- **Neo4j Connectivity from the python jupyter :**
- **Analysis & Data Aggregations**

- **1. Displaying the Data types :**

Srno	Python code
1	GRAPHDB.dtypes
Output	EmpNumber object EmpDepartment object EmpJobRole object EmpEnvironmentSatisfaction int64 EmpLastSalaryHikePercent int64 EmpWorkLifeBalance int64 ExperienceYearsAtThisCompany int64 ExperienceYearsInCurrentRole int64 YearsSinceLastPromotion int64 YearsWithCurrManager int64 PerformanceRating int64 dtype: object
2	GRAPHDB.dtypes.value_counts()
Output	int64 8 object 3 Name: count, dtype: int64
3	GRAPHDB.nunique()
Output	EmpNumber 1200 EmpDepartment 6 EmpJobRole 19 EmpEnvironmentSatisfaction 4 EmpLastSalaryHikePercent 15 EmpWorkLifeBalance 4 ExperienceYearsAtThisCompany 37 ExperienceYearsInCurrentRole 19 YearsSinceLastPromotion 16 YearsWithCurrManager 18 PerformanceRating 3 dtype: int64

1. Particular data of Columno. 0 : EmpNumber of Aggregation

Srno	Python code
1	# identifying the types of Employee Numbers GRAPHDB["EmpNumber"].unique()#Here displayed the columun'EmpNumber' of number of unique numbers
Output	array(['E1001000', 'E1001006', 'E1001007', ..., 'E100994', 'E100995', 'E100998'], dtype=object)
2	GRAPHDB["EmpNumber"].nunique()
Output	1200
3	EmpNumbers = GRAPHDB["EmpNumber"].unique() # I have saved it EmpNumbers
Output	array(['E1001000', 'E1001006', 'E1001007', ..., 'E100994', 'E100995', 'E100998'], dtype=object)
4	GRAPHDB.loc[:,"EmpNumber"].value_counts()
Output	EmpNumber E1001000 1 E100346 1 E100342 1 E100341 1 E100340 1 .. E1001718 1 E1001717 1 E1001716 1 E1001713 1 E100998 1 Name: count, Length: 1200, dtype: int64
5	# 2nd Method to be display it EmpNumbers_counts = GRAPHDB["EmpNumber"].value_counts().reset_index() EmpNumbers_counts.columns = ["EmpNumber", "Count"] print(EmpNumbers_counts)
Output	EmpNumber Count 0 E1001000 1 1 E100346 1 2 E100342 1 3 E100341 1

	4 E100340 1 ... ... 1195 E1001718 1 1196 E1001717 1 1197 E1001716 1 1198 E1001713 1 1199 E100998 1 [1200 rows x 2 columns]

- **2. Particular data of Columno. 1 : Employee Departments of Aggregation**
-

Srno	Python code
1	# identifying the types of Departments GRAPHDB["EmpDepartment"].unique() #Here displaying the columun'EmpDepartment' of number of unique numbers
Output	array(['Sales', 'Human Resources', 'Development', 'Data Science', 'Research & Development', 'Finance'], dtype=object)
2	# Count the Number of uniques of "EmpDepartment" GRAPHDB["EmpDepartment"].nunique()
Out	6
3	GRAPHDB.loc[:,"EmpDepartment"].value_counts()
Out	EmpDepartment Sales 373 Development 361 Research & Development 343 Human Resources 54 Finance 49 Data Science 20 Name: count, dtype: int64
4	department_counts = GRAPHDB["EmpDepartment"].value_counts().reset_index() department_counts.columns = ["EmpDepartment", "Count"] print(department_counts)
Out	EmpDepartment Count 0 Sales 373 1 Development 361 2 Research & Development 343 3 Human Resources 54 4 Finance 49 5 Data Science 20

5	EmpDepartments = GRAPHDB["EmpDepartment"].unique() EmpDepartments
out	array(['Sales', 'Human Resources', 'Development', 'Data Science', 'Research & Development', 'Finance'], dtype=object)

- **3. Particular data of Columno. 2 : Employee JobRoles of Aggregation**
-

Srno	Python code
1	# identifying the types of Employees JobRoles GRAPHDB["EmpJobRole"].unique()
Output	array(['Sales Executive', 'Manager', 'Developer', 'Sales Representative', 'Human Resources', 'Senior Developer', 'Data Scientist', 'Senior Manager R&D', 'Laboratory Technician', 'Manufacturing Director', 'Research Scientist', 'Healthcare Representative', 'Research Director', 'Manager R&D', 'Finance Manager', 'Technical Architect', 'Business Analyst', 'Technical Lead', 'Delivery Manager'], dtype=object)
2	GRAPHDB["EmpJobRole"].nunique()
Out	19
3	GRAPHDB.loc[:,"EmpJobRole"].value_counts()
Out	EmpJobRole Sales Executive 270 Developer 236 Manager R&D 94 Research Scientist 77 Sales Representative 69 Laboratory Technician 64 Senior Developer 52 Manager 51 Finance Manager 49 Human Resources 45 Technical Lead 38 Manufacturing Director 33 Healthcare Representative 33 Data Scientist 20 Research Director 19 Business Analyst 16 Senior Manager R&D 15 Delivery Manager 12 Technical Architect 7 Name: count, dtype: int64

4	Jobrole_counts = GRAPHDB["EmpJobRole"].value_counts().reset_index() Jobrole_counts.columns = ["EmpJobRole", "Count"] print(Jobrole_counts)
out	EmpJobRole Count 0 Sales Executive 270 1 Developer 236 2 Manager R&D 94 3 Research Scientist 77 4 Sales Representative 69 5 Laboratory Technician 64 6 Senior Developer 52 7 Manager 51 8 Finance Manager 49 9 Human Resources 45 10 Technical Lead 38 11 Manufacturing Director 33 12 Healthcare Representative 33 13 Data Scientist 20 14 Research Director 19 15 Business Analyst 16 16 Senior Manager R&D 15 17 Delivery Manager 12 18 Technical Architect 7
5	EmpjobRoles = GRAPHDB["EmpJobRole"].unique() EmpjobRoles
Out	array(['Sales Executive', 'Manager', 'Developer', 'Sales Representative', 'Human Resources', 'Senior Developer', 'Data Scientist', 'Senior Manager R&D', 'Laboratory Technician', 'Manufacturing Director', 'Research Scientist', 'Healthcare Representative', 'Research Director', 'Manager R&D', 'Finance Manager', 'Technical Architect', 'Business Analyst', 'Technical Lead', 'Delivery Manager'], dtype=object)

- **4. Particular data of Columno. 4 : Employee Performance of Aggregation**
-

Srno	Python code
1	GRAPHDB["PerformanceRating"].unique()
Out	array([3, 4, 2], dtype=int64)
2	GRAPHDB["PerformanceRating"].nunique()
Out	3
3	GRAPHDB.loc[:,"PerformanceRating"].value_counts()
Out	PerformanceRating 3 874 2 194 4 132 Name: count, dtype: int64
4	PerformanceRatings = GRAPHDB["PerformanceRating"].unique() PerformanceRatings
Out	array([3, 4, 2], dtype=int64)