**Design Principles**

**SOLID**

S- single responsibility

O – Open for extension and closed for modification

L -Liskova substitution principal (sub class can be substituted to the parent class)

I – Interface Segregation (we should not compel client to substitute all methods)

D – Dependency inverstion (Inversion of control) always the main function should depends on the sub modules.

**12 Factor Cloud Principle**

**Design Pattern:**

[**https://www.javatpoint.com/design-patterns-in-java**](https://www.javatpoint.com/design-patterns-in-java)

**Creational Pattern:**

1. **Singleton design pattern**
2. **Factory pattern**
3. **Abstract Factory pattern**
4. **Builder Pattern**
5. **Prototype Pattern**

**Structural Design Pattern**

1. **Façade Pattern**
2. **Proxy Pattern**
3. **Adapter Pattern**
4. **Composite Pattern**
5. **Decorator Pattern**
6. **Bridge Pattern**
7. **Flyweight pattern**

**Behavioural design pattern**

1. **Chain of responsibility pattern**
2. **Interpreter Pattern**
3. **Command Pattern**
4. **Iterator Pattern**
5. **Mediator Pattern**
6. **Observer pattern**
7. **State pattern**
8. **Template pattern**
9. **Memanto pattern**
10. **Strategy pattern**
11. **Visitor Pattern**

**DB Atomic Transaction**

**DB load balancer**

* **Oracle Golden gate**
* **DB Link**

**DB SQL**

[**https://artoftesting.com/sql-queries-for-interview#sql\_minus\_operator**](https://artoftesting.com/sql-queries-for-interview#sql_minus_operator)

* **Different dept available in a table -**
* *selec Distinct(dept) from employee*
* **Count of employee in dept –** *sele count(\*) from employee where dpt=’eee’*
* **Max,min,avg of salary –** *sele max(\*),min(\*) ,avg(\*) from employ where dpt =’333’*
* **Salary between 2000 and 4000 –** *select \* from employ where sal* ***between*** *2000* ***and*** *4000*
* **Employee join date in 2021 –** *select \* from emp where j\_date between 01/01/2021 and 31/12/2021*
* **Employee not working in dept -***select \* from empl where* ***Not*** *dept=”eee”*
* **Unique employee id of both table -***select empid from emp* ***union*** *select empid from depdetail*
* **Common record between two table -***sele empid from emp* ***Intersect*** *sele emo from depdetail*
* **Name replace with space with \_ -** *select REPLACE(Name,’ ‘,’\_’) from emp*
* **Position of a string –** *select INSTR(Name,’word’) from employee*
* **Concat first name and last name –** *select concat(firstname,lastname) from emp*
* **Extract first name alone -***- select mid(fullname,1,Locate(‘ ‘,fullname’) from emp*
* **Upper and lower of name -***select upper(name) ,lower(name) from emp*
* **Trim the name –** *select LTRIM(RTRIM(name)) from emp*
* **Dep is null -***select \* from where dept IS NULL*
* **Date now –** *select sysdate from dual*
* **Present in one table not in other table -***sel empid from emp* ***Minue*** *sele emp from depdetail*

*SELECT EmployeeSalary.\**

*FROM EmployeeSalary*

***LEFT JOIN***

***ManagerSalary USING (EmpId)***

*WHERE ManagerSalary.EmpId IS NULL;*

* **Second top salary in a Department .**
* **3rd top salary in a Department.**

***SELECT TOP 1 Salary***

***FROM (***

***SELECT DISTINCT TOP N Salary***

***FROM Employee where Dept =’EEE’***

***ORDER BY Salary DESC***

***)***

***ORDER BY Salary ASC;***

**Without using Top command**

***SELECT Salary***

***FROM EmployeeSalary Emp1***

***WHERE 2 = (***

***SELECT COUNT( DISTINCT ( Emp2.Salary ) )***

***FROM EmployeeSalary Emp2***

***WHERE Emp2.Salary > Emp1.Salary***

***)***

* **Create a table with same data as old table**

***CREATE TABLE NewTable SELECT \* FROM EmployeeSalary;***

* **Create a table with out data as old table (Always where condition is false)**

***CREATE TABLE NewTable SELECT \* FROM EmployeeSalary WHERE 1=0;***

* **Fetch project-wise count of employees sorted by project’s count in descending order.**

***SELECT Project, count(EmpId) EmpProjectCount***

***FROM EmployeeSalary***

***GROUP BY Project***

***ORDER BY EmpProjectCount DESC;***

* **employee names and salary records. Display the employee details even if the salary record is not present for the employee.**

***SELECT E.FullName, S.Salary***

***FROM EmployeeDetails E***

***LEFT JOIN***

***EmployeeSalary S***

***ON E.EmpId = S.EmpId;***

**Self Join**

*DB Question :*

*Atomicity*

***Joins:***

Chart, bubble chart

Description automatically generated