

## Single/Multiple/Joins/Sub-Queries

This work sheet will take you through:

- Writing single and multiple table SQL queries
- Defining and using three types of joins
- Writing non-correlated and correlated sub queries

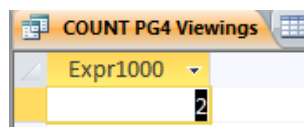
### COUNT () Syntax

- This syntax is used to count the number of rows returned

Example Count () Syntax: Query showing how many viewings have there been for property PG4?

```
SELECT COUNT(pno)
FROM Viewing
WHERE pno="pg4";
```

Result: 2



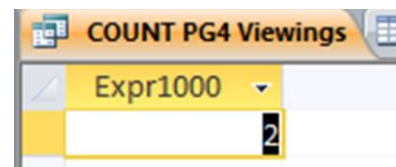
Rno	Pno	Date	Comment
CR56	PA14	24-May-98	Too small
CR56	PG36	28-Apr-98	Too remote
CR56	PG4	26-May-98	
CR62	PA14	14-May-98	No dining room
CR76	PG4	20-Apr-98	

### COUNT () Example – Labelling Results

Example labelling results:

Query: How many viewings have there been for property PG4 where results are labelled?

```
SELECT COUNT(pno) AS ['Viewings for PG4']
FROM Viewing
WHERE pno="pg4";
```



## MIN () Syntax

- Returns the minimum value in the expression. May be followed by the OVER clause.

### Query: Which property has the lowest rental value?

```
SELECT MIN(rent) AS ["Minimum rental"]
FROM Property_for_Rent;
```

Result:

"Minimum rental"
£350.00

## MAX () Syntax

Query: Which member of staff earns the highest salary?

```
SELECT MAX(Salary) AS [Highest salary]
FROM Staff;
```

Result:

Expr1000
£30,000.00

Is information missing from this result?

## Non-Correlated Sub Query Example

```
SELECT Fname, Lname, Salary
FROM Staff
WHERE salary IN (Select MAX(salary)
FROM Staff);
```

Result:

MAX Salary with staff details		
Fname	Lname	Salary
John	White	£30,000.00
*		

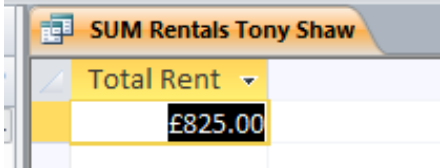
## SUM () Syntax - Correlated Sub Query

- SUM function is used to return the sum of an expression.

Query: What is the total rental value for properties owned by Tony Shaw?

```
SELECT SUM (rent) AS [Total Rent]
FROM Property_for_Rent
WHERE ono LIKE "CO93";
```

Result:



SUM Rentals Tony Shaw	
Total Rent	£825.00

## AVG () Syntax

- SQL AVG function is used to return the average of an expression in a SELECT statement.

What is the average salary of staff?

```
SELECT AVG(salary) AS [Average Salary]
FROM Staff;
```

Write your result here:

## Multiple Table Queries – Join

- SQL query can be used to link tables - for tables where relationships exist
- Relationships between tables are established through PK and FK
- Refers to the common columns in each table
- To link related tables in SQL the SELECT command and WHERE clause is used
- Join – an operation which brings together the data from two or more related tables into one result table
- Three types of joins – equi-joins, natural joins and outer joins/right outer join.

## Example Equi-Join – Listing properties managed by each branch

```
SELECT branch.bno, property_for_rent.bno, property_for_rent.street  
FROM Branch, Property_for_rent  
WHERE branch.bno= Property_for_rent.bno  
ORDER By branch.bno;
```

Write your result here:

## Example Natural Join – Listing properties managed by each branch

- Same as equi-join except the duplicate columns are eliminated.

```
SELECT branch.bno, property_for_rent.street
FROM Branch, Property_for_rent
WHERE branch.bno= Property_for_rent.bno
ORDER By branch.bno;
```

Write your result here:

## Right Outer Join

- When joining two tables, the rows in one table does not have a matching row in the other table.
- Using outer join we can produce information for rows that do not have matching values in common columns.

What is the difference with these two queries that is significant when joining tables?

## QUERY ONE:

```
SELECT property_for_rent.pno, property_for_rent.pcode, property_for_rent.type,
viewing.date
FROM property_for_rent
```

RIGHT JOIN Viewing

ON property\_for\_rent.pno= viewing.pno;

### **QUERY TWO:**

SELECT property\_for\_rent.pno, property\_for\_rent.pcode, property\_for\_rent.type,  
viewing.date

FROM Viewing

RIGHT JOIN property\_for\_rent

ON property\_for\_rent.pno= viewing.pno;

Write your result here: