO n FROM ENTRAINEUR E WHERE
IDSPORTIFENTRAINEUR=id ;
return n;
END;
/

```

```

SQL> CREATE or REPLACE FUNCTION question4(id in number) return number IS
n number ;
BEGIN
SELECT COUNT(E.idSport) INTO n FROM ENTRAINEUR E WHERE IDSPORTIFENTRAINEUR=id ;
return n;
END;
/ 2 3 4 5 6 7
Function created.
SQL> DECLARE
CURSOR cr IS SELECT nom, prenom, idSportif FROM Sportifs WHERE
idSportif >= 1 AND idSportif <= 10;
nba number;
BEGIN
FOR item in cr
LOOP
dbms_output.put_line('Le sportif '||item.nom||' '||item.prenom||' entraine '||
question4(item.idSportif)||' sports !');
END LOOP;
EXCEPTION WHEN NO_DATA_FOUND THEN dbms_output.put_line('pas de donnee trouve');
CLOSE cr;
END;
/ 3 4 5 6 7 8p 9 10 11 12 13 14
Le sportif BOUTAHAR Abderahim entraine 5 sports !
Le sportif BOUROUBI Anis entraine 8 sports !
Le sportif BOUZIDI Amel entraine 4 sports !
Le sportif LACHEMI Bouzid entraine 3 sports !
Le sportif AAKOUB Linda entraine 0 sports !
Le sportif ABBAS Sophia entraine 3 sports !
Le sportif HADJ Zouhir entraine 4 sports !
Le sportif HAMADI Hani entraine 0 sports !
Le sportif ABDELMOUMEN Nadia entraine 0 sports !
Le sportif ABAD Abdelhamid entraine 0 sports !
PL/SQL procedure successfully completed.

```

5. Procédure qui permet d'ajouter une séance a partir de tous ces attributs en vérifiant l'unicité de la clé primaire et l'existence des clé étrangère

```

CREATE OR REPLACE Procedure question5 (idgym in number, idsp in number,
isportifentrain in number, jr in VARCHAR2,
hor in FLOAT, dur in NUMBER) IS

```

```

CURSOR cr_Pk_verif IS SELECT * FROM Seances WHERE
IDGYMNASE = idgym AND IDSPORT = idsp AND
isportifentrain = IDSPORTIFENTRAINEUR AND
JOUR = jr ;

```

```
CURSOR cr_Fk_idgym IS SELECT IDGYMNASE FROM GYMNASES WHERE  
IDGYMNASE=idgym;  
CURSOR cr_Fk_sport IS SELECT IDSPORT FROM SPORTS WHERE IDSPORT=idsp;  
CURSOR cr_Fk_sportif IS SELECT IDSPORTIF FROM SPORTIFS WHERE  
IDSPORTIF=isportifentrain;  
Pk_exist EXCEPTION;  
Fk_idgym_exist EXCEPTION;  
Fk_sport_exist EXCEPTION;  
Fk_sportif_exist EXCEPTION;  
i NUMBER;
```

```
BEGIN  
i:=0;  
--verifier clés primaire  
FOR item in cr_Pk_verif  
LOOP  
i:=i+1;  
END LOOP;  
IF (i > 0) THEN RAISE Pk_exist;  
END IF;  
-- verifier cle etrangere  
--! 1 IDGYMNASE dans GYMNASES (IDGYMNASE)
```

```
FOR item in cr_Fk_idgym  
LOOP  
i:=i+1;  
END LOOP;  
IF (i=0) THEN RAISE Fk_idgym_exist;  
END IF;
```

```
--* (IDSPORT) dans SPORTS (IDSPORT)  
FOR item in cr_Fk_sport  
LOOP  
i:=i+1;  
END LOOP;  
IF (i=0) THEN RAISE Fk_sport_exist;  
END IF;
```

```
--* (IDSPORTIFENTRAINEUR) dans SPORTIFS (IDSPORTIF)  
FOR item in cr_Fk_sportif  
LOOP  
i:=i+1;
```

```

END LOOP;
IF (i=0) THEN RAISE Fk_sportif_exist;
END IF;
INSERT INTO SEANCES VALUES (idgym,idsp,
isportifentrain, jr,
hor ,dur );

EXCEPTION
WHEN Pk_exist THEN dbms_output.put_line('Erreur Cle primaire deja utilisé');
WHEN Fk_idgym_exist THEN dbms_output.put_line('Erreur! table GYMNASSE cle
etrangere inexistante');
WHEN Fk_sport_exist THEN dbms_output.put_line('Erreur! table sport cle
etrangere inexistante');
WHEN Fk_sportif_exist THEN dbms_output.put_line('Erreur! table SPORTIFS cle
etrangere inexistante');
END;
/

```

```

Procedure created.
SQL> SET SERVEROUTPUT ON
SQL> EXECUTE question5(17,2,57,'Dimanche',17.0,60);
Erreur Cle primaire deja utilisé
PL/SQL procedure successfully completed.
SQL>

```