***Rapport TP N°3***

***BD2***

***b14d35***

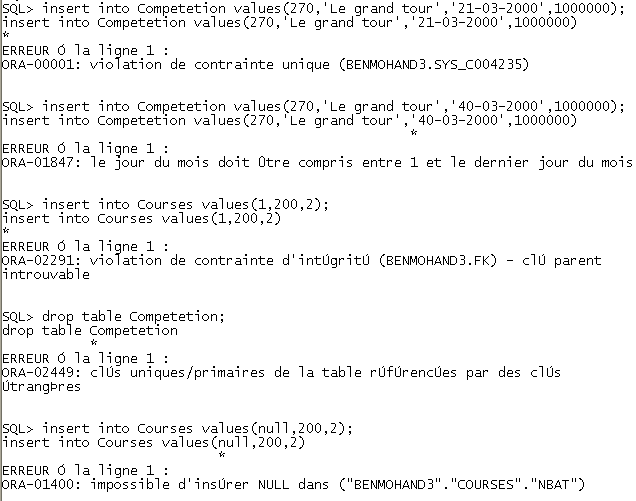
***1337***

***ISIL B (L3)***

***G 02***

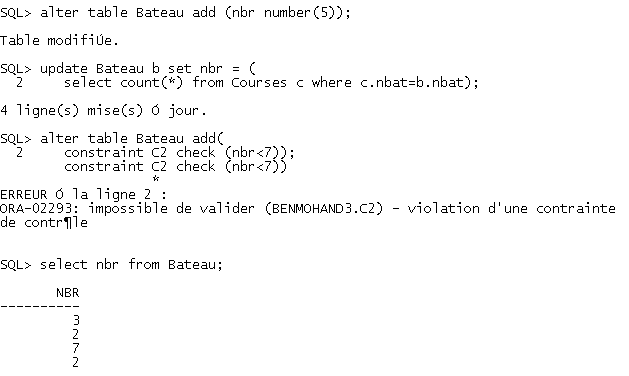
***Question 1 :***

|  |
| --- |
| insert into Competetion values(270,'Le grand tour','21-03-2000',1000000);  insert into Competetion values(270,'Le grand tour','40-03-2000',1000000);  insert into Courses values(1,200,2);  drop table Competetion;  insert into Courses values(null,200,2); |

******

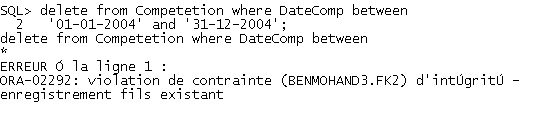
***Question 2 :***

|  |
| --- |
| alter table Bateau add (nbr number(5));  update Bateau b set nbr = (  select count(\*) from Courses c where c.nbat=b.nbat);  alter table Bateau add(  constraint C2 check (nbr<7));  select nbr from Bateau; |

******

***Question 3 :***

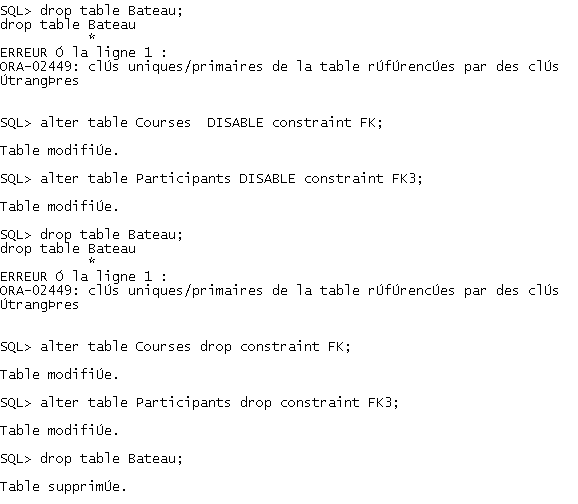
|  |
| --- |
| delete from Competetion where DateComp between  '01-01-2004' and '31-12-2004'; |

******

On doit supprimer les contraintes d’intégrités qui sont les clés étrangères de la table Courses qui référencé a celle de Competetion. Ou bien on supprime tout les tuples de Courses qui ayant référence à une course ayant passé en 2004.

***Question 4 :***

|  |
| --- |
| drop table Bateau;  alter table Courses DISABLE constraint FK;  alter table Participants DISABLE constraint FK3;  drop table Bateau;  alter table Courses drop constraint FK;  alter table Participants drop constraint FK3;  drop table Bateau; |

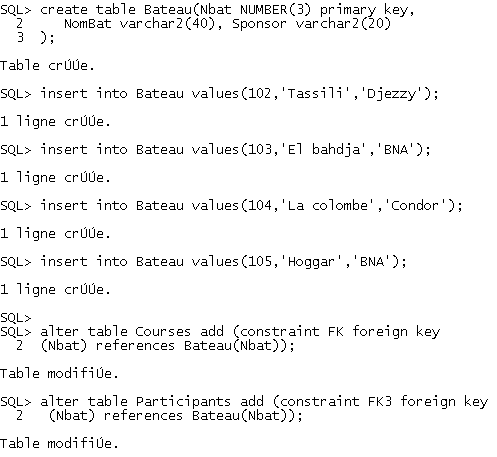
******

\*La suppression n’est pas possible car elle est référencie pas des clés étrangères.

\*La suppression n’est possible que quand les contraintes sont supprimées.

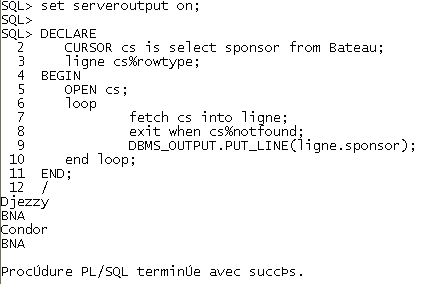
***Question 5 :***

|  |
| --- |
| create table Bateau(Nbat NUMBER(3) primary key,  NomBat varchar2(40), Sponsor varchar2(20)  );  insert into Bateau values(102,'Tassili','Djezzy');  insert into Bateau values(103,'El bahdja','BNA');  insert into Bateau values(104,'La colombe','Condor');  insert into Bateau values(105,'Hoggar','BNA');  alter table Courses add (constraint FK foreign key  (Nbat) references Bateau(Nbat));  alter table Participants add (constraint FK3 foreign key  (Nbat) references Bateau(Nbat)); |

******

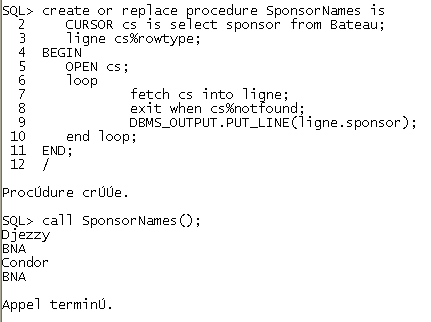
***Question 6 :***

|  |
| --- |
| set serveroutput on;  DECLARE  CURSOR cs is select sponsor from Bateau;  ligne cs%rowtype;  BEGIN  OPEN cs;  loop  fetch cs into ligne;  exit when cs%notfound;  DBMS\_OUTPUT.PUT\_LINE(ligne.sponsor);  end loop;  END;  / |

******

***Question 7 :***

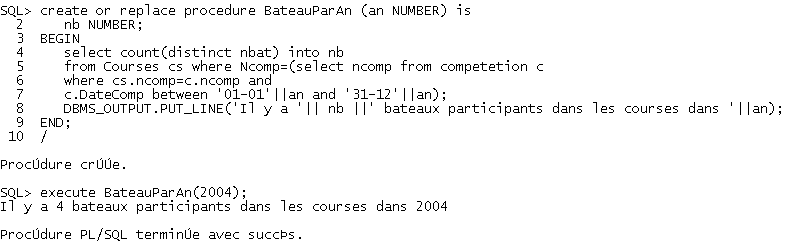
|  |
| --- |
| create or replace procedure SponsorNames is  CURSOR cs is select sponsor from Bateau;  ligne cs%rowtype;  BEGIN  OPEN cs;  loop  fetch cs into ligne;  exit when cs%notfound;  DBMS\_OUTPUT.PUT\_LINE(ligne.sponsor);  end loop;  END;  /  call SponsorNames(); |

******

La différence entre le programme et la procédure est que le programme on l’écrit à chaque fois, mais la procédure on écrit son code une fois puis on fait juste des appels.

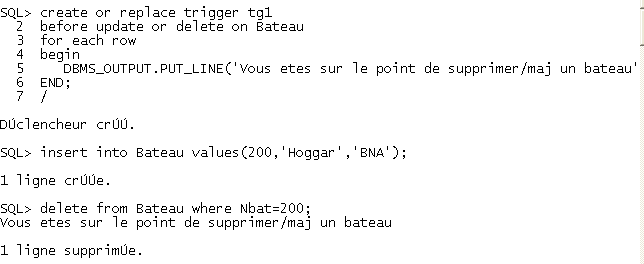
***Question 8 :***

|  |
| --- |
| create or replace procedure BateauParAn (an NUMBER) is  nb NUMBER;  BEGIN  select count(distinct nbat) into nb  from Courses cs where Ncomp=(select ncomp from competetion c  where cs.ncomp=c.ncomp and  c.DateComp between '01-01'||an and '31-12'||an);  DBMS\_OUTPUT.PUT\_LINE('Il y a '|| nb ||' bateaux participants dans les courses dans '||an);  END;  /  execute BateauParAn(2004); |

******

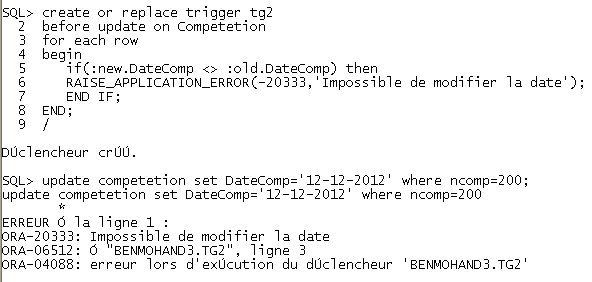
***Question 9 :***

|  |
| --- |
| create or replace trigger tg1  before update or delete on Bateau  for each row  begin  DBMS\_OUTPUT.PUT\_LINE('Vous etes sur le point de supprimer/maj un bateau');  END;  /  insert into Bateau values(200,'Hoggar','BNA');  delete from Bateau where Nbat=200; |

******

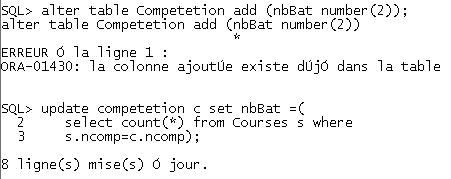
***Question 10 :***

|  |
| --- |
| create or replace trigger tg2  before update on Competetion  for each row  begin  if(:new.DateComp <> :old.DateComp) then  RAISE\_APPLICATION\_ERROR(-20333,'Impossible de modifier la date');  END IF;  END;  /  update competetion set DateComp='12-12-2012' where ncomp=200; |

******

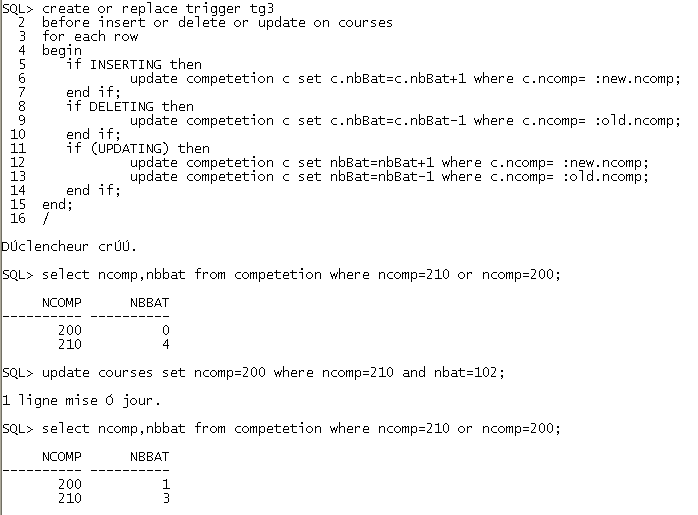
***Question 11 :***

|  |
| --- |
| alter table Competetion add (nbBat number(2));  update competetion c set nbBat =(  select count(\*) from Courses s where  s.ncomp=c.ncomp); |

******

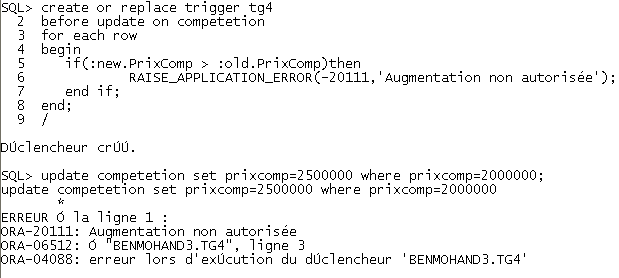
***Question 12 :***

|  |
| --- |
| create or replace trigger tg3  before insert or delete or update on courses  for each row  begin  if INSERTING then  update competetion c set c.nbBat=c.nbBat+1 where c.ncomp= :new.ncomp;  end if;  if DELETING then  update competetion c set c.nbBat=c.nbBat-1 where c.ncomp= :old.ncomp;  end if;  if (UPDATING) then  update competetion c set nbBat=nbBat+1 where c.ncomp= :new.ncomp;  update competetion c set nbBat=nbBat-1 where c.ncomp= :old.ncomp;  end if;  end;  /  select ncomp,nbbat from competetion where ncomp=210 or ncomp=200;  update courses set ncomp=200 where ncomp=210 and nbat=102;  select ncomp,nbbat from competetion where ncomp=210 or ncomp=200; |

******

***Question 13 :***

|  |
| --- |
| create or replace trigger tg4  before update on competetion  for each row  begin  if(:new.PrixComp > :old.PrixComp)then  RAISE\_APPLICATION\_ERROR(-20111,'Augmentation non autorisée');  end if;  end;  /  update competetion set prixcomp=2500000 where prixcomp=2000000; |

******