

## Sql queries

1. Total customers

**Total Customers**  
39717

```
mysql> select count(id) from finance_1;
+-----+
| count(id) |
+-----+
|      39717 |
+-----+
```

2. Max Interest

**Max Interest**  
24.59%

```
mysql> select max(int_rate) from finance_1;
+-----+
| max(int_rate) |
+-----+
|      24.59 |
+-----+
1 row in set (0.03 sec)
```

3. Min Interest

**Min Interest**  
5.42%

```
mysql> select min(int_rate) from finance_1;
+-----+
| min(int_rate) |
+-----+
|      5.42 |
+-----+
1 row in set (0.03 sec)
```

4. Recovery rate

**Recovery Rate**  
80.58%

```
mysql> select CONCAT(ROUND((SUM(total_rec_prncp)/SUM(Total_pymnt)*100),2),'%') as Recovery_Rate from
finance_2;
+-----+
| Recovery_Rate |
+-----+
| 80.58%       |
+-----+
1 row in set (0.03 sec)
```

5. Enquiry in last 6 months

### Enquiry In 6 Months

34522

```
mysql> select sum(inq_last_6mths) from finance_2;
+-----+
| sum(inq_last_6mths) |
+-----+
|          34522 |
+-----+
1 row in set (0.02 sec)
```

### 6. Term slicer

#### Term

36 months      60 months

```
mysql> select DISTINCT term from finance_1;
+-----+
| term      |
+-----+
| 36 months |
| 60 months |
+-----+
```

### 7. Loan Status Slicer

#### Loan Status

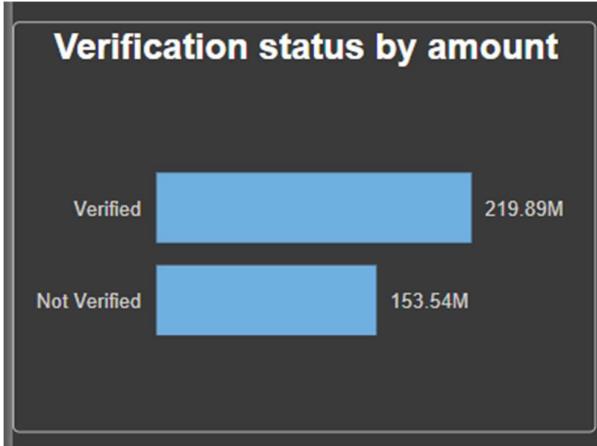
Charged Off

Current

Fully Paid

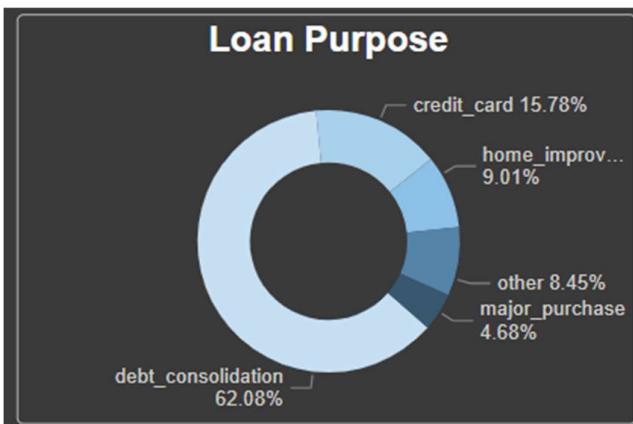
```
mysql> select DISTINCT loan_status from finance_1;
+-----+
| loan_status |
+-----+
| Fully Paid |
| Charged Off |
| Current     |
+-----+
```

### 8. Verification status by amount



```
mysql> select ab.verification_status as 'Status',ROUND(sum(qr.total_pymnt)) as 'Total Payment' from finance_2 qr inner join finance_1 ab on qr.id = ab.id group by verification_status having (verification_status = 'verified' or verification_status='not verified');
+-----+-----+
| Status | Total Payment |
+-----+-----+
| Verified | 219892308 |
| Not Verified | 153541418 |
+-----+-----+
```

#### 9. Top 5 loan purpose



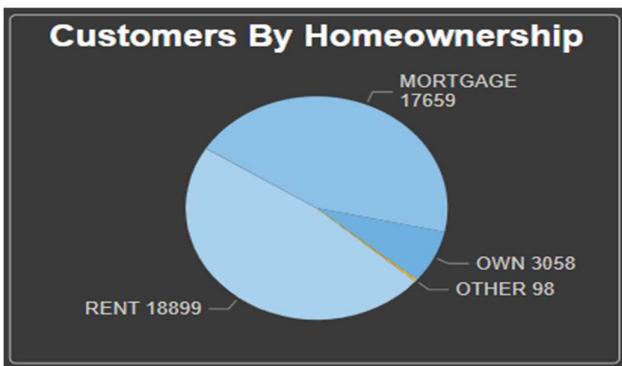
```
mysql> select COUNT(ID) as Total_IDs,
    -> purpose from finance_1
    -> where purpose in ("debt_consolidation","credit_card","car","home_improvement","small_business","major_purpose","moving","house","vacation","educational","wedding","medical","renewable_energy","other")
    -> group by purpose
    -> order by Total_ids desc;
+-----+-----+
| Total_IDs | purpose |
+-----+-----+
| 18641 | debt_consolidation |
| 5130 | credit_card |
| 3993 | other |
| 2976 | home_improvement |
| 1828 | small_business |
| 1549 | car |
| 947 | wedding |
| 693 | medical |
| 583 | moving |
| 381 | house |
| 381 | vacation |
| 325 | educational |
| 103 | renewable_energy |
+-----+-----+
13 rows in set (0.07 sec)
```

```

mysql> select COUNT(ID) as Total_IDs,
-> purpose from finance_1
-> where purpose in ("debt_consolidation","credit_card","car","home_improvement","small_business",
-> "major_purpose","moving","house","vacation","educational","wedding","medical","renewable_energy",
-> "other")
-> group by purpose
-> order by Total_ids desc
-> limit 5;
+-----+-----+
| Total_IDs | purpose |
+-----+-----+
| 18641 | debt_consolidation |
| 5130 | credit_card |
| 3993 | other |
| 2976 | home_improvement |
| 1828 | small_business |
+-----+-----+
5 rows in set (0.08 sec)

```

## 10. Customers by homeownership

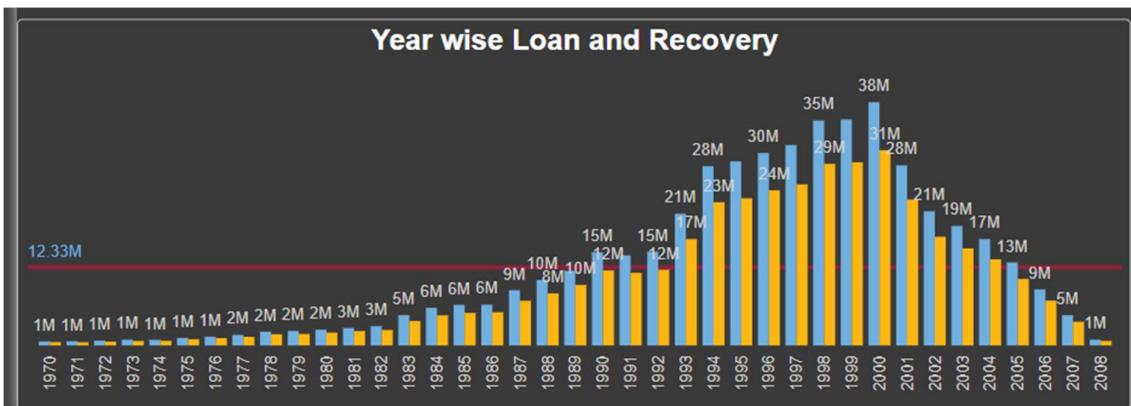


```

mysql> select COUNT(ID) as Total_IDs, Home_ownership from finance_1 where home_ownership in
("mortgage","rent","own","other") group by home_ownership;
+-----+-----+
| Total_IDs | Home_ownership |
+-----+-----+
| 18899 | RENT |
| 3058 | OWN |
| 17659 | MORTGAGE |
| 98 | OTHER |
+-----+-----+

```

## 11. Year wise loan and recovery



```

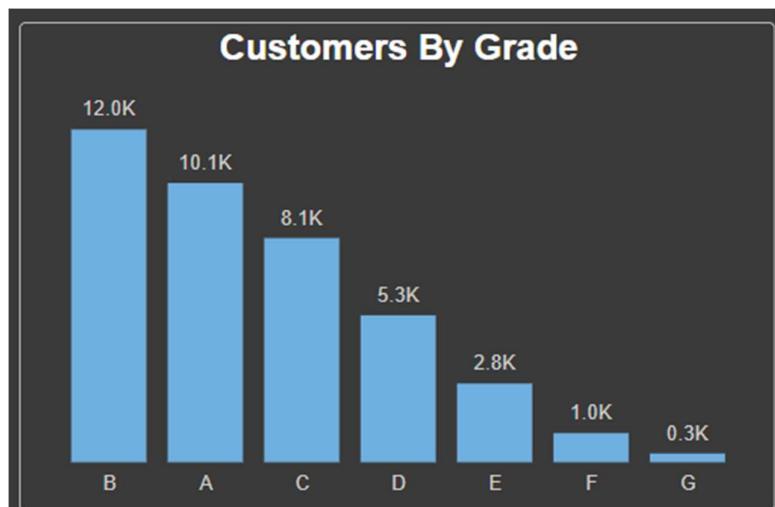
mysql> select DISTINCT left(earliest_cr_line, 4) As Years,
->      sum(total_pymnt) as Total_payment,
->      sum(total_rec_prncp) as Total_Recovery
->      from finance_2
->      group by years
->      order by years asc;
+-----+-----+-----+
| Years | Total_payment | Total_Recovery |
+-----+-----+-----+
| 1946  | 7029.694131   | 6000          |
| 1950  | 14601.24      | 14475         |
| 1954  | 23935.406294  | 19800         |
| 1956  | 17611.52692   | 15000         |
| 1959  | 6514.520632   | 6000          |
| 1961  | 31893.338973  | 27000         |
| 1962  | 81560.97961   | 65217.86      |
| 1963  | 71596.193922  | 61205.34      |
| 1964  | 114780.27705199999 | 89620.36      |
| 1965  | 175927.153215  | 138605.25     |
| 1966  | 163264.449066  | 128795.44     |
| 1967  | 207705.03822200003 | 152721.5      |
| 1968  | 271964.05168   | 195369.82     |
| 1969  | 505335.571544  | 425725.38     |
| 1970  | 634182.2618640002 | 518741.98     |
| 1971  | 669336.1734259999 | 510043.5599999994 |
| 1972  | 761643.8054160002 | 599282.66     |
| 1973  | 909570.9603770002 | 711434.96     |
| 1974  | 876878.5129430003 | 729417.0199999999 |
| 1975  | 1189641.9068439999 | 990068.5      |
| 1976  | 1388283.1152300003 | 1158590       |
| 1977  | 1663704.776308001  | 1359989.14    |
| 1978  | 2173188.4563920004 | 1775512.9100000004 |
| 1979  | 2282333.9496960007 | 1815998.37    |
| 1980  | 2482911.581481   | 2029580.6600000001 |
| 1981  | 2750211.039102001 | 2266926.67    |
| 1982  | 3035418.971888001 | 2429797.5700000003 |
| 1983  | 4779291.5254104985 | 3864653.57    |
| 1984  | 5931109.635597996 | 4749930.47    |
| 1985  | 6403355.898277992 | 5131735.8100000005 |

```

1986	6435356.543083002	5240099.819999999
1987	8713226.69431001	7036481.740000001
1988	10337477.002311008	8207064.819999996
1989	11767982.982844707	9536513.240000004
1990	14687098.10811399	11825052.99
1991	14212785.325833973	11455854.700000001
1992	14795667.24680799	11936831.030000005
1993	20761152.744623017	16768090.880000003
1994	28255913.559224717	22584326.590000007
1995	29020117.958615527	23212411.020000026
1996	30336094.445617903	24464843.650000036
1997	31588374.332349986	25390029.660000004
1998	35451968.57676207	28620362.720000006
1999	35635801.45350508	28874426.749999996
2000	38346518.98755418	30784861.880000006
2001	28399201.65253414	23008263.64999997
2002	21148549.680101003	17126929.45
2003	18835694.987898197	15302794.390000004
2004	16764612.494438104	13585600.070000019
2005	13085425.313202985	10493660.149999999
2006	8833321.939363007	7069812.729999995
2007	4767811.066175998	3732219.339999994
2008	899458.816603	718663.739999996

53 rows in set (0.05 sec)

## 12.Customer by grade

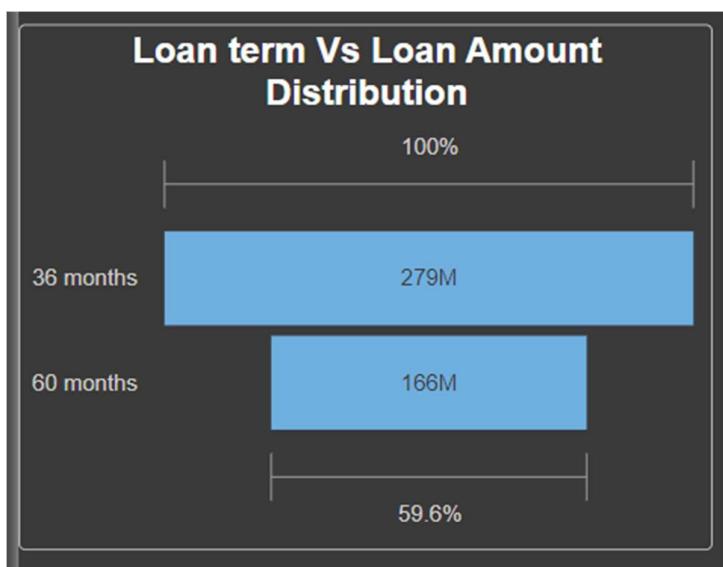


```

mysql> select grade, COUNT(ID) as Total_id
-> from finance_1
-> group by grade
-> order by Total_id desc;
+-----+-----+
| grade | Total_id |
+-----+-----+
| B     | 12020  |
| A     | 10085  |
| C     | 8098   |
| D     | 5307   |
| E     | 2842   |
| F     | 1049   |
| G     | 316    |
+-----+-----+
7 rows in set (0.05 sec)

```

### 13.Loa term vs loan amount distribution



```

mysql> UPDATE finance_1
-> SET loan_amnt = REPLACE(loan_amnt, '$', '')
-> ;
Query OK, 39717 rows affected (1.91 sec)
Rows matched: 39717  Changed: 39717  Warnings: 0

```

```

mysql> UPDATE finance_1 SET loan_amnt = REPLACE(loan_amnt, ',', '');
Query OK, 39705 rows affected (1.94 sec)
Rows matched: 39717  Changed: 39705  Warnings: 0

```

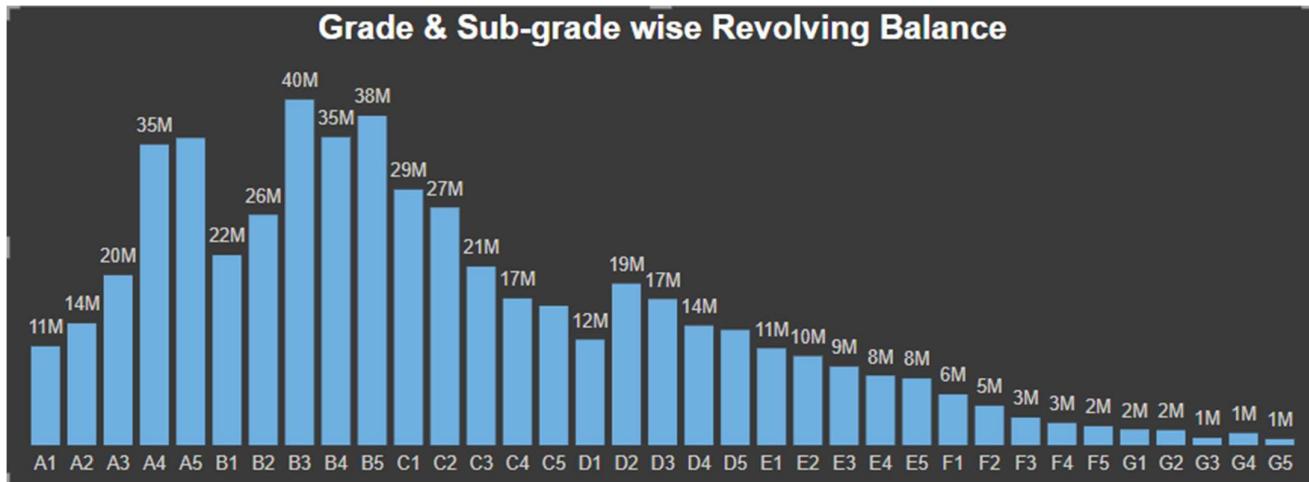
```

mysql> ALTER TABLE finance_1 MODIFY loan_amnt INT;
Query OK, 39717 rows affected (0.47 sec)
Records: 39717  Duplicates: 0  Warnings: 0

```

```
mysql> select (term), SUM(loan_amnt) from finance_1 GROUP BY term;
+-----+-----+
| term | SUM(loan_amnt) |
+-----+-----+
| 36 months | 279118475 |
| 60 months | 166486575 |
+-----+-----+
```

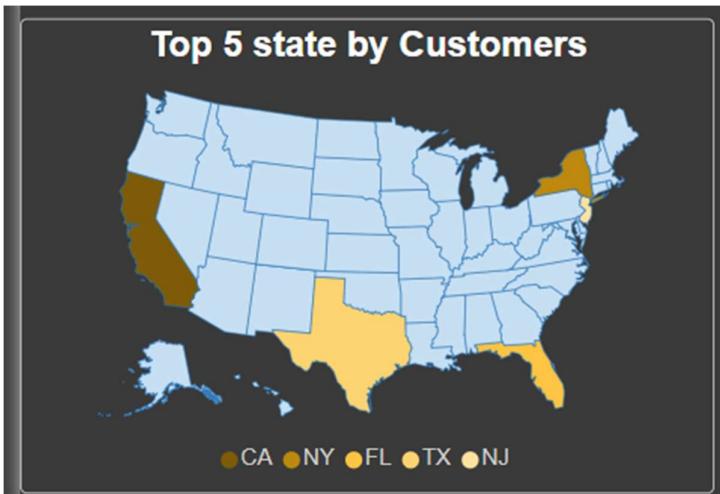
14. Grade and sub grade wise loan amount



```
mysql> Select ab.sub_grade as 'Grade and Sub grade',SUM(qr.revol_bal) as 'Revolving Balance' from finance_2
qr inner join finance_1 ab on qr.id = ab.id group by sub_grade order by sub_grade desc;
+-----+-----+
| Grade and Sub grade | Revolving Balance |
+-----+-----+
| G5 | 701515 |
| G4 | 1390628 |
| G3 | 832193 |
| G2 | 1729627 |
| G1 | 1808763 |
| F5 | 2187323 |
| F4 | 2551064 |
| F3 | 3175435 |
| F2 | 4528248 |
| F1 | 5840746 |
| E5 | 7669868 |
| E4 | 7990991 |
| E3 | 9039059 |
| E2 | 10242033 |
| E1 | 11132588 |
| D5 | 13252474 |
| D4 | 13742947 |
| D3 | 16793781 |
| D2 | 18570972 |
| D1 | 12130255 |
| C5 | 16015609 |
| C4 | 16867691 |
| C3 | 20531370 |
| C2 | 27321114 |
| C1 | 29384926 |
| B5 | 37858666 |
| B4 | 35405811 |
```

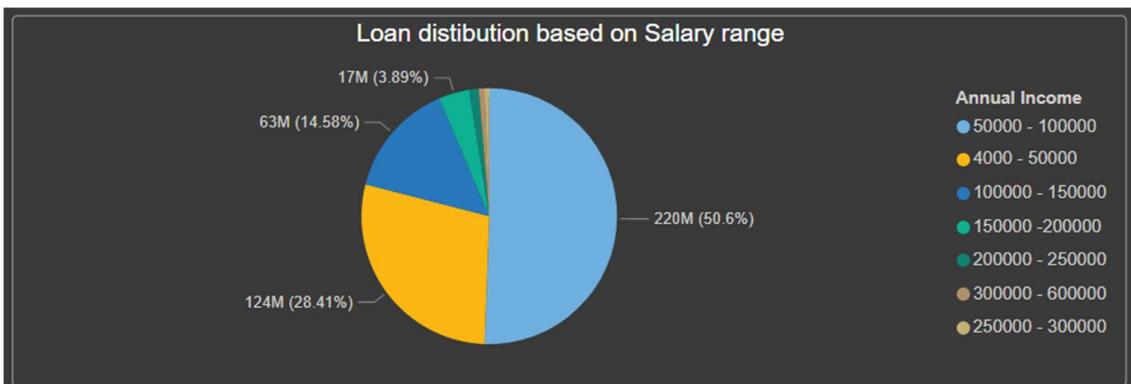
B3	39723554
B2	26478439
B1	21842079
A5	35303045
A4	34557156
A3	19543922
A2	14004780
A1	11365196

15.Top 5 state by customer



```
mysql> select addr_state as State, COUNT(id) as Total_no_of_Customers
-> from finance_1
-> group by addr_state
-> order by Total_no_of_Customers
-> desc
-> limit 5;
+-----+-----+
| State | Total_no_of_Customers |
+-----+-----+
| CA    |          7099 |
| NY    |          3812 |
| FL    |          2866 |
| TX    |          2727 |
| NJ    |          1850 |
+-----+-----+
5 rows in set (0.05 sec)
```

16.Loan distribution based on salary range



```
mysql> alter table finance_1 add column Salary_range int;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

```
mysql> alter table finance_1 MODIFY column Salary_range varchar(20);
Query OK, 39717 rows affected (0.25 sec)
Records: 39717  Duplicates: 0  Warnings: 0
```

```
mysql> update finance_1 SET annual_inc = REPLACE(annual_inc,'$', ' ');
Query OK, 39717 rows affected (0.66 sec)
Rows matched: 39717  Changed: 39717  Warnings: 0
```

```
mysql> update finance_1 SET annual_inc = REPLACE(annual_inc,',', ' ');
Query OK, 39717 rows affected (0.76 sec)
Rows matched: 39717  Changed: 39717  Warnings: 0
```

```
mysql> update finance_1
      -> set salary_range =
      -> CASE
      ->     WHEN annual_inc BETWEEN 4000 AND 50000 THEN '4000-50000'
      ->     WHEN annual_inc BETWEEN 50000 AND 100000 THEN '50000-100000'
      ->     WHEN annual_inc BETWEEN 100000 AND 150000 THEN '100000-150000'
      ->     WHEN annual_inc BETWEEN 150000 AND 200000 THEN '150000-200000'
      ->     WHEN annual_inc BETWEEN 200000 AND 250000 THEN '200000-250000'
      ->     WHEN annual_inc BETWEEN 250000 AND 300000 THEN '250000-300000'
      ->     WHEN annual_inc BETWEEN 300000 AND 600000 THEN '300000-600000'
      ->     ELSE 'Above 600000'
      -> END;
Query OK, 39717 rows affected (1.11 sec)
Rows matched: 39717  Changed: 39717  Warnings: 0
```

```
mysql> UPDATE finance_1 SET funded_amnt = REPLACE(funded_amnt, '$', '');
Query OK, 39717 rows affected (1.13 sec)
Rows matched: 39717  Changed: 39717  Warnings: 0
```

```
mysql> UPDATE finance_1 SET funded_amnt = REPLACE(funded_amnt, ',', '');
Query OK, 39705 rows affected (1.18 sec)
Rows matched: 39717  Changed: 39705  Warnings: 0
```

```
mysql> SELECT SUM(funded_amnt) as Loan_amount, Salary_range
      -> FROM finance_1
      -> GROUP BY Salary_range
      -> ORDER BY Loan_amount DESC;
+-----+-----+
| Loan_amount | Salary_range |
+-----+-----+
| 219988400  | 50000-100000 |
| 123534850  | 4000-50000  |
| 63419200   | 100000-150000|
| 16902275   | 150000-200000|
| 5209550    | 200000-250000|
| 2643150    | 250000-300000|
| 2411675    | 300000-600000|
| 701225     | Above 600000 |
+-----+-----+
8 rows in set (0.04 sec)
```

