PL SQL Assignment 4

Roll No: U18CO021

University database has a table named as "Students" which stores the details as follows:

ID	Name	Age	Stream	Marks
1	Arun	21	BSc	55
2	Nisarg	22	BTech	65
3	Aarav	24	MTech	70
4	Dhairya	27	PhD	75
5	Dhruv	23	MCA	65
6	Niraj	22	MCom	74
7	Kevin	24	MCA	78

Create Table and Insert Values

```
create table students (
  id number primary key,
  name varchar(50),
  age number,
  stream varchar(20),
  marks number
);
insert into students values(1, 'Arun', 21, 'BSc', 55);
insert into students values(2, 'Nisarg', 22, 'BTech', 65);
insert into students values(3, 'Aarav', 24, 'Mtech', 70);
insert into students values(4, 'Dhariya', 27, 'PhD', 75);
insert into students values(5, 'Dhruv', 23, 'MCA', 65);
insert into students values(6, 'Niraj', 22, 'MCom', 74);
insert into students values(7, 'Kevin', 24, 'MCA', 78);
```

1) Create a PL/SQL trigger that does not allow to insert details of a student whose age is greater than 30.After creation, try insertion of details of a student whose is 32. Next, insert a row with age= 26.Show your output for both the cases.

```
create or replace trigger age_listener
```

```
before insert on students
for each row
when (NEW.id > 0)
declare
  exp exception;
  pragma exception_init(exp, -2001);
begin
  if :NEW.age > 30
  then
    raise exp;
  end if;
end;
/
```

```
1   insert into students values(8, 'Kevin', 32, 'MCA', 78);
2
```

```
ORA-02001: user SYS is not permitted to create indexes with freelist groups ORA-
06512: at "SQL_QQDDVSGPUNLFDNPPCQTFJJYGZ.AGE_LISTENER", line 7
ORA-06512: at "SYS.DBMS_SQL", line 1721
```

```
1   insert into students values(8, 'Kevin', 26, 'MCA', 78);
2
```

```
1 row(s) inserted.
```

2)Create a PL/SQL trigger that keeps track of changes in marks of any student. After any updation in marks, display old marks,new marks and the difference.After the trigger has been successfully created, change the marks of Dhairya from75 to 80 and show the result.

```
create or replace trigger insert_listener
before update on students
for each row
when (NEW.id > 0)
declare
```

```
  diff number;
begin
  diff := :NEW.marks  - :OLD.marks;
  dbms_output.put_line('Old Marks: ' || :OLD.marks);
  dbms_output.put_line('New Marks: ' || :NEW.marks);
  dbms_output.put_line('Marks Difference: ' || diff);
end;
/
```

```
1   update students
2   set marks = 80
3   where name='Dhariya';


1 row(s) updated.
Old Marks: 75
New Marks: 80
Marks Difference: 5
```