Topics

1. Review
2. Defining Subqueries
3. Describing the types of problems Subqueries can solve
4. Listing types of Subqueries
5. Writing single row and multiple row Subqueries

Group Functions?

- AVG
- COUNT
- MAX
- MIN
- SUM
- STDDEV
- VARIANCE

- What is the difference between a where clause and a having clause used with group by?
- The Where clause filters out rows that don’t meet a search condition
- The Having clause filters out entire groups that don’t meet a search condition
- Sometimes interchangeable – not always

- How do group functions handle null values?
- Min?
- Ignore any null values
- Avg?
- Ignore null values and exclude that row from the row count total divisor

Subqueries

- Who earns more money than Wilson?
- Who earns more than their manager?
- How do you answer these business questions with SQL?

Subqueries

- PROBLEM: Who earns more money than Wilson ?
- Solution: 2 steps
  - Find out how much Wilson earns
  - Find out who earns more than that amount
- This requires two queries … but
- We should be able to reduce this to a single action
- We need to pass information from one query into the second query
- We need a Subquery to define Wilson’s salary and pass it to the main query that produces the results.

Subqueries

- SELECT <select list>
- FROM <table>
- WHERE <expression> <operator>
-   (SELECT <select list>
-     FROM <table>
-      WHERE <expression> <operator>)

SUBQUERY

- Subquery Syntax
- A Subquery is a SELECT statement that is imbedded in a clause of another SELECT statement.
- Useful when you need to select rows from a table with a condition that depends on data from the same table or other tables
- Where are subqueries used?
- On the following clauses:
- WHERE clause
- FROM clause
- HAVING clause
- Single-row operator < >, =, <, >, etc.
- Multiple-row operators IN, ANY, ALL, EXISTS

SUBQUERY

- OTHER TERMS USED
- Nested SELECT
- Sub-SELECT
- Inner SELECT

- What is the ORDER of OPERATION?
- The Subquery generally executes first and its output is then the fed to the main or OUTER query

SUBQUERY GUIDELINES

- Enclose subquery in parentheses
- Place the Subquery on the right side of the comparison operator for readability
- You can do it the other way
- SELECT * from employees WHERE (select salary from employees where last_name = 'Abel') < salary
- This is awkward and not intuitive
- SELECT * from employees WHERE salary >= (select salary from employees where last_name = 'Abel')

SUBQUERY GUIDELINES

- ORDER BY clause in the Subquery is only needed when performing TOP-N analysis
- Normally the order by clause is only found at the end of the SQL statement.
- TOP-N analysis refers to finding the top number of rows.
- Example top seven salaries
-
- Use single row operators with single row subqueries
- Use multi row operators with multi row subqueries

SUBQUERY TYPES

- Single Row Subqueries – returns one row in the “inner” result set
- Multi Row Subqueries – returns multiple rows in the “inner” result set
- Multi-column Subqueries – returns multiple columns in the “inner” result set

Building a subquery

```
SELECT SALARY FROM STAFF WHERE NAME = 'Wilson'

    SALARY
 68,674.50
```

```
SELECT NAME,
      SALARY
 FROM STAFF
WHERE SALARY >
      (SELECT SALARY
       FROM STAFF
       WHERE NAME = 'Wilson')
```

```
NAME              SALARY
Daniels        69,260.25
Williams       69,456.50
Quill          69,818.00
Lu             70,010.00
Hanes          70,659.80
Graham         71,000.00
Fraye          71,150.00
Jones          71,234.00
Molinare       72,959.20
```

Subqueries

- Who makes more than Wilson?
- Subselect on the left works, but we have to change the sign and it is not intuitive

```
SELECT * FROM STAFF
WHERE (SELECT SALARY FROM STAFF
       WHERE NAME = 'Wilson') <= SALARY
```

- Always put sub select on the right

Subqueries

- What is the lowest salary for those who make more than Wilson

```
LOWSALARY
69,260.25
```

```
SELECT MIN(SALARY) LOWSALARY
FROM
     (SELECT NAME, SALARY
      FROM   STAFF
      WHERE  SALARY >
          (SELECT SALARY
           FROM   STAFF
           WHERE  NAME = 'Wilson'))
```

Subqueries

- What is the name of the person who has the lowest salary for those who make more than Wilson

```
Column NAME or expression in SELECT list not valid.
SELECT NAME, MIN(SALARY) LOWSALARY
FROM   (SELECT NAME, SALARY
        FROM   STAFF
        WHERE SALARY >
           (SELECT SALARY
            FROM   STAFF
            WHERE NAME = 'Wilson'))
```

Subqueries

- What is the name of the person who has the lowest salary for those who make more than Wilson

```
SELECT * FROM STAFF
WHERE  SALARY >
      (SELECT SALARY FROM STAFF
       WHERE NAME = 'Wilson')
order by salary
FETCH FIRST ROW ONLY
```

```
 ID   NAME         DEPT   JOB      YEARS     SALARY
240   Daniels        10   Mgr          5  69,260.25
```

Subqueries

- Who are the top five earners from the group that make less than Wilson?

```
   SALARY
68,674.50
```

```
NAME             SALARY
Lea           68,555.50
Sanders       68,357.50
Plotz         68,352.80
Pernal        68,171.25
O'Brien       68,006.00
```

```
SELECT NAME, SALARY FROM STAFF
WHERE SALARY <
             (SELECT SALARY
              FROM STAFF
              WHERE NAME = 'Wilson' )
ORDER BY SALARY DESC
FETCH FIRST 5 ROWS ONLY
```

Subqueries – Multi Row Subqueries

- What are the minimum salaries for each job and who is the employee

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFF
WHERE  SALARY =
           (SELECT MIN(SALARY)
            FROM   STAFF
            GROUP BY JOB)
```

- What's wrong with the query above?
- A GROUP BY implies multiple rows being returned
- An = requires a single result to compare with
- You cannot have SALARY = to n different items

```
Result of SELECT more than one row.
```

Subqueries – Multi Row Subqueries

- To use a Subquery that returns more than one row you need to use a Multiple-row operator
- IN – Equal to any member in the list
- ANY – Compare value to each value returned by the subquery
- ANY – Returns true if the condition matches at least one value in the subquery result set
- ALL – Compare value to every value returned by the subquery
- ALL – Returns true if the condition matches all the values in the subquery result set

Minimum salary for each job

```
NAME       JOB      SALARY
Yamaguchi  Clerk  60,505.90
Marenghi   Mgr    67,506.75
Davis      Sales  65,454.50

           SELECT NAME,
                  JOB,
                  SALARY
           FROM   STAFF
           WHERE  SALARY IN
                      (SELECT MIN(SALARY)
                       FROM   STAFF
                       GROUP BY JOB)
```

Minimum salary for each job

- What if more than one clerk has the same minimum salary? -- STAFFM

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFFM
WHERE  SALARY IN
        (SELECT MIN(SALARY)
         FROM STAFFM
         GROUP BY JOB)
```

```
NAME        JOB        SALARY
Yamaguchi   Clerk   69,581.78
Faraday     Clerk   69,581.78
Marenghi    Mgr     67,506.75
Davis       Sales   65,454.50
********  End of data   ****
```

Subqueries – Multi Row Subqueries

- PROBLEM: Show me information on the lowest paid employee in each department

```
NAME            DEPT       SALARY
Kermisch          15    62,258.50
James             20    63,504.60
Abrahams          38    62,009.75
Burke             66    60,988.00
Gafney            84    63,030.50
Yamaguchi         42    60,505.90
Lundquist         51    63,369.80
Daniels           10    69,260.25
********   End of data   ******
```

```
SELECT name,
       dept,
       SALARY
FROM STAFF
WHERE SALARY IN (
      select min(salary)
      from   staff
      group by dept)
```

Manager Salaries

```
SELECT NAME, JOB, SALARY
FROM   STAFF
WHERE  JOB = 'Mgr'
```

```
NAME        JOB        SALARY
Sanders     Mgr     68,357.50
Marenghi    Mgr     67,506.75
Hanes       Mgr     70,659.80
Plotz       Mgr     68,352.80
Fraye       Mgr     71,150.00
Molinare    Mgr     72,959.20
Lu          Mgr     70,010.00
Daniels     Mgr     69,260.25
Jones       Mgr     71,234.00
Lea         Mgr     68,555.50
Quill       Mgr     69,818.00
********  End of data   ******
```

Top Saleperson salary

- How does the top Sales earner compare in salary to the manager salaries

```
SELECT NAME, JOB, SALARY
FROM   STAFF
WHERE SALARY >= ALL
       (SELECT SALARY
        FROM STAFF
        WHERE JOB = 'Sales')
ORDER BY SALARY

NAME          JOB        SALARY
Graham        Sales   71,000.00
Fraye         Mgr     71,150.00
Jones         Mgr     71,234.00
Molinare      Mgr     72,959.20
********     End of data    ****
```

```
SELECT NAME, JOB, SALARY
FROM   STAFF
WHERE  SALARY >=
       (SELECT MAX(SALARY)
        FROM STAFF
        WHERE JOB = 'Sales')
ORDER BY SALARY

NAME          JOB        SALARY
Graham        Sales   71,000.00
Fraye         Mgr     71,150.00
Jones         Mgr     71,234.00
Molinare      Mgr     72,959.20
********     End of data    ****
```

- Change ALL to ANY

```
SELECT NAME, JOB, SALARY
FROM   STAFF
WHERE SALARY >= Any
       (SELECT SALARY
        FROM STAFF
        WHERE JOB = 'Sales')
ORDER BY SALARY
```

```
NAME        JOB        SALARY
Davis       Sales   65,454.50
Rothman     Sales   66,502.83
Quigley     Sales   66,808.30
Gonzales    Sales   66,858.20
Marenghi    Mgr     67,506.75
Smith       Sales   67,654.50
Edwards     Sales   67,844.00
Koonitz     Sales   68,001.75
O'Brien     Sales   68,006.00
Pernal      Sales   68,171.25
Plotz       Mgr     68,352.80
Sanders     Mgr     68,357.50
Lea         Mgr     68,555.50
Wilson      Sales   68,674.50
Daniels     Mgr     69,260.25
Williams    Sales   69,456.50
```

Change ALL to ANY

- Could have used an easier equivalent

```
SELECT COUNT(*) TOTALSTAFF
FROM (
     SELECT NAME, JOB, SALARY
     FROM   STAFF
     WHERE SALARY >= Any
       (SELECT SALARY
        FROM STAFF
        WHERE JOB = 'Sales'))


          TOTALSTAFF
                  23
```

```
SELECT COUNT(*) TOTALSTAFF
FROM (
     SELECT NAME, JOB, SALARY
     FROM   STAFF
     WHERE SALARY  >=
       (SELECT MIN(SALARY)
        FROM STAFF
        WHERE JOB = 'Sales'))


          TOTALSTAFF
                  23
```

Subqueries – Multi Row Subqueries

- The ANY operator is a logical operator that compares a value with a set of values returned by a subquery.
- The ANY operator must be preceded by a comparison operator >, >=, <, <=, =, <> and followed by a subquery.
- X = ANY (…) The values in column c must match one or more values in the set to evaluate to true.
- X > ANY (…) The values in column c must be greater than the smallest value in the set to evaluate to true.
- X < ANY (…) The values in column c must be smaller than the biggest value in the set to evaluate to true.
- X <> ANY (…) The values in column c must not match one or more values in the set to evaluate to true.

Subqueries – Multi Row Subqueries

- PROBLEM: WHO EARNS LESS THAN MINIMUM OF ANY JOB CATEGORY

```
SELECT JOB,MIN(SALARY)
FROM   STAFF
GROUP BY JOB
ORDER BY JOB
```

```
JOB     MIN ( SALARY
Clerk       60,505.90
Mgr         67,506.75
Sales       65,454.50
```

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFF
WHERE
       SALARY < ANY
          (SELECT MIN(SALARY)
           FROM   STAFF
           GROUP BY JOB)
ORDER BY SALARY
```

Subqueries – Multi Row Subqueries

- PROBLEM: Do both of these exclude the manager minimum job category or does only one work?

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFF
WHERE  JOB <> 'Mgr' AND
       SALARY < ANY
            (SELECT MIN(SALARY)
             FROM   STAFF
             GROUP BY JOB)
ORDER BY SALARY
```

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFF
WHERE
       SALARY < ANY
            (SELECT MIN(SALARY)
             FROM   STAFF
             GROUP BY JOB
             HAVING JOB <> 'Mgr')
ORDER BY SALARY
```

- Are they both the same?
- No!

- What was the problem with the where clause substituting for a having clause

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFF
WHERE  SALARY < ANY
       (SELECT MIN(SALARY)
        FROM STAFF
        WHERE JOB <> 'Mgr'
        GROUP BY JOB)
ORDER BY SALARY
```

- It was placed in the wrong area before

Subqueries – Multi Row Subqueries

- PROBLEM: Display employees with a salary less than people with sales job

```
SELECT NAME,
       JOB,
       SALARY
FROM   STAFF
WHERE SALARY < ANY(
select salary
from   staff
where job = 'Sales')
```

Subqueries – Multi Row Subqueries

- It makes more business sense to remove sales jobs from the result set

```
 ID   NAME          DEPT   JOB       SALARY
130   Yamaguchi       42   Clerk  60,505.90
330   Burke           66   Clerk  60,988.00
200   Scoutten        42   Clerk  61,508.60
180   Abrahams        38   Clerk  62,009.75
170   Kermisch        15   Clerk  62,258.50
110   Ngan            15   Clerk  62,508.20
120   Naughton        38   Clerk  62,954.75
350   Gafney          84   Clerk  63,030.50
230   Lundquist       51   Clerk  63,369.80
 80   James           20   Clerk  63,504.60
190   Sneider         20   Clerk  64,252.75
250   Wheeler         51   Clerk  64,460.00
300   Davis           84   Sales  65,454.50
 70   Rothman         15   Sales  66,502.83
 60   Quigley         38   Sales  66,808.30
```

Subqueries – Multi Row Subqueries

- No Sales people in our result set

```
ID    NAME          DEPT   JOB        SALARY
10    Sanders         20   Mgr     68,357.50
30    Marenghi        38   Mgr     67,506.75
50    Hanes           15   Mgr     70,659.80
80    James           20   Clerk   63,504.60
00    Plotz           42   Mgr     68,352.80
10    Ngan            15   Clerk   62,508.20
20    Naughton        38   Clerk   62,954.75
30    Yamaguchi       42   Clerk   60,505.90
70    Kermisch        15   Clerk   62,258.50
80    Abrahams        38   Clerk   62,009.75
90    Sneider         20   Clerk   64,252.75
00    Scoutten        42   Clerk   61,508.60
10    Lu              10   Mgr     70,010.00
```

Subqueries – Multi Row Subqueries

```
SELECT id,
       name,
       dept,
       job,
       SALARY
FROM   staff
WHERE  job != 'Sales' and salary < ANY
      (SELECT salary
      FROM   staff
      WHERE  job = 'Sales')
```

ALL does not get the same result

Subqueries – Multi Row Subqueries

- PROBLEM: Display employees with a salary less than people with sales job
- Looking at the outer query, the result shows employees who are not in sales and whose salary is less than ANY salary that is returned by the inner subquery
- The inner subquery sends back all the salaries for job equal to sales.
- Since the outer query is looking for a salary less than ANY of the sales salaries then it is looking for a value that is less than the maximum value returned by the inner Subquery.
- There are a variety of combinations using IN, ANY and ALL which would provide a result set that you are looking for.

Subqueries – Multi Row Subqueries

- IN – Equal to any member in the list
- ANY – Compare value to each value returned by the subquery
- ALL – Compare value to every value returned by the subquery
- > ALL -- greater than all means more than the maximum
- < ALL -- less than all means less than the minimum
- NOT – can be used to produce the “opposite” results for any of the above

Subqueries – Null results

- PROBLEM: Who earns more money than Xx ?

```
SELECT
     NAME,
     SALARY
FROM STAFF
WHERE SALARY >
     (SELECT SALARY
      FROM   STAFF
      WHERE NAME = 'Xx')
```

- This is OK. You will just receive an empty result set.

```
NAME          SALARY
********* End of data  *
```

Subqueries – Multi-Column Results

- A multiple-column subquery returns more than one column to the outer query
- The multiple-column subquery can be listed in the outer query's
- FROM clause
- WHERE clause
- HAVING clause

Subqueries – Multi-Column Results

- PROBLEM: Show the employee or employees in each department whose current salary is the lowest (or minimum) salary in the department.

```
NAME          DEPT       SALARY
Daniels         10    69,260.25
Burke           66    60,988.00
James           20    63,504.60
Abrahams        38    62,009.75
Yamaguchi       42    60,505.90
Gafney          84    63,030.50
Kermisch        15    62,258.50
Lundquist       51    63,369.80
********   End of data   ********
```

Subqueries – Multi-Column Results

- PROBLEM: Show the employee or employees in each department whose current salary is the lowest (or minimum) salary in the department.

```
SELECT NAME,
       DEPT,
       SALARY
FROM   STAFF
WHERE  (DEPT, SALARY) IN
       (SELECT DEPT, MIN(SALARY)
        FROM   STAFF
        GROUP BY DEPT)
```

Subqueries

- PROBLEM: Provide details on employees who earn more than their manager ?

```
SELECT DEPT, ID, NAME, JOB, SALARY
FROM STAFF  S
WHERE S.SALARY >
      (SELECT SALARY
       FROM STAFF
       WHERE JOB = 'Mgr' AND
       DEPT = S.DEPT)
```

```
Result of SELECT more than one row.
```

Subqueries

- Check managers

```
DEPT        ID    NAME        JOB         SALARY
  10       160    Molinare    Mgr      72,959.20
  10       210    Lu          Mgr      70,010.00
  10       240    Daniels     Mgr      69,260.25
  10       260    Jones       Mgr      71,234.00
  15        50    Hanes       Mgr      70,659.80
  20        10    Sanders     Mgr      68,357.50
  38        30    Marenghi    Mgr      67,506.75
  42       100    Plotz       Mgr      68,352.80
  51       140    Fraye       Mgr      71,150.00
  66       270    Lea         Mgr      68,555.50
  84       290    Quill       Mgr      69,818.00
```

Employees who earn more than their managers

```
SELECT DEPT, ID, NAME, JOB, SALARY
FROM STAFF  S
WHERE S.SALARY >
      (SELECT SALARY
       FROM STAFF
       WHERE JOB = 'Mgr' AND
       DEPT = S.DEPT     AND
       DEPT <> 10)
```

```
  DEPT        ID   NAME       JOB        SALARY
    38        40   O'Brien    Sales   68,006.00
    66       280   Wilson     Sales   68,674.50
    66       310   Graham     Sales   71,000.00
 *******   End of data   ********
```

- Notice in this case we need to have a WHERE clause predicate which references the outer STAFF column value

Subqueries

- PROBLEM: Find the job with the lowest average salary. Display the job ID and that average salary.
- Sometimes there are simpler approaches than sub-selects

```
SELECT JOB,
       AVG(SALARY)
FROM   STAFF
GROUP BY JOB
ORDER BY AVG(SALARY) ASC
FETCH FIRST 1 ROW ONLY
```

```
JOB                       AVG ( SALARY )
Clerk  62,612.6125000000000000000000000
********  End of data  ********
```

- Using DESC instead would give you the highest average salary job

Subqueries

- PROBLEM: Find the job with the highest average salary. Display the job ID and that average salary.

```
SELECT JOB,
       AVG(SALARY)
FROM   STAFF
GROUP BY JOB
ORDER BY AVG(SALARY) desc
FETCH FIRST 1 ROW ONLY
```

```
 JOB                          AVG ( SALARY )
 Mgr    69,805.8000000000000000000000000000
 ********  End of data  ********
```

Subqueries

- Is the Select statement the only statement that can use a subquery?
- You want to have a separate table for anyone older than 49 and you don’t want to keep the charge amount
- less rows and less columns in new table

```
CREATE TABLE PATIENTB AS (
SELECT PATIENTNO,
       FIRSTNAME,
       LASTNAME,
       BIRTHDATE,
       YEAR(CURRENT DATE) - YEAR(BIRTHDATE) AGE
FROM   PATIENT3
WHERE  YEAR(CURRENT DATE) - YEAR(BIRTHDATE) > 49 ) WITH DATA
```

Subqueries

- Now you want to include the younger patients as well in this new table

```
INSERT INTO  PATIENTB (
SELECT PATIENTNO,
       FIRSTNAME,
       LASTNAME,
       BIRTHDATE,
       YEAR(CURRENT DATE) - YEAR(BIRTHDATE) AGE
FROM   PATIENT3
WHERE  YEAR(CURRENT DATE) - YEAR(BIRTHDATE) < 50 )
```

Subqueries

- Remove the patients who are within five years of the oldest patient

```
delete from patientb
where age >
(select max(age)- 5 from patientb)
```

```
  2 rows deleted from PATIENTB in DBS311
```

INSURANCE4

- A few more companies were added to INSURANCE4

```
INSNUM  INSURENAME                   INSPHONE     MAXPAYOUT
  555   Manulife                7,056,663,344  1,000,000.00
  666   Royal Insurance         4,167,774,444  2,500,000.00
  888   Cut Rate Insurers       9,058,883,333    100,000.00
  444   SureHealth Ins          6,132,225,555  4,000,000.00
  111   Sun Life                6,135,654,444  2,000,000.00
  222   Co-Operators            4,163,653,434  4,000,000.00
  333   Canada Life             9,052,347,655    400,000.00
  777   Desjardins              5,134,578,987  4,500,000.00
  999   Foresters               8,884,356,754  2,900,000.00
```

PATIENT4

- A few more patients were added to PATIENT4 (A complete list shows on a blackboard handout)

```
PATIENTNO  FIRSTNAME        LASTNAME         PINSURNUM  BIRTHDATE      CHARGE
   123     Karen            Wong                555     12/25/67     6,000.00
   456     Bill             Trimble             666     07/01/78    80,000.00
   789     Tom              Seaver              888     02/22/84    39,999.00
   246     John             Howard              444     04/15/98    12,000.00
   333     Mandy            Suarez              444     11/25/77     3,200.00
   555     Kumar            Rajendra            222     04/04/88     3,300.00
   777     Wendy            Thomas              222     01/24/93    50,000.00
   888     Casey            Stengal             111     02/18/66    44,000.00
   999     Ted              Mendez              111     11/11/68     2,200.00
   321     Mary             Worth               666     07/01/78     8,000.00
   432     Jerry            Lowell              888     04/30/61     5,999.00
   654     Wei              Tsung               444     05/15/77       900.00
   543     Farah            Sanchez             999     11/24/87    16,500.00
   654     John             Brown               777     04/02/84    60,000.00
```

Problem

- Which patients had a lower charge than Karen Wong?

```
PATIENTNO  FULLNAME                             CHARGE
   123     Karen Wong                         6,000.00
```

```
PATIENTNO  FULLNAME                 INSURENAME                 CHARGE
   694     Delores Quintana         Canada Life              4,400.00
   555     Kumar Rajendra           Co-Operators             3,300.00
   979     Vicenzo Lucchessi        Cut Rate Insurers        4,200.00
   432     Jerry Lowell             Cut Rate Insurers        5,999.00
   473     Masatoshi Yoshimura      Foresters                5,400.00
   999     Ted Mendez               Sun Life                 2,200.00
   654     Wei Tsung                SureHealth Ins             900.00
   333     Mandy Suarez             SureHealth Ins           3,200.00
```

Solution

```
SELECT PATIENTNO,
       TRIM(FIRSTNAME) || ' ' || LASTNAME FULLNAME,
       INSURENAME,
       CHARGE
FROM  PATIENT4 INNER JOIN INSURANCE4
ON PINSURNUM = INSNUM
WHERE CHARGE <
            (SELECT CHARGE
             FROM   PATIENT4
             WHERE PATIENTNO = 123)
ORDER BY INSURENAME
```

Problem

- Which companies paid out more than Royal Insurance paid out?

```
ROYALPAYOUT
  99,900.00
```

```
PINSURNUM  INSURENAME                                      HIGERPAYOUT
   222     Co-Operators                                     107,300.00
   111     Sun Life                                         106,199.00
```

```
SELECT PINSURNUM,
       INSURENAME,
       SUM(CHARGE) TOTALPAYOUT
FROM   PATIENT4,INSURANCE4
WHERE  PINSURNUM = INSNUM
GROUP  BY PINSURNUM, INSURENAME
HAVING SUM(CHARGE) >
           (SELECT SUM(CHARGE)
            FROM PATIENT4
            WHERE PINSURNUM = 666)
```

Lab 3

- Lab 3 for already available and Lecture 3 Powerpoint will be uploaded sometime tonight.