BUISNESS REPORT FOR CAPSTONE PROJECT:

PART 1 – EXCEL:

From the problem statement we are told to help analyse visualize data that will help the company identify the potential customer who have higher probability of purchasing the loan. This will increase the success ratio while at the same time reduce the cost of the campaign.

TASK/QUESTION:

1. . Understanding Data and Missing Value treatment:
   a.) Basic data statistics:
   For Numerical Variables:

balance		*age*		*duration*	
Mean	1477.190428	Mean	41.21021021	Mean	264.1601602
Standard Error	88.43752675	Standard Error	0.3312958	Standard Error	8.686910076
Median	459	Median	40	Median	188
Mode	0	Mode	32	Mode	168
Standard Deviation	2771.356095	Standard Deviation	10.47125351	Standard Deviation	274.5668302
Sample Variance	7680414.607	Sample Variance	109.6471502	Sample Variance	75386.94426
Kurtosis	18.60595268	Kurtosis	0.223844898	Kurtosis	18.42784741
Skewness	3.666405296	Skewness	0.639932097	Skewness	3.380619048
Range	28645	Range	68	Range	3020
Minimum	-1680	Minimum	15	Minimum	5
Maximum	26965	Maximum	83	Maximum	3025
Sum	1450601	Sum	41169	Sum	263896
Count	982	Count	999	Count	999

- Age: From the above count we see that the mean of the age is 41. The average is age is 40 and the most occurred age group is 32.The maximum age is 83 and minimum age is 15.The skewness observed is positive and almost near zero and almost symmetrical.
- Balance: The average of all the balances in the account is 1477.1904. The minimum of the balance is in negative –1680 /- which means people are in debt and maximum is 26965/- and skewness is positive.
- Duration: The average/ mean is 264.16 seconds of call. The median here is 188 and 168 and the skewness is positive and greater than zero where the range of the duration of one call is 3025 seconds.

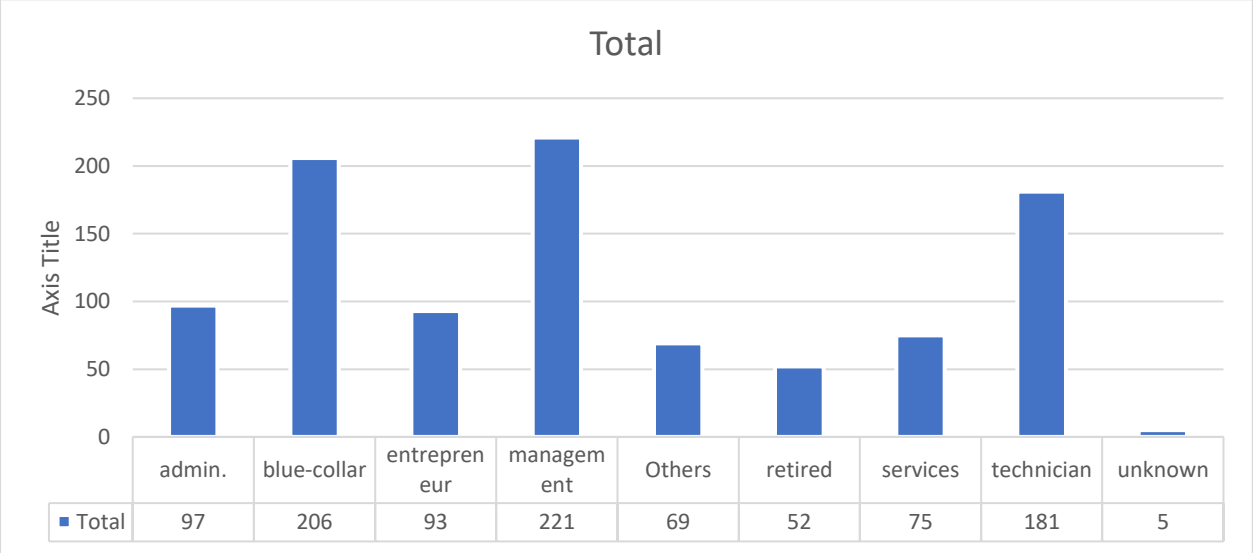
campaign		*previous*	
Mean	2.646646647	Mean	0.556556557
Standard Error	0.090203437	Standard Error	0.053150955
Median	2	Median	0
Mode	1	Mode	0
Standard Deviation	2.851056526	Standard Deviation	1.679940163
Sample Variance	8.128523313	Sample Variance	2.822198952
Kurtosis	30.62905644	Kurtosis	48.51218781
Skewness	4.552791534	Skewness	5.781464521
Range	31	Range	20
Minimum	1	Minimum	0
Maximum	32	Maximum	20
Sum	2644	Sum	556
Count	999	Count	999

- Campaign: The average number of calls performed during this campaign and for this client is 2.64. The minimum number of calls are 1 during that period and maximum goes up to 32 calls. The skewness is positive.

- Previous: The number of calls performed before and for this client is 0.55. The number of calls surprisingly before could be max 0 or min 0 as well. The skewness id greater than 5 for this case.

b.) For Categorical values:

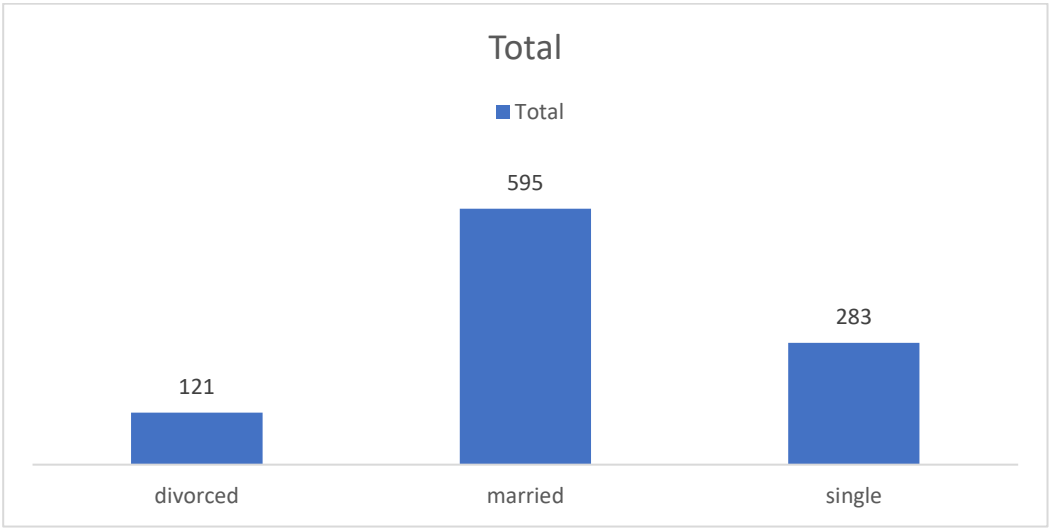
- Job:



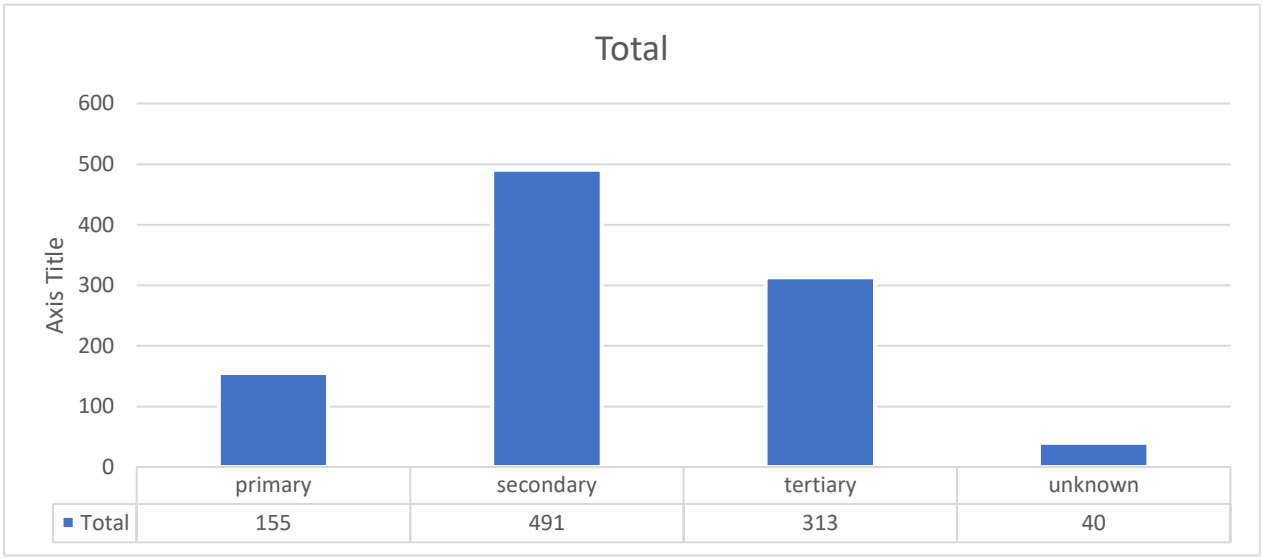
From the above graph we can see the categories of the job people can hold. The maximum number of job category are management with 221 job count followed by blue- collar with 206 and there are 5 of them whose job is unidentified.

- Marital status:

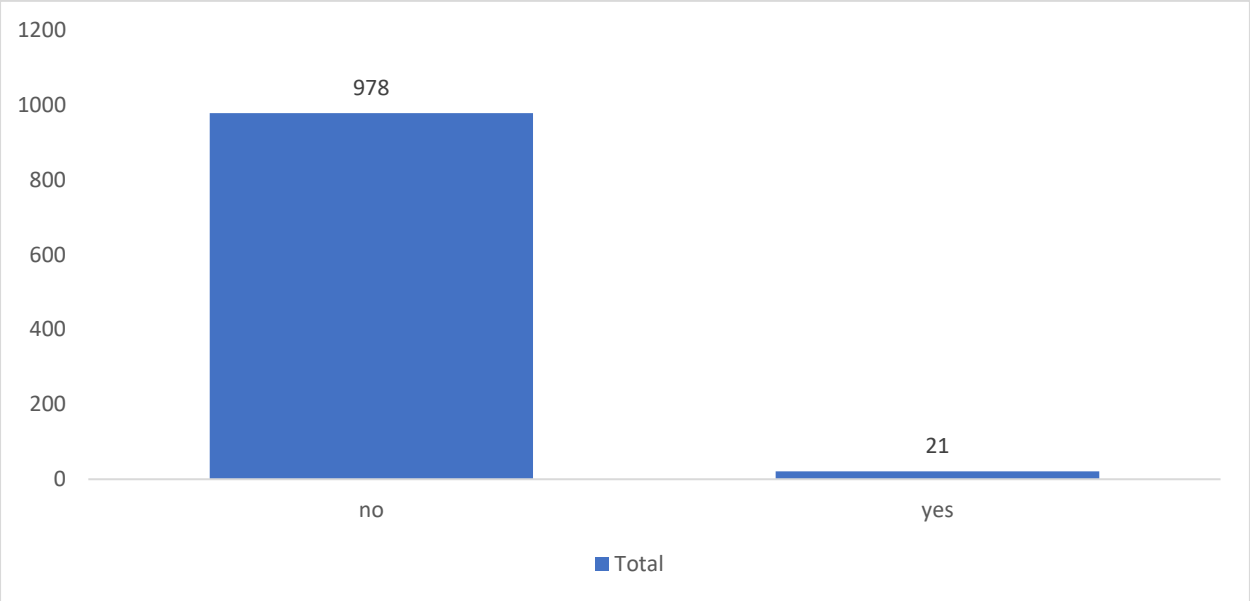
As we can see there high number of count for married people 595 and 121 and divorced which is a few count and 283 single.



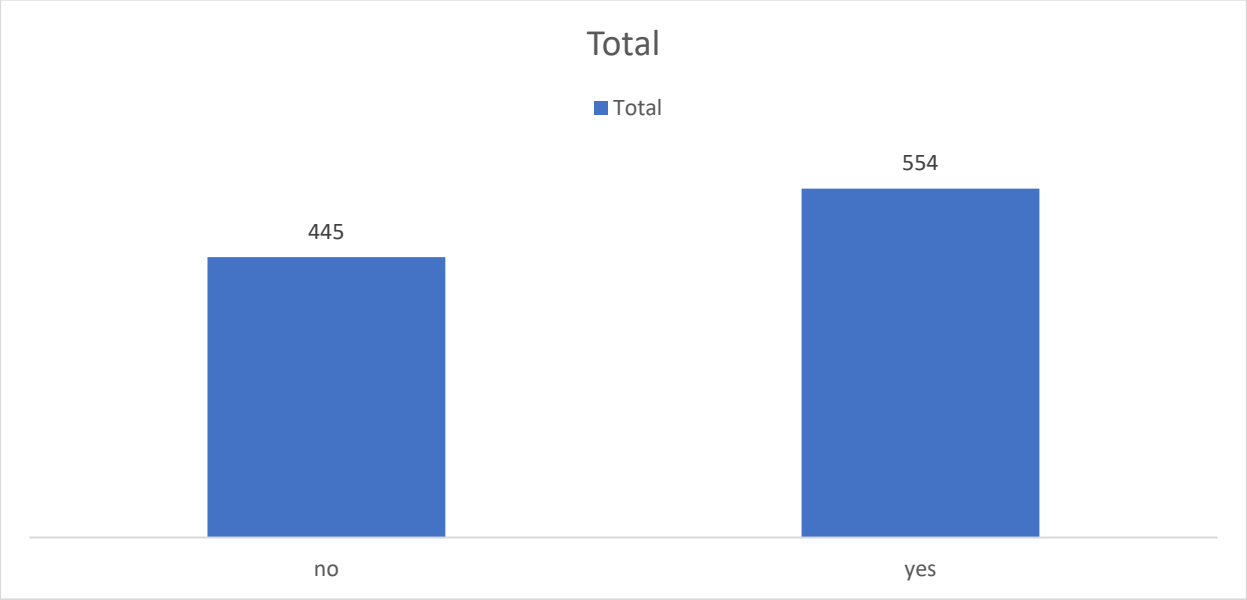
- Education: 491 people have secondary level of education from the pivot table we achieved. The number of people with tertiary level is 313 and top primary level of education 155.



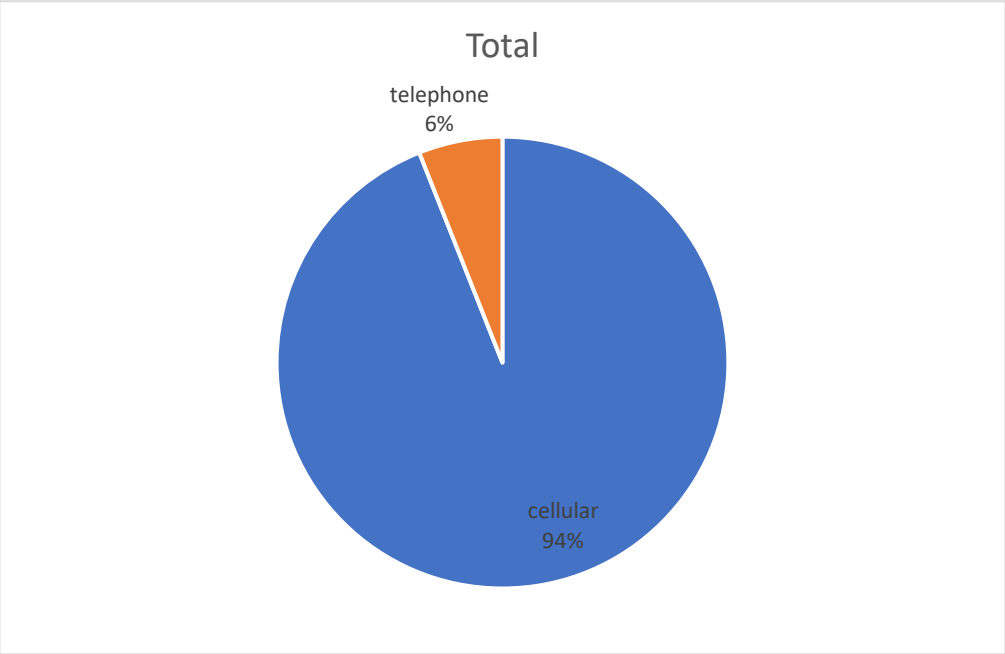
- Default: The number of people with credit default 978 and only 21 people with no default.



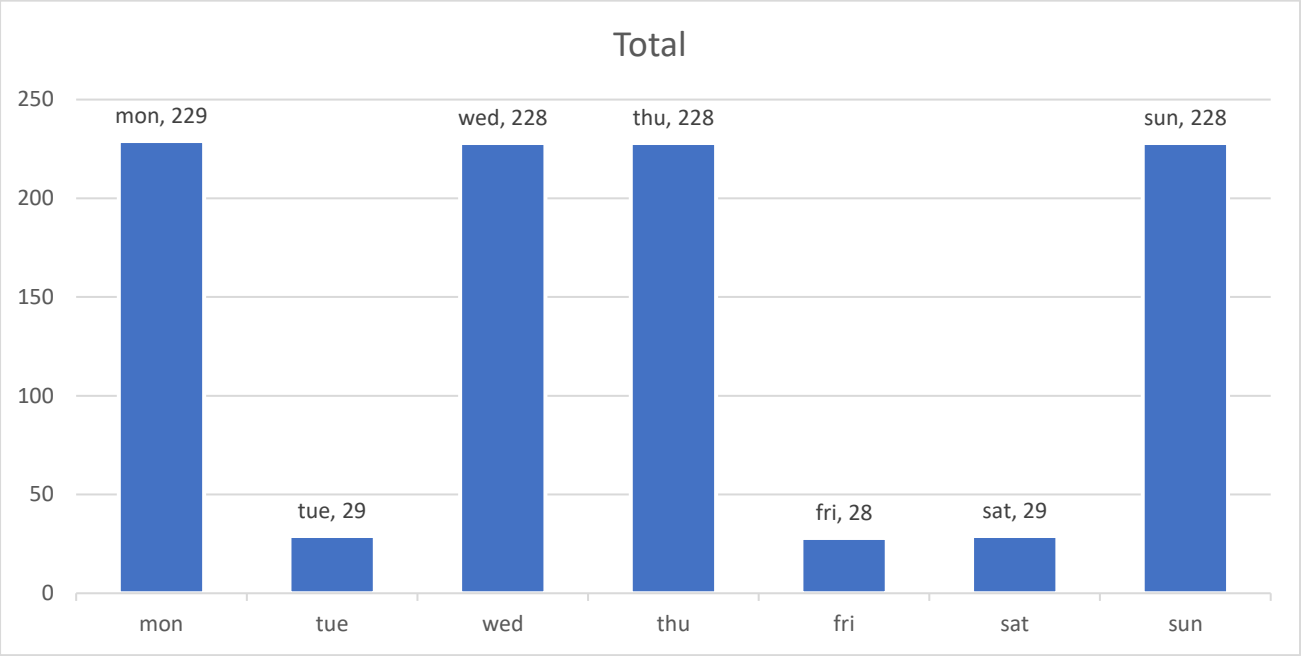
- Housing: 445 people do not have house loans and 554 people have house loans.



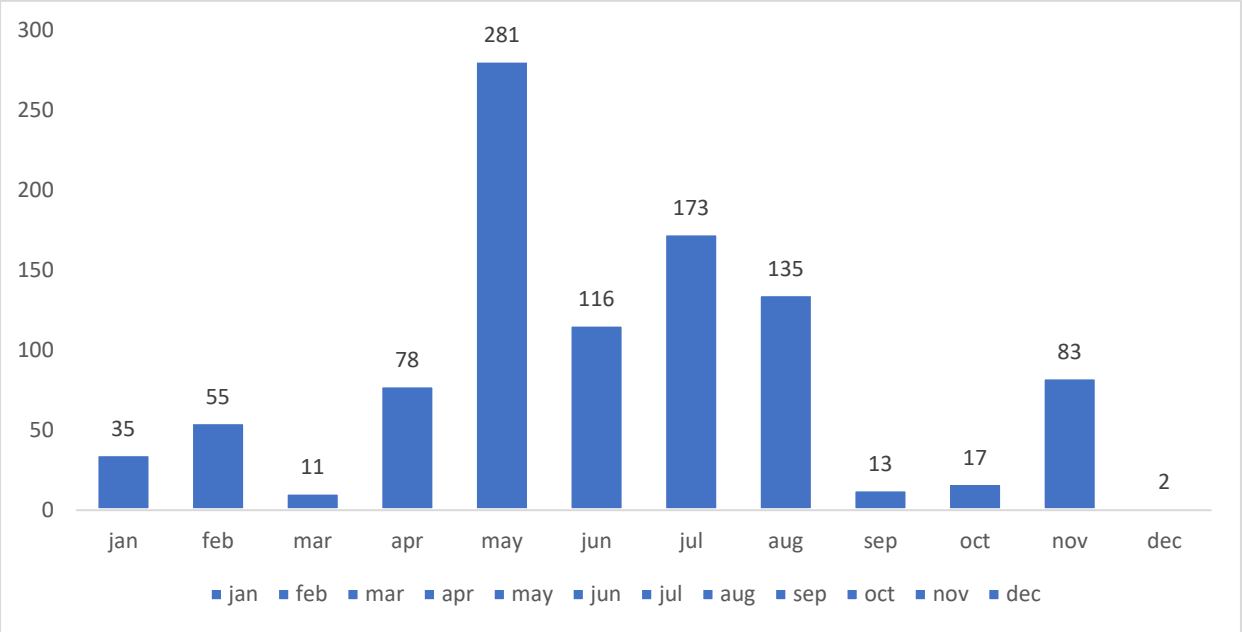
- Contact: Customers using telephones is 6% but the ones using cellular 94%.



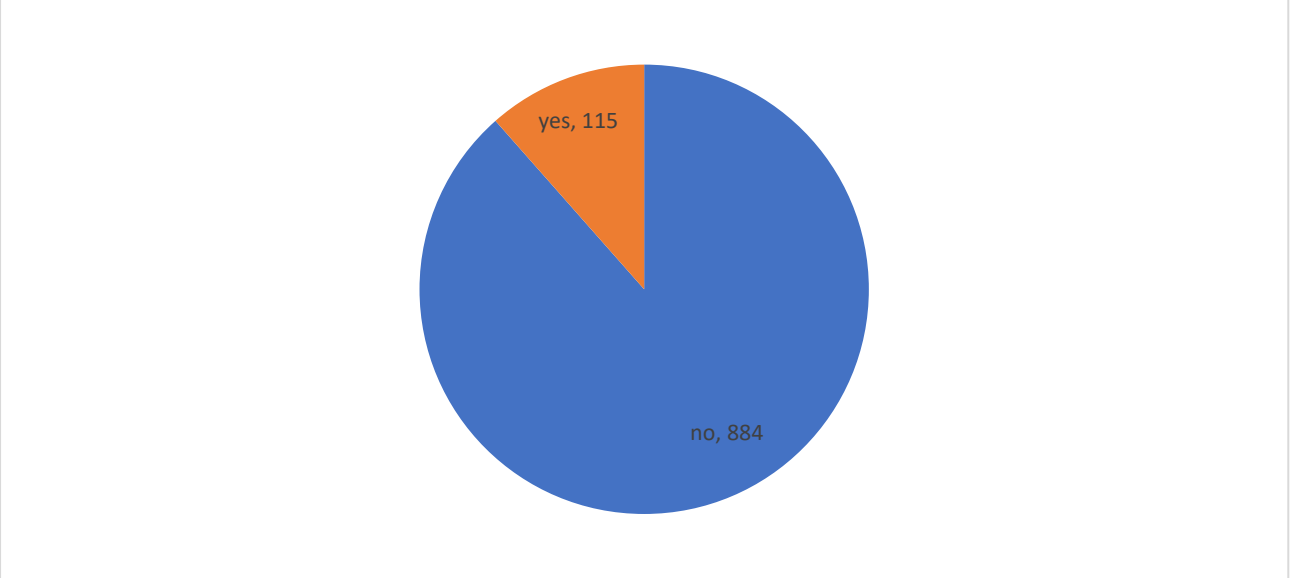
- Day: The last contact week is maximum on Monday with a count of 229 and minimum for Saturday and Tuesday.



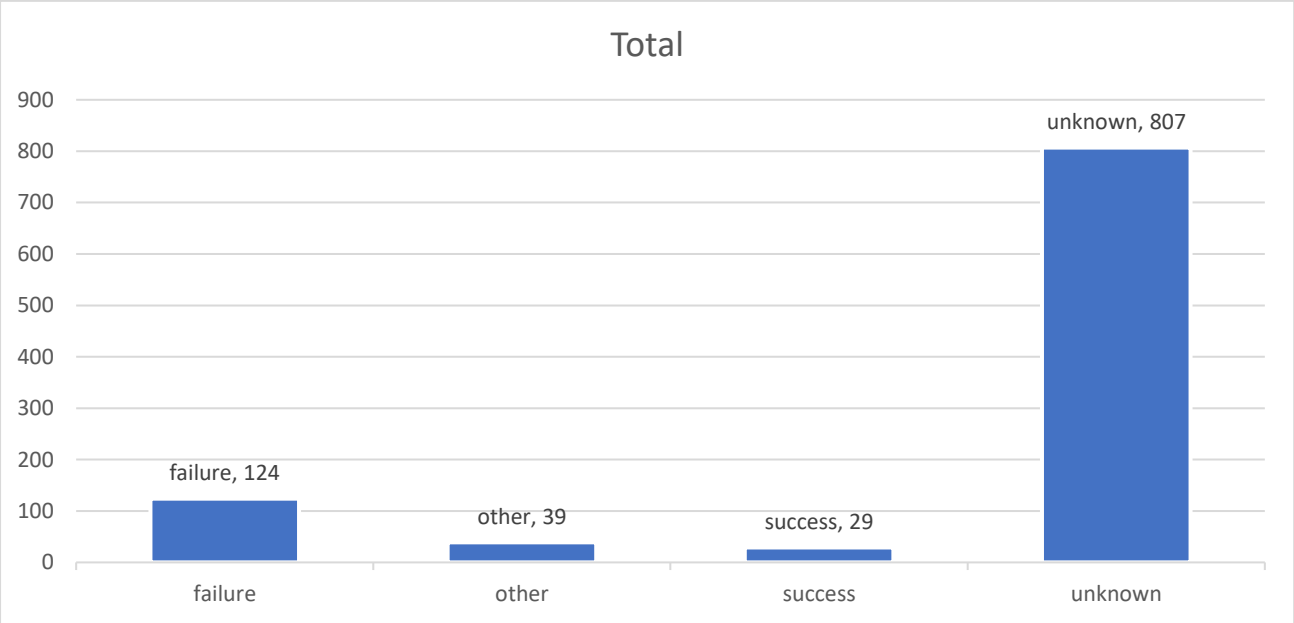
- Month: The last contact month of the year is with maximum calls is may with 281 as the record and minimum is in the month of march.



- Loan Deposit: Only 115 have applied to loan deposit and 884 have not applied to loan deposit.



- Pout comes: The result of previous campaign is maximum unknown and has only 3% success and 12% failure results.



b.) When we check for missing values, we see that balance has missing values hence we find out the median for the balance and replace the missing values with 459.The number of missing values are 17.

c.) We see that Contact and Balance has missing values and has been replaced with
Categorical: contact- most occurring contact medium is cellular.
Numerical: Balance – Median of the balance is 459 and hence been replaced.

2.)

a.) From the Pivot tables and descriptive statistics we observe that the mean age of customer is 41.

b.) The number of people less than the age of 45 is 649 and the percentage is 65%(rounding).

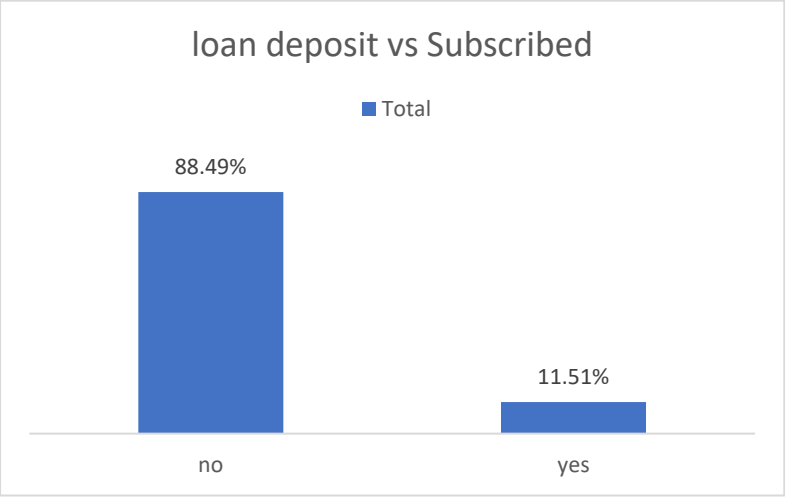
c.) The minimum and maximum from descriptive statistics:

min = -1680
max = 26965

3.)

a.) We find the pivot table of loan deposit where the maximum customer of 88.49% have denied loan and minimum customer 11.15% accepted loan.

Subscribed	Count of Loan deposit
no	88.49%
yes	11.51%
Grand Total	**100.00%**



b.) The categories for loan deposit:

Job Type	Count of Loan deposit
admin.	97
blue-collar	206
entrepreneur	93
management	221
Others	69
retired	52
services	75
technician	181
unknown	5
Grand Total	**999**

c.)From the above table it is highlighted that the management-based employees have the maximum number employees take loan deposit of 221.

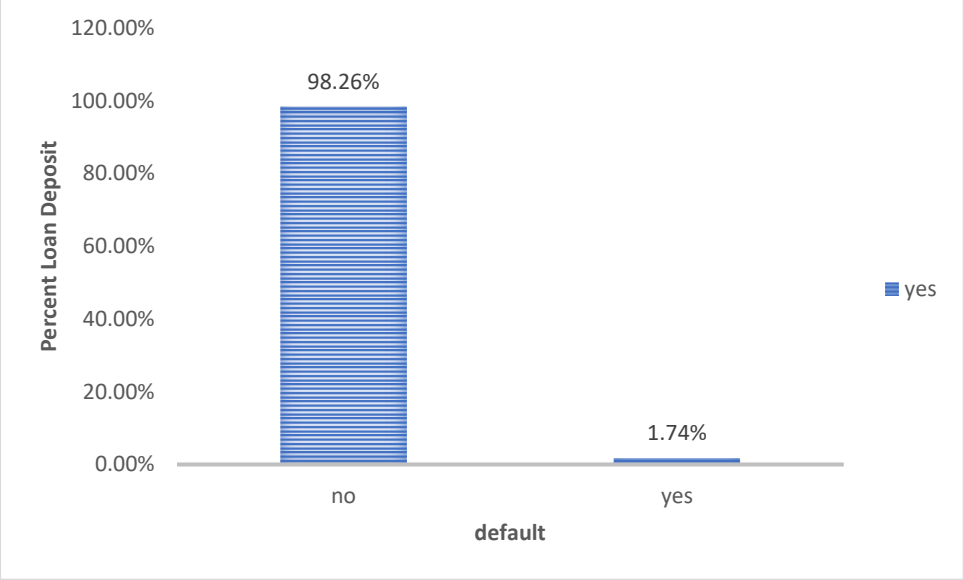
4.)

a.) Y- axis is the Percent loan deposit after referring the chart since default is given as x-axis the only possible outcome is loan deposit.

b.)To plot the graph after getting the pivot table.

given below: We find the pivot table and convert the basic numbers to percent of grand total to get the values. 98.26% people did not take the loan

Loan taken		Grand Total
no	98.26%	98.26%
yes	1.74%	1.74%
Grand Total	**100.00%**	**100.00%**



c.)The Category of people have taken loan deposit but 98.26% have no defaults while 1.74% have loan deposits.

5.)

a.) It looks like the people who use the telephonic mode of contact have taken more loan deposit than the people who have used cellular mode of contact. We see that 16.67% of people who use telephone take more loan deposits than cellular.

Count of Loan deposit	Column Labels

Mode of contact	no	yes	Grand Total
cellular	88.82%	11.18%	100.00%
telephone	83.33%	16.67%	100.00%
Grand Total	**88.49%**	**11.51%**	**100.00%**

b.) So from the given table below we can see that what category of people have accepted the loan deposits at that particular age. We have filtered out the people who haven't taken loan and found column percent of the required people. The highest number of loan deposit is 35.65% in the age group of 25-34

c.) We see that the management people have the greatest loan deposits of 11.30% from the age of 25-34 from the given table highlighted.

type of job	15-24	25-34	35-44	45-54	55-64	65-74	75-84
admin.	0.00%	3.48%	1.74%	4.35%	3.48%	0.00%	0.00%
blue-collar	0.87%	8.70%	2.61%	3.48%	0.00%	0.00%	0.00%
entrepreneur	0.00%	1.74%	4.35%	2.61%	2.61%	0.00%	0.00%
management	0.00%	11.30%	7.83%	4.35%	0.87%	0.00%	0.00%
Others	2.61%	3.48%	0.87%	0.00%	1.74%	0.00%	0.00%
retired	0.00%	0.00%	0.00%	0.00%	3.48%	0.87%	4.35%
services	0.00%	4.35%	0.87%	0.87%	0.00%	0.00%	0.00%
technician	0.00%	2.61%	6.09%	1.74%	1.74%	0.00%	0.00%
Grand Total	**3.48%**	**35.65%**	**24.35%**	**17.39%**	**13.91%**	**0.87%**	**4.35%**

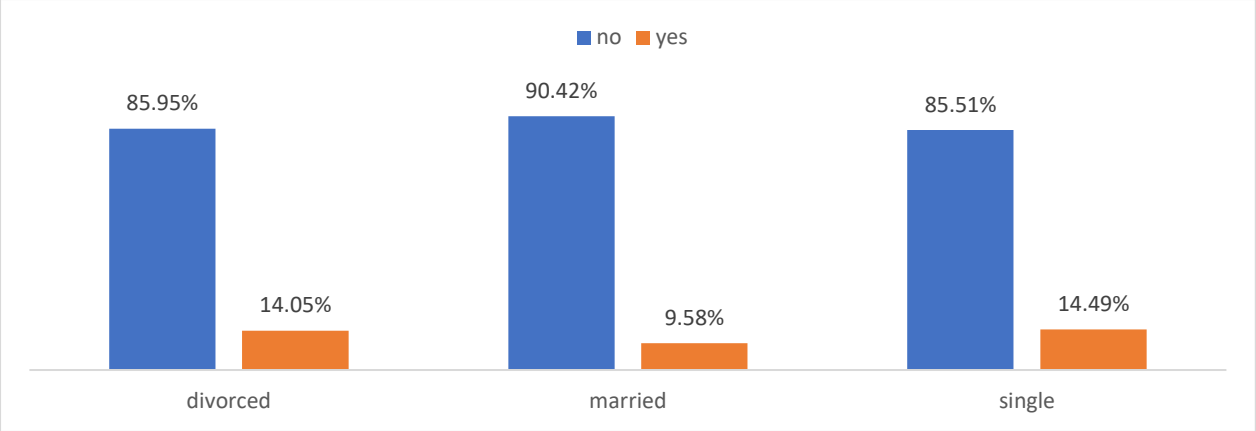
6.) PART-1

a.) The hypothesis statement: "customers who are married are a better target audience for a loan deposit when compared to customers who are not married."

From the excel we get the pivot table where we can infer which category has accepted the loan deposits.

Marital Status	no	yes
divorced	85.95%	14.05%
married	90.42%	9.58%
single	85.51%	14.49%
Grand Total	**88.49%**	**11.51%**

b.) Visualize that both divorce and single people combine take more home loans than married people that is 28% in total.



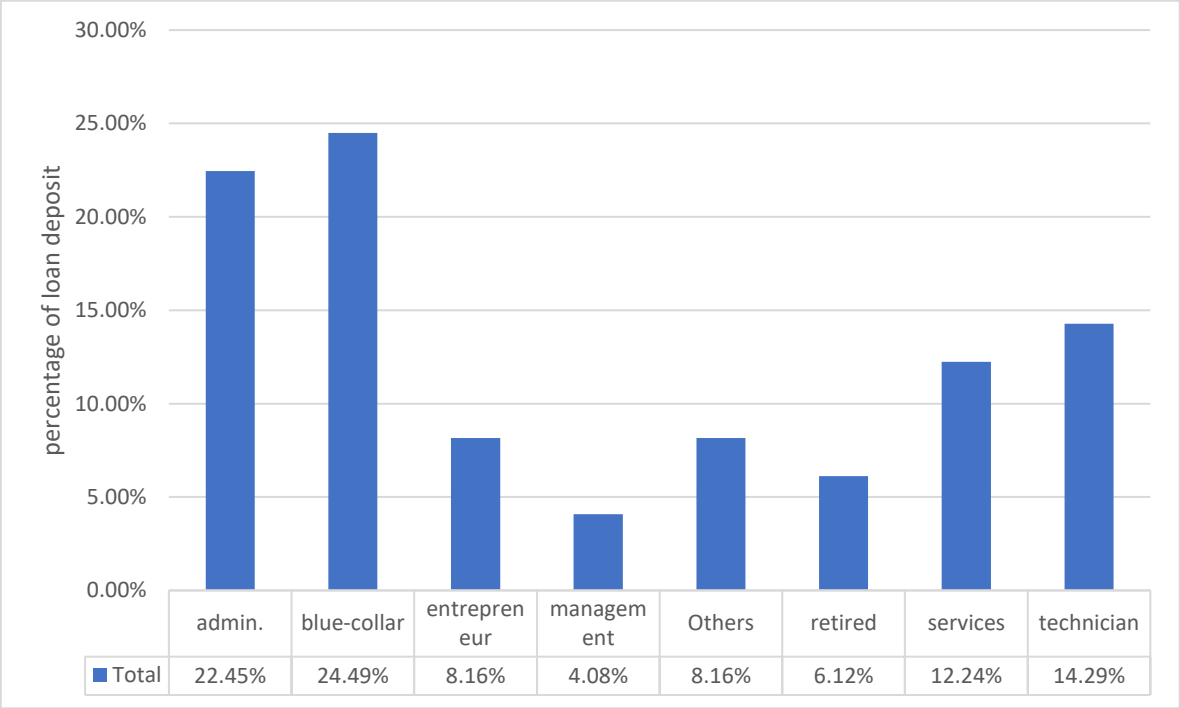
d.) The hypothesis in the given statement as seen is false since we see that 28.5% both single and divorced people combined take more loan deposits than the married people that is 9%.

PART-2

a.) The hypothesis statement: "Blue collar customers with secondary level of education are more likely to make a loan deposit when compared to other category of jobs."
Creating the pivot table of Job and who took the most deposits. From the table we see the blue-collar people who take most loans.

job type	Count of Loan deposit
admin.	22.45%
blue-collar	24.49%
entrepreneur	8.16%
management	4.08%
Others	8.16%
retired	6.12%
services	12.24%
technician	14.29%

b.) Visualization to infer which category took most loans



c.) Now we can say that the hypothesis is true since the blue-collar people with secondary level of education have the highest level of loan deposit percent, after filtering the pivot table. The blue collar has 24.49% and the lowest is management.

PART 2- SQL

This capstone project is us writing queries given to us in this database and the given domain is Sports.

TASK/QUESTION:

1.) This question asks us to query the list of people who have ever been seeded regardless of them being in singles or doubles. Also, these players have different registration number for both doubles and singles.

CODE:

```
-- question 1
select distinct p.name, p.pid from player p
join registration r using(pid)
join playedin pi using(RegistrNum)
where pi.seed is not NULL;
```

From the above code we see that initially we added the database to my SQL to carry out our code. We tend to use distinct to remove any repetition. We take the names from player p and PID and join with registration using PID and join with played in . Where we mention that we must seed in played in is not null.

RESULT:

name	PID
Roger Federer	0
Nicolas Almagro	31
Lleyton Hewitt	18
Fernando Gonzalez	9
Andy Murray	14
Jarkko Nieminen	16
Stanislas Wawrinka	30
Rafael Nadal	1
Novak Djokovic	13
Marcos Baghdatis	10
Richard Gasquet	17
Jose Acasuso	26
Tommy Robredo	6
Ivan Ljubicic	3
Agustin Calleri	29
Radek Stepanek	19
David Ferrer	15
Mario Ancic	8
Dominik Hrbaty	21
Marat Safin	25
Mikhail Youzhny	24
Andy Roddick	5
David Nalbandian	7
Sebastien Grosjean	27
Robin Soderling	22
Tommy Haas	11
Juan Carlos Ferrero	23
Tomas Berdych	12
Dmitry Tursunov	20
Xavier Malisse	28
Nikolay Davydenko	2
James Blake	4
Mike Bryan	129
Bob Bryan	128
Todd Perry	148
Wesley Moodie	147
Jaroslav Levinsky	146
Frantisek Cermak	145
Marcin Matkowski	140
Mariusz Furstenberg	139
Daniel Nestor	131
Mark Knowles	130
Tripp Phillips	152
Ashley Fisher	151

From the above we find the result of all people that are seeded for the matches where have 95 rows returned along with their PID.

2.) In this we must query of how to get the list of tournaments with more than 5 rounds as well as print the name of the tournament, the tournament type, the start and end dates, and the number of rounds.

CODE:

```
-- question 2
select name, TType, StartDate, EndDate, NumRounds from tournament
where NumRounds > 5;
```

From the above code we see that we select the required fields to be printed in the table from the tournament and where we use the function > so that we can filter out the number of rounds greater than 5.

RESULT:

name	TType	StartDate	EndDate	NumRounds
Australian Open	Singles	15-01-2007	28-01-2007	7
Australian Open	Doubles	15-01-2007	28-01-2007	6
French Open	Singles	26-05-2007	10-06-2007	7
French Open	Doubles	26-05-2007	10-06-2007	6
US Open	Singles	27-08-2007	09-09-2007	7
US Open	Doubles	27-08-2007	09-09-2007	6
Wimbledon	Singles	25-06-2007	08-07-2007	7
Wimbledon	Doubles	25-06-2007	08-07-2007	6

From the above the result where 8 rows are returned, we see that we get all the names of tournament and their type along with the mentioned dates and which all matches have more than 5 tournaments.

3.) This task has 2 parts to it where we must list out name, tournament type, surface type, and the number of rounds it has and sort the results in descending order by the number of rounds.

CODE:

```
-- question3
select name, TType, NumRounds, Surface
from tournament
order by NumRounds desc;
```

From the above code we can see that all we need to do is use the **order by** function so that we can order the number of rounds by descending order. Also print out the type of match, surface from tournament table.

name	TType	NumRounds	Surface
Australian Open	Singles	7	Hard
French Open	Singles	7	Clay
US Open	Singles	7	Hard
Wimbledon	Singles	7	Grass
Australian Open	Doubles	6	Hard
French Open	Doubles	6	Clay
US Open	Doubles	6	Hard
Wimbledon	Doubles	6	Grass
Brasil Open 2007	Singles	5	Clay
Countrywide Classic	Singles	5	Hard
BMW Open	Singles	5	Clay
Heineken Open	Singles	5	Hard
Brasil Open 2007	Doubles	4	Clay
Countrywide Classic	Doubles	4	Hard
BMW Open	Doubles	4	Clay
Heineken Open	Doubles	4	Hard

From the above result we receive 16 rows with the number of rounds in descending order of all the games singles and doubles.

4.) In this task we must list the names, tournament types, and lengths (in days) of all tournaments that were longer than one week.

CODE:

```
-- question 4
select name, ttype, datediff(enddate,startdate) as length_in_days
from tournament
having length_in_days > 7;
```

From the above code we see how we managed to get the difference between the 2 dates that by using **datediff** and name it as length_in_days which we can take from the tournaments and mention that the number of days should be greater than 7.(LENGTH_IN_DAYS)

RESULT:

name	ttype	length_in_days
Australian Open	Singles	13
Australian Open	Doubles	13
French Open	Singles	15
French Open	Doubles	15
US Open	Singles	13
US Open	Doubles	13
Wimbledon	Singles	13
Wimbledon	Doubles	13

We get the above table after executing the code and in return we get 8 rows that have the length of days for both double and single matches.

5.) This is a very interesting task where we are told to find out the players in both single and double tournament who have played against Tommy Haas from the country.

CODE:

```
-- question 5
create view th_registr as
select registrnum from registration where pid = (select pid from player where name = 'Tommy Haas');

select * from th_registr;
create view against_haas as
select registrnum1 from matches where registrnum1 not in(select * from th_registr) and registrnum2 in(select * from th_registr) union
select registrnum2 from matches where registrnum2 not in(select* from th_registr) and registrnum1 in(select *from th_registr);

select * from against_haas;

select distinct name, pid from player
join registration using(pid)
where registrnum in (select * from against_haas) and ccode in('RUS','CHI','USA');
```

In this question we tend to use method of creating separate tables like registr where we get registration number of tommy Haas and every game he played from registration table.

We create another table against_haas to find the people who played against Tommy Haas. In this we check for both the register number 1 and 2 with the table we created that is registr whether either of the registered number is of tommy Haas. So first we select registered number 1 from matches where this is not in the table registr (which has all registered number of Tommy Haas) and select the registered number 2 from registr; this will give us all registernum2 of Tommy Haas and his opponent which is registered number1 vice versa. We union both the table to get all the players

If the registered number of Tommy Haas is available, we can extract that making sure that the other registered number is not of Tommy Haas, but the opponent player and we do vice versa. From this we can get registered number of the opponent player.

Once both the tables are created, we can code the part where we want the names of these players. We use distinct so that no player is repeated. Join with registration using PID. Now we extract the registered number of opponent player from against Haas and making sure that these players belong from Russia, Chile, USA. This helps us extract the names of these player and give us the required result.

name	pid
Fernando Gonzalez	9
Nikolay Davydenko	2
James Blake	4
Dmitry Tursunov	20
Zack Fleishman	61

This is the result that we achieve that shows the PID and name of the player who played against Tommy Haas. Creating tables becomes easier since we don't have to complicate the solution by adding subquery since it makes the code more efficient and understandable.

6.) Again, this question has two parts where we must get the name of all players who have lost again Roger Federer and the tournament name that they lost in. In this question we must find out all the final rounds of the tournament in which Roger Federer has won.

CODE:

```
-- question 6

create view match_ids as
select mid from matchresults mr
where winner in (select registrnum from registration where pid =(select pid from player where name = "Roger Federer")) union
select mid from tiebreaker
where winnertb = 0;

select * from match_ids;

create view mt_federer as
select mid, registrnum1, registrnum2, tid from matches join match_ids using(mid);

select * from mt_federer;

create view type_single as
select name,numrounds,ttype,m.*
from matches m join tournament t using (tid)
where m.round=t.numrounds and ttype='Singles';

select * from type_single;

create view finals as
select s.* from type_single s join tournament t using(tid)
join mt_federer using(mid);
select * from finals;

select p.name, f.name from player p
join registration r using(PID)
join finals f where  f.registrnum2 = r.registrnum;
```

We create 4 different tables to get our result here. Create match id table where we select the mid from match result where we mention that the matchresult must be the winning match of Roger Federer. Hence we mention to take registration number and mention whose pid we want and we also take the mid from the tiebreaker where we get all the tiebreaking winning match of Roger Federer. Hence get all the winning matches of Roger Federer.

We create a second table mt_federer where we get all the mid, registration number of Federer and his opponent and tournament id of all the matches winning matches.

From the question it has also been mentioned that the winning games must be of type singles. Upon this condition we create another table. This table is type single where we take the name, number of round type and entire match field and join with the tournament where we must satisfy the condition that the match type is single, and we extract the last round of the single matches.

We create one last table which will give us the name of the winning tournament and its type being single, final rounds (total rounds) as well as the registration number of Federer and opponent. We join the type single with mt_Federer. We give this name as finals.

Lastly, we extract the name from player to get player names and finals table to get the tournament names we join it with registration using PID and join finals where we put the condition that all the names should be for these registration number.

Hence, we get the result:

RESULT:

Column1	Column2
name	name
Fernando Gonzalez	Australian Open
Novak Djokovic	US Open
Rafael Nadal	Wimbledon

From the above result we how we got all the three opponents for a type of single match and all the tournament names. Creating tables is a better method to reduce the usage of subquery.