**Python code**

def number\_cleaner(raw\_number):

"""Number\_cleaner function cleans a mobile phone number

Then returns the cleaned mobile number

Or raise error if length provided in not correct.

Keyword argument:

raw\_number --- mobile number to be cleaned

Local variable:

clean\_number --- variable to store a cleaned mobile number

"""

clean\_number = ''

for i in raw\_number:

if i.isdigit():

clean\_number += i

if clean\_number[0] != '0': clean\_number = '0' + clean\_number

if len(clean\_number) == 10:

return clean\_number

raise Exception("Number provided is not the correct length")

**SQL Code**

1…

create table tele\_sales(

sale\_id int identity(1,1) PRIMARY KEY,

dialled\_number varchar(15) not null,

sale\_date datetime DEFAULT CURRENT\_TIMESTAMP,

sale\_result varchar(12),

user\_ext varchar(4) not null,

CHECK (sale\_result IN ('Voicemail', 'Wrong Number', 'Answered'))

);

2…

select user\_ext, count(sale\_date)

from Tele\_sales

group by user\_ext

order by user\_ext, sale\_date

3…

select dialled\_number, count(dialled\_number)

from Tele\_sales

group by dialled\_number

order by count(dialled\_number) desc

4…

select \*, count(dialled\_number)

from Tele\_sales

group by dialled\_number

5…

select date(sale\_date), round(sum(case when sale\_result = 'Answered' then 1 else 0 end) \* 100 / count(sale\_result), 1) as percentage

from Tele\_sales

group by date(sale\_date)

6…

select dialled\_number, count(dialled\_number)

from Tele\_sales

group by dialled\_number

order by count(dialled\_number) desc

limit 10

7…

select user\_ext

from Tele\_sales

group by user\_ext

having count(dialled\_number) < 100

8…

select timediff(max(time(sale\_date)), min(time(sale\_date)))

from Tele\_sales

group by user\_ext