**Functions Assignments**

1. Create a table student. It has four fields rollno, name, marks and grade.

Note – Grade column has be generated by a function get\_Grade(). This function takes marks as argument and generates the grade as per the following table

|  |  |
| --- | --- |
| **Marks Criteria** | **Grade** |
| Greater than or equal to 70 | Distinction |
| Between 60 and 69 | First Class |
| Between 50 and 59 | Second Class |
| Less than 50 | Failed. |

When the record is inserted then the grade value should be generated.

1. Create a function Raise\_Sal that will accept salary as a parameter and returns 15% raised salary. Use this function in the select statement of the emp table to see the result.
2. Create a function Yearly\_Raise that will take the salary, deptno and job as the parameters and raise the salary according to different criteria.

|  |  |
| --- | --- |
| **Criteria** | **Raise** |
| Clerk employees of deptno 20 earning salary above 1000 | 20% |
| Clerk employees of deptno 20 earning salary less 1000 | 15% |
| Clerk employees of deptno 20 earning salary above 1000 | 25% |
| Clerk employees of deptno 20 earning salary less than 1000 | 18% |
| Clerk employees of deptno 30 having any salary | 10% |

Use this function to update salaries of the employees of job Clerk in the table emp.

4. Run the following script.

Create Table Company\_Product (ProductId Varchar(20));

Create a function check\_productid that will take a productid (string) as a parameter. The function should check the last 4 characters of that productid. If they are KPIT (only upper case) then a new record must get created in the table Company\_Product else throw a relevant error message.

5. Considering the emp table create a function Last\_Employee which will take the job type as the parameter and display the last employee (s) joined in that job.

6. Considering the emp table create a function TopN\_Salary that will take the top Nth number as the parameter and returns the highest salary at that position.