

```

CREATE OR REPLACE PROCEDURE ajout_proj (pn_projet projet.n_projet%TYPE,
    pnom_projet projet.nom_projet%TYPE,
    pn_equipe projet.n_equipe%TYPE,
    pn_cher_resp projet.n_cher_resp%TYPE,
    pspecialite specialite.specialite%TYPE,
    retour OUT NUMBER) IS

    CURSOR c1 IS SELECT n_chercheur, nom_chercheur FROM chercheur
    WHERE specialite=pspecialite AND n_equipe=pn_equipe;

    mauvaise_equ_cher_resp EXCEPTION;
    pas_chercheur_dans_specialite EXCEPTION;
    enfant_sans_parent EXCEPTION;
    PRAGMA EXCEPTION_INIT(enfant_sans_parent,-2291);

    spec specialite.nom_specialite%TYPE;

    mess VARCHAR(80);

BEGIN

    SELECT nom_specialite INTO spec FROM specialite s
    WHERE s.specialite=pspecialite;

    INSERT INTO projet VALUES
    (pn_projet,pnom_projet,pn_equipe,pn_cher_resp,0,0);
    INSERT INTO compte_rendu VALUES ('enregistrement du projet
    '||pnom_projet||' dans PROJET');

    FOR c1_rec IN c1 LOOP
        INSERT INTO travailler (n_projet, n_chercheur) VALUES
    (pn_projet,c1_rec.n_chercheur);
        INSERT INTO compte_rendu
        VALUES ('Affectation du chercheur
    '||c1_rec.nom_chercheur||' a ce projet');
    END LOOP;

    INSERT INTO compte_rendu VALUES ('Transaction reussie');
    retour:=0;
    COMMIT;

EXCEPTION

    WHEN enfant_sans_parent THEN
        rollback;
        mess:=substr(sqlerrm,1,80);
        if mess LIKE '%FK_PROJET_EQ%' then
            INSERT INTO compte_rendu VALUES ('equipe responsable inconnue');
        end if;

```

```

        if mess LIKE '%FK_PROJET_CHER%' then
            INSERT INTO compte_rendu VALUES ('chercheur responsable inconnu');
        end if;
        retour:=1;

    WHEN no_data_found THEN
        rollback;
        INSERT INTO compte_rendu VALUES ('specialite inconnue');
        retour:=1;

    WHEN mauvaise_equ_cher_resp then
        rollback;
        insert into compte_rendu values ('le chercheur responsable
n'appartient pas a l'equipe');
        retour:=1;

    WHEN pas_chercheur_dans_specialite then
        rollback;
        insert into compte_rendu values ('aucun chercheur de cette equipe
n'appartient a la specialite');
        retour:=1;

    WHEN dup_val_on_index THEN
        rollback;
        INSERT INTO compte_rendu VALUES ('numero de projet deja present');
        retour:=1;

    WHEN others THEN
        rollback;
        mess:=substr(sqlerrm,1,80);
        insert into compte_rendu values (mess);
        retour:=1;

END ajout_proj;
/

```

```

CREATE OR REPLACE PROCEDURE fin_projet (pn_projet projet.n_projet%TYPE, retour
OUT NUMBER) IS

```

```

    pas_chercheur_dans_travailler EXCEPTION;
    nom_proj projet.nom_projet%TYPE;
    mess VARCHAR (80);

```

```

    CURSOR c1 is SELECT t.n_chercheur, c.nom_chercheur from travailler t,
chercheur c
    where t.n_projet=pn_projet
    and t.n_chercheur=c.n_chercheur;

```

```

BEGIN

```

```
SELECT nom_projet INTO nom_proj FROM projet p
WHERE p.n_projet=pn_projet;
```

```
FOR c1_rec IN c1 LOOP
```

```
    DELETE from travailler t
    where t.n_projet = pn_projet
    and t.n_chercheur=c1_rec.n_chercheur;
```

```
    INSERT into compte_rendu
    values ('suppression du travail de ' ||c1_rec.nom_chercheur);
```

```
END LOOP;
```

```
DELETE from projet p
where p.n_projet=pn_projet;
```

```
INSERT into compte_rendu
values ('suppression du projet ' ||nom_proj);
```

```
COMMIT;
retour:=0;
```

```
EXCEPTION
```

```
    When pas_chercheur_dans_travailler then
        rollback;
        insert into compte_rendu values ('pas de chercheur affecté a ce
projet');
        retour:=1;
```

```
    when no_data_found then
        rollback;
        insert into compte_rendu values ('numero de projet inconnu');
        retour:=1;
```

```
    when others then
        rollback;
        mess:=substr(sqlerrm,1,80);
        insert into compte_rendu values (mess);
        retour:=1;
```

```
END fin_projet;
```

```
/
```