**Project Phase II**

**Webfactory**

Write code for 5 SQL reports from the tables. For each report include:

Description of Query Report (including purpose, benefit, use in business metric, etc)

SQL statement code, Output screen shot of report (after you write your query

Remember, the reports have to use joins, aggregate functions (AVG, SUM, etc), multiple row conditions (ANY, ALL etc), a conditional command (CASE or DECODE), not necessarily all in one

**QUERY 1**

**Purpose :**

To identify how many websites are handled by each developer on a given date, and whether that developer is available to work on a new incoming work order.

**Benefit :**

The goal is to automate the process of delegation of workers to incoming websites. A developer can work on a maximum of 5 websites at once. Hence, their availability is identified, in order to assign them to new incoming work orders. The no. of websites they are currently working on, the website\_IDs of those websites as well as the no. of websites the developer is available to work on are identified.

**Use in business metric:**

This query is run in order to optimize the business process and reduce the time taken by an account manager to identify the availability of a developer and manually assign the employee to a website.

**Code:**

select e.employee\_id, count(w.website\_id) AS "No.of Websites Assigned", listagg(w.website\_id,', ') within group(order by w.website\_id) as "Website\_ID",

(case

when count(w.website\_id)=4 then 'Developer Available to work on 1 Website'

when count(w.website\_id)=3 then 'Developer Available to work on 2 Website'

when count(w.website\_id)=2 then 'Developer Available to work on 3 Website'

when count(w.website\_id)=1 then 'Developer Available to work on 4 Websites'

when count(w.website\_id)=0 then 'Developer Available to work on 5 Websites'

else 'Developer Unavailable' end) AS "Availability"

from w\_website w join w\_work\_order o on w.order\_id=o.order\_id

join work\_emp\_association a on a.order\_id=o.order\_id

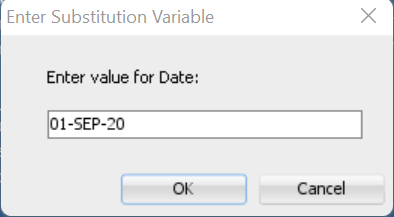
right outer join w\_employee e on a.employee\_id=e.employee\_id

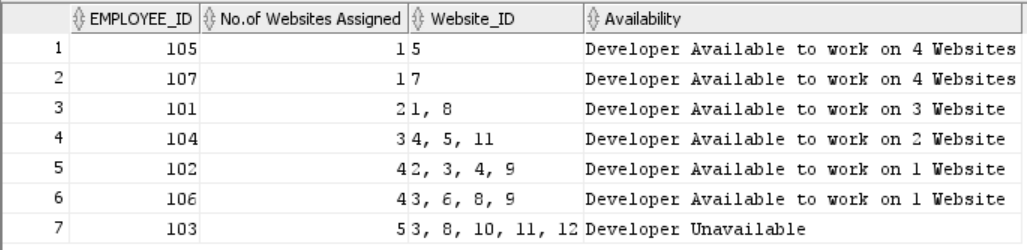
where e.employee\_role='Developer' AND '&date'<o.completion\_date

group by e.employee\_id

order by 2;

**Output:**

****



**QUERY 2**

**Purpose :**

To identify the average number of days taken to build a website with various customization requirements.

**Benefit :**

The objective is to quote price the customization requirements according to the efforts spent. Calculation of average time spent for each customization requirement can help in pricing the customization options commensurate with the amount of days spent in building the website. This can also help allocate the developers to do the task as per their availability to pick the tasks and the no of days required to build the website.

**Use in business metric:**

Having an average time it takes to build a website with a specified customization type will assist in negotiating the timeline and pricing with the client for the delivery of the project. It will help the account managers give a time estimate for additional time needed to do customizations.

**Code:**

select w\_customization.customization\_id, concat(round(avg(build\_date - start\_date)),' days') days\_to\_build\_website , w\_customization.customization\_description

from w\_website

inner join w\_work\_order

on w\_website.order\_id = w\_work\_order.order\_id

inner join web\_cust\_association

on web\_cust\_association.website\_id = w\_website.website\_id

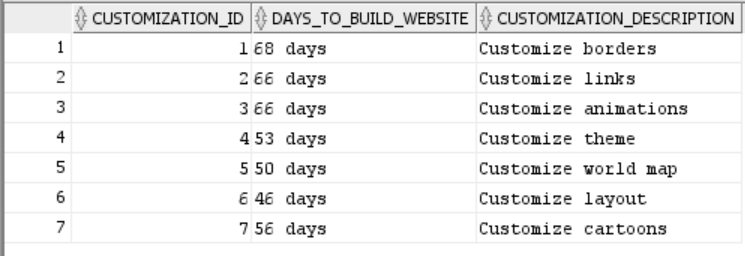
inner join w\_customization

on w\_customization.customization\_id = web\_cust\_association.customization\_id

group by w\_customization.customization\_id, w\_customization.customization\_description

Order by w\_customization.customization\_id ;

**Output:**



**QUERY 3:**

**Purpose :**

To identify the total amount of the invoice for each website including customization charge, template charge and hosting charge.

**Benefit:**

The objective is to automate the process of finding the total amount for each website inclusive of customization charge, template charge and hosting charge as this is needed to prepare the final bill for the client for each website. One website can have multiple customizations and templates. The bill is prepared by adding all the customization charges and template charges and hosting charge for each website and presented to the client as the final bill as doing it manually would be a huge problem.

**Use in business metric:**

This query is run to prepare the final bill for the client for each website that they have ordered. This is very necessary to prepare the final receipt for the client so that client can see the bill for each website that they have ordered and also the bill breakup for customization charge, template charge and hosting charge and the website id as well.

**Code:**

select i.Invoice\_ID, w.website\_id, sum(c.Customization\_Charge) "Customization Charge in $" , sum(h.Hosting\_Charge) "hosting charge in $" ,sum(t.Template\_Charge) "template charge in $",sum(c.Customization\_Charge)+sum(h.Hosting\_Charge)+sum(t.Template\_Charge) "Total Amount in $"

from W\_INVOICE i left join W\_WEBSITE w

on i.INVOICE\_ID =w.Invoice\_ID

left join W\_HOSTING h`1

on h.Hosting\_ID= w.Hosting\_ID

left join Web\_Cust\_Association wca

on w.Website\_id= wca.Website\_id

left join W\_Customization c

on c.Customization\_ID= wca.Customization\_ID

left join Web\_Temp\_Association wta

on w.Website\_id= wta.Website\_id

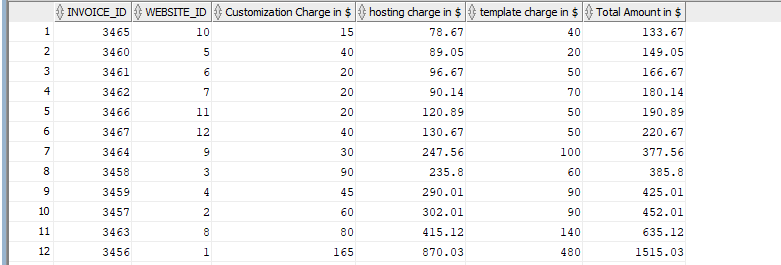
left join W\_TEMPLATE\_TYPE t

on wta.template\_id=t.template\_id

group by (w.Website\_id, i.INVOICE\_ID)

order by 6;

**Output:**



**Query 4:**

**Purpose:**

To calculate bonus received by each employee based on the revenue they have generated for the company by working on websites.

**Benefit:**

The objective is to automate the process of finding the bonus amount for each employee. This bonus is calculated according to the revenue that the employee has generated for the company. The revenue generated by each employee is calculated by adding the template charges, customization charges, hosting charges for the websites that the employee has worked on. If the revenue generated is greater than $2000 then the employee will receive a bonus of $100. If the revenue is greater than $1500 then the employee will receive a $70 bonus. If the revenue is greater than $1000 then the employee will receive a $50 bonus. If the revenue is greater than $500 then the employee will receive a $30 bonus. If it's below $500 then no bonus will be given to the employee. This process is very tedious and tiresome to do it manually.

**Use in Business Metric:**

Bonus amount is required to be calculated by the HR managers of the company to decide which employee has made the most contribution to the revenue generated. Bonus is an important part of the payslip of the employees and it motivates the employees to work harder to achieve the goal and receive appreciation for that.

**Code:**

select e.employee\_id, sum(c.Customization\_Charge) "Customization Charge in $" , sum(h.Hosting\_Charge)"hosting charge in $" ,sum(t.Template\_Charge)"template charge in $",sum(c.Customization\_Charge)+sum(h.Hosting\_Charge)+sum(t.Template\_Charge)"Total Amount in $", (case when sum(c.Customization\_Charge)+sum(h.Hosting\_Charge)+sum(t.Template\_Charge)>2000 then 100 when sum(c.Customization\_Charge)+sum(h.Hosting\_Charge)+sum(t.Template\_Charge)>1500 then 70 when sum(c.Customization\_Charge)+sum(h.Hosting\_Charge)+sum(t.Template\_Charge)>1000 then 50 when sum(c.Customization\_Charge)+sum(h.Hosting\_Charge)+sum(t.Template\_Charge)>500 then 30 else 0 END)"bonus in $ given to employees"

from W\_EMPLOYEE e join Work\_Emp\_Association wea

on e.employee\_ID =wea.employee\_ID

join w\_work\_order o

on o.order\_id=wea.order\_id

join w\_website w

on o.order\_id=w.order\_id

join W\_HOSTING h

on h.Hosting\_ID= w.Hosting\_ID

join Web\_Cust\_Association wca

on w.Website\_id= wca.Website\_id

join W\_Customization c

on c.Customization\_ID= wca.Customization\_ID

join Web\_Temp\_Association wta

on w.Website\_id= wta.Website\_id

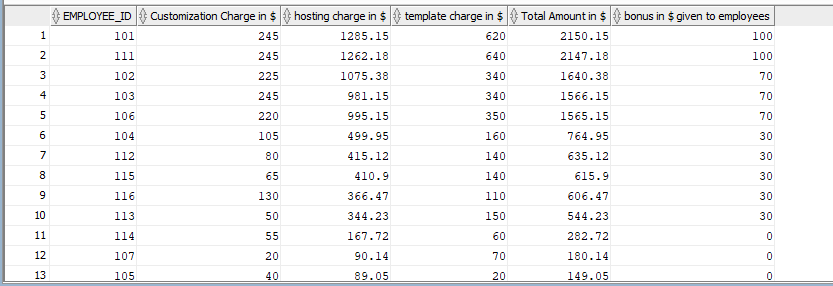
join W\_TEMPLATE\_TYPE t

on wta.template\_id=t.template\_id

group by ( e.employee\_id)

order by 5 desc;

**Output:**



**Query 5:**

**Purpose:**

To identify the clients and the no of orders placed which involve costlier hosting plans.

**Benefit:**

The goal is to get as many orders as possible and earn from the websites built, hence it will be helpful to identify the clients that are willing to pay more for costlier hosting packages and in the past have placed more orders. As these are important clients and the revenue generated has major contributions from them, it becomes important to keep track of the orders placed by them.

**Use in Business Metric:**

Profit earned is proportional to the number of orders placed. However, the major contribution in revenue comes from costlier packages. Running this report can help identify the clients that place orders with expensive hosting plans, so that we can retain them by providing discounts or other strategies if the number of orders declines to maintain the revenue.

**Code:**

SELECT CLIENT\_NAME,COUNT(B.ORDER\_ID) NO\_OF\_ORDERS FROM W\_CLIENT A , W\_WORK\_ORDER B, W\_WEBSITE C

WHERE A.CLIENT\_ID=B.CLIENT\_ID

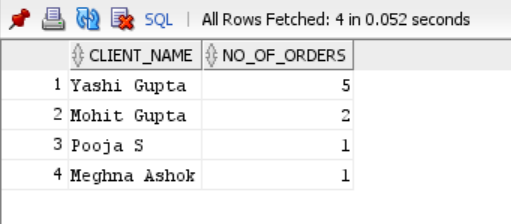
AND B.ORDER\_ID=C.ORDER\_ID

AND C.HOSTING\_ID IN (SELECT HOSTING\_ID FROM W\_HOSTING WHERE HOSTING\_PACKAGE IN ( 'deluxe','premium','economy','platinum'))

GROUP BY CLIENT\_NAME

ORDER BY 2 DESC ;

**Output:**



**Updated ERD :**

