Create a multiply function that accepts two numbers and returns their .1
.product

```
DROP FUNCTION
task1=# create function multiple  (x integer , y integer )
RETURNS integer AS $$
begin

return  x*y;
end;

$$ LANGUAGE plpgsql;
CREATE FUNCTION
task1=# select multiple (3 , 2);
 multiple
----------
        6
(1 row)

task1=#
```

Create a hello_world function that takes a name as input and returns a .2
.personalized welcome message for that name

```
task1=# create  function hello (name varchar(50))
RETURNS varchar AS $$
begin

return concat('welcome' , name);
end;

$$ LANGUAGE plpgsql;
CREATE FUNCTION
task1=# select hello ('shady')
task1-# ;
    hello
--------------
 welcomeshady
(1 row)

task1=#
```

3. Create a function that accepts a number and determines whether it is odd or even.

```
task1=# create function odd_even (number int)
RETURNS char AS $$
BEGIN
    if number % 2 = 0 then
        RETURN 'Even';
    else
        RETURN 'Odd';
    end if;
end;
$$ LANGUAGE plpgsql
task1-# ;
CREATE FUNCTION
task1=# select odd_even(20);
 odd_even
----------
 Even
(1 row)

task1=#
```

4. Create a function that takes a Student ID as input and retrieves all information related to that student.

```
(1 row)

task1=# create function id_stu (number int)
RETURNS student AS $$
BEGIN
    return (select student from student where id=number);
end;
$$ LANGUAGE plpgsql
;
CREATE FUNCTION
task1=# select id_stu(2);
                       id_stu
-------------------------------------------------------
 (2,"Sara Hany",sara@iti.com,Giza,2,1990-08-23,Female)
(1 row)

task1=# select id_stu(1);
                        id_stu
-------------------------------------------------------
 (1,"Ahmed Shawky",ahmed@iti.com,Cairo,1,1995-01-10,Male)
(1 row)

task1=#
```

5. Implement a function that takes the name of a subject and calculates the average grades for that subject.

```
DROP FUNCTION
task1=# CREATE OR REPLACE FUNCTION subject_ave(subject_name text)
RETURNS float AS $$
BEGIN
  RETURN (
    SELECT AVG(grades.grade)
    FROM grades
    JOIN subject ON grades.sub_id = subject.id
    WHERE subject.sub_name = subject_name
  );
END;
$$ LANGUAGE plpgsql;
CREATE FUNCTION
task1=# select subject_ave('Python');
 subject_ave
-------------
          84
(1 row)

task1=# select subject_ave('HTML');
 subject_ave
-------------
          85
(1 row)

task1=#
```