**Consider the below relations with the data, design MySQL solutions to the queries.**

CREATE TABLE "AGENTS"

(

"AGENT\_CODE" CHAR(6) NOT NULL PRIMARY KEY,

"AGENT\_NAME" CHAR(40),

"WORKING\_AREA" CHAR(35),

"COMMISSION" NUMBER(10,2),

"PHONE\_NO" CHAR(15),

"COUNTRY" VARCHAR2(25)

);

INSERT INTO AGENTS VALUES ('A007', 'Ramasundar', 'Bangalore', '0.15', '077-25814763', '');

INSERT INTO AGENTS VALUES ('A003', 'Alex ', 'London', '0.13', '075-12458969', '');

INSERT INTO AGENTS VALUES ('A008', 'Alford', 'New York', '0.12', '044-25874365', '');

INSERT INTO AGENTS VALUES ('A011', 'Ravi Kumar', 'Bangalore', '0.15', '077-45625874', '');

INSERT INTO AGENTS VALUES ('A010', 'Santakumar', 'Chennai', '0.14', '007-22388644', '');

INSERT INTO AGENTS VALUES ('A012', 'Lucida', 'San Jose', '0.12', '044-52981425', '');

INSERT INTO AGENTS VALUES ('A005', 'Anderson', 'Brisban', '0.13', '045-21447739', '');

INSERT INTO AGENTS VALUES ('A001', 'Subbarao', 'Bangalore', '0.14', '077-12346674', '');

INSERT INTO AGENTS VALUES ('A002', 'Mukesh', 'Mumbai', '0.11', '029-12358964', '');

INSERT INTO AGENTS VALUES ('A006', 'McDen', 'London', '0.15', '078-22255588', '');

INSERT INTO AGENTS VALUES ('A004', 'Ivan', 'Torento', '0.15', '008-22544166', '');

INSERT INTO AGENTS VALUES ('A009', 'Benjamin', 'Hampshair', '0.11', '008-22536178', '');

CREATE TABLE "CUSTOMER"

( "CUST\_CODE" VARCHAR2(6) NOT NULL PRIMARY KEY,

"CUST\_NAME" VARCHAR2(40) NOT NULL,

"CUST\_CITY" CHAR(35),

"WORKING\_AREA" VARCHAR2(35) NOT NULL,

"CUST\_COUNTRY" VARCHAR2(20) NOT NULL,

"GRADE" NUMBER,

"OPENING\_AMT" NUMBER(12,2) NOT NULL,

"RECEIVE\_AMT" NUMBER(12,2) NOT NULL,

"PAYMENT\_AMT" NUMBER(12,2) NOT NULL,

"OUTSTANDING\_AMT" NUMBER(12,2) NOT NULL,

"PHONE\_NO" VARCHAR2(17) NOT NULL,

"AGENT\_CODE" CHAR(6) NOT NULL REFERENCES AGENTS

);

INSERT INTO CUSTOMER VALUES ('C00013', 'Holmes', 'London', 'London', 'UK', '2', '6000.00', '5000.00', '7000.00', '4000.00', 'BBBBBBB', 'A003');

INSERT INTO CUSTOMER VALUES ('C00001', 'Micheal', 'New York', 'New York', 'USA', '2', '3000.00', '5000.00', '2000.00', '6000.00', 'CCCCCCC', 'A008');

INSERT INTO CUSTOMER VALUES ('C00020', 'Albert', 'New York', 'New York', 'USA', '3', '5000.00', '7000.00', '6000.00', '6000.00', 'BBBBSBB', 'A008');

INSERT INTO CUSTOMER VALUES ('C00025', 'Ravindran', 'Bangalore', 'Bangalore', 'India', '2', '5000.00', '7000.00', '4000.00', '8000.00', 'AVAVAVA', 'A011');

INSERT INTO CUSTOMER VALUES ('C00024', 'Cook', 'London', 'London', 'UK', '2', '4000.00', '9000.00', '7000.00', '6000.00', 'FSDDSDF', 'A006');

INSERT INTO CUSTOMER VALUES ('C00015', 'Stuart', 'London', 'London', 'UK', '1', '6000.00', '8000.00', '3000.00', '11000.00', 'GFSGERS', 'A003');

INSERT INTO CUSTOMER VALUES ('C00002', 'Bolt', 'New York', 'New York', 'USA', '3', '5000.00', '7000.00', '9000.00', '3000.00', 'DDNRDRH', 'A008');

INSERT INTO CUSTOMER VALUES ('C00018', 'Fleming', 'Brisban', 'Brisban', 'Australia', '2', '7000.00', '7000.00', '9000.00', '5000.00', 'NHBGVFC', 'A005');

INSERT INTO CUSTOMER VALUES ('C00021', 'Jacks', 'Brisban', 'Brisban', 'Australia', '1', '7000.00', '7000.00', '7000.00', '7000.00', 'WERTGDF', 'A005');

INSERT INTO CUSTOMER VALUES ('C00019', 'Yearannaidu', 'Chennai', 'Chennai', 'India', '1', '8000.00', '7000.00', '7000.00', '8000.00', 'ZZZZBFV', 'A010');

INSERT INTO CUSTOMER VALUES ('C00005', 'Sasikant', 'Mumbai', 'Mumbai', 'India', '1', '7000.00', '11000.00', '7000.00', '11000.00', '147-25896312', 'A002');

INSERT INTO CUSTOMER VALUES ('C00007', 'Ramanathan', 'Chennai', 'Chennai', 'India', '1', '7000.00', '11000.00', '9000.00', '9000.00', 'GHRDWSD', 'A010');

INSERT INTO CUSTOMER VALUES ('C00022', 'Avinash', 'Mumbai', 'Mumbai', 'India', '2', '7000.00', '11000.00', '9000.00', '9000.00', '113-12345678','A002');

INSERT INTO CUSTOMER VALUES ('C00004', 'Winston', 'Brisban', 'Brisban', 'Australia', '1', '5000.00', '8000.00', '7000.00', '6000.00', 'AAAAAAA', 'A005');

INSERT INTO CUSTOMER VALUES ('C00023', 'Karl', 'London', 'London', 'UK', '0', '4000.00', '6000.00', '7000.00', '3000.00', 'AAAABAA', 'A006');

INSERT INTO CUSTOMER VALUES ('C00006', 'Shilton', 'Torento', 'Torento', 'Canada', '1', '10000.00', '7000.00', '6000.00', '11000.00', 'DDDDDDD', 'A004');

INSERT INTO CUSTOMER VALUES ('C00010', 'Charles', 'Hampshair', 'Hampshair', 'UK', '3', '6000.00', '4000.00', '5000.00', '5000.00', 'MMMMMMM', 'A009');

INSERT INTO CUSTOMER VALUES ('C00017', 'Srinivas', 'Bangalore', 'Bangalore', 'India', '2', '8000.00', '4000.00', '3000.00', '9000.00', 'AAAAAAB', 'A007');

INSERT INTO CUSTOMER VALUES ('C00012', 'Steven', 'San Jose', 'San Jose', 'USA', '1', '5000.00', '7000.00', '9000.00', '3000.00', 'KRFYGJK', 'A012');

INSERT INTO CUSTOMER VALUES ('C00008', 'Karolina', 'Torento', 'Torento', 'Canada', '1', '7000.00', '7000.00', '9000.00', '5000.00', 'HJKORED', 'A004');

INSERT INTO CUSTOMER VALUES ('C00003', 'Martin', 'Torento', 'Torento', 'Canada', '2', '8000.00', '7000.00', '7000.00', '8000.00', 'MJYURFD', 'A004');

INSERT INTO CUSTOMER VALUES ('C00009', 'Ramesh', 'Mumbai', 'Mumbai', 'India', '3', '8000.00', '7000.00', '3000.00', '12000.00', 'Phone No', 'A002');

INSERT INTO CUSTOMER VALUES ('C00014', 'Rangarappa', 'Bangalore', 'Bangalore', 'India', '2', '8000.00', '11000.00', '7000.00', '12000.00', 'AAAATGF', 'A001');

INSERT INTO CUSTOMER VALUES ('C00016', 'Venkatpati', 'Bangalore', 'Bangalore', 'India', '2', '8000.00', '11000.00', '7000.00', '12000.00', 'JRTVFDD', 'A007');

INSERT INTO CUSTOMER VALUES ('C00011', 'Sundariya', 'Chennai', 'Chennai', 'India', '3', '7000.00', '11000.00', '7000.00', '11000.00', 'PPHGRTS', 'A010');

CREATE TABLE "ORDERS"

(

"ORD\_NUM" NUMBER(6,0) NOT NULL PRIMARY KEY,

"ORD\_AMOUNT" NUMBER(12,2) NOT NULL,

"ADVANCE\_AMOUNT" NUMBER(12,2) NOT NULL,

"ORD\_DATE" DATE NOT NULL,

"CUST\_CODE" VARCHAR2(6) NOT NULL REFERENCES CUSTOMER,

"AGENT\_CODE" CHAR(6) NOT NULL REFERENCES AGENTS,

"ORD\_DESCRIPTION" VARCHAR2(60) NOT NULL

);

1. find agents who receive commissions between 0.12 and 0.14 (begin and end values are included). Return agent\_Code, name, city, and commission.
2. retrieve the details of the agent whose names begin with any letter between 'A' and 'L' (not inclusive). Return agent\_code, name, city, commission.
3. find all those customers who does not have any grade. Return customer\_id, cust\_name, city, grade, agent\_code.
4. find the highest purchase amount ordered by each customer. Return CUST\_CODE, maximum purchase amount.
5. calculate total purchase amount of all orders. Return total purchase amount.
6. determine the number of customers who received at least one grade for their activity.
7. find the highest purchase amount ordered by each customer on a particular date. Return, order date and highest purchase amount.
8. find the highest order (purchase) amount by each customer on a particular order date. Filter the result by highest order (purchase) amount above 2000.00. Return CUST\_CODE, order date and maximum purchase amount.
9. count all the orders generated on '2012-08-17'. Return number of orders.
10. find those agents who generated orders for their customers but are not located in the same city. Return ORD\_NUM, cust\_name, cust\_code (orders table), agent\_code (orders table).
11. find those customers who are served by a salesperson and the salesperson earns commission in the range of 12% to 14% (Begin and end values are included.). Return cust\_name AS "Customer", city AS "City".
12. find all orders executed by the salesperson and ordered by the customer whose grade is greater than or equal to 200. Compute purch\_amt\*commission as “Commission”. Return customer name, commission as “Commission%” and Commission.
13. find the order values greater than the average order value of 10th October 2012. Return ord\_no, purch\_amt, ord\_date, cust\_code, agent\_code.
14. find the sums of the amounts from the orders table, grouped by date, and eliminate all dates where the sum was not at least 1000.00 above the maximum order amount for that date.
15. find details of all orders excluding those with ord\_date equal to '2012-09-10' and agent\_code higher than 5005 or purch\_amt greater than 1000.Return ord\_no, purch\_amt, ord\_date, cust\_code and salesman\_id.

Note:

1. Any missing data needs to be added and then query to be executed
2. Attribute names may vary, consider appropriate name when executing the solutions

**Consider the 25 documents of the Restaurant Collection given below:**

{"address": {"building": "1007", "coord": [-73.856077, 40.848447], "street": "Morris Park Ave", "zipcode": "10462"}, "borough": "Bronx", "cuisine": "Bakery", "grades": [{"date": {"$date": 1393804800000}, "grade": "A", "score": 2}, {"date": {"$date": 1378857600000}, "grade": "A", "score": 6}, {"date": {"$date": 1358985600000}, "grade": "A", "score": 10}, {"date": {"$date": 1322006400000}, "grade": "A", "score": 9}, {"date": {"$date": 1299715200000}, "grade": "B", "score": 14}], "name": "Morris Park Bake Shop", "restaurant\_id": "30075445"}

{"address": {"building": "469", "coord": [-73.961704, 40.662942], "street": "Flatbush Avenue", "zipcode": "11225"}, "borough": "Brooklyn", "cuisine": "Hamburgers", "grades": [{"date": {"$date": 1419897600000}, "grade": "A", "score": 8}, {"date": {"$date": 1404172800000}, "grade": "B", "score": 23}, {"date": {"$date": 1367280000000}, "grade": "A", "score": 12}, {"date": {"$date": 1336435200000}, "grade": "A", "score": 12}], "name": "Wendy'S", "restaurant\_id": "30112340"}

{"address": {"building": "351", "coord": [-73.98513559999999, 40.7676919], "street": "West   57 Street", "zipcode": "10019"}, "borough": "Manhattan", "cuisine": "Irish", "grades": [{"date": {"$date": 1409961600000}, "grade": "A", "score": 2}, {"date": {"$date": 1374451200000}, "grade": "A", "score": 11}, {"date": {"$date": 1343692800000}, "grade": "A", "score": 12}, {"date": {"$date": 1325116800000}, "grade": "A", "score": 12}], "name": "Dj Reynolds Pub And Restaurant", "restaurant\_id": "30191841"}

{"address": {"building": "2780", "coord": [-73.98241999999999, 40.579505], "street": "Stillwell Avenue", "zipcode": "11224"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1402358400000}, "grade": "A", "score": 5}, {"date": {"$date": 1370390400000}, "grade": "A", "score": 7}, {"date": {"$date": 1334275200000}, "grade": "A", "score": 12}, {"date": {"$date": 1318377600000}, "grade": "A", "score": 12}], "name": "Riviera Caterer", "restaurant\_id": "40356018"}

{"address": {"building": "97-22", "coord": [-73.8601152, 40.7311739], "street": "63 Road", "zipcode": "11374"}, "borough": "Queens", "cuisine": "Jewish/Kosher", "grades": [{"date": {"$date": 1416787200000}, "grade": "Z", "score": 20}, {"date": {"$date": 1358380800000}, "grade": "A", "score": 13}, {"date": {"$date": 1343865600000}, "grade": "A", "score": 13}, {"date": {"$date": 1323907200000}, "grade": "B", "score": 25}], "name": "Tov Kosher Kitchen", "restaurant\_id": "40356068"}

{"address": {"building": "8825", "coord": [-73.8803827, 40.7643124], "street": "Astoria Boulevard", "zipcode": "11369"}, "borough": "Queens", "cuisine": "American ", "grades": [{"date": {"$date": 1416009600000}, "grade": "Z", "score": 38}, {"date": {"$date": 1398988800000}, "grade": "A", "score": 10}, {"date": {"$date": 1362182400000}, "grade": "A", "score": 7}, {"date": {"$date": 1328832000000}, "grade": "A", "score": 13}], "name": "Brunos On The Boulevard", "restaurant\_id": "40356151"}

{"address": {"building": "2206", "coord": [-74.1377286, 40.6119572], "street": "Victory Boulevard", "zipcode": "10314"}, "borough": "Staten Island", "cuisine": "Jewish/Kosher", "grades": [{"date": {"$date": 1412553600000}, "grade": "A", "score": 9}, {"date": {"$date": 1400544000000}, "grade": "A", "score": 12}, {"date": {"$date": 1365033600000}, "grade": "A", "score": 12}, {"date": {"$date": 1327363200000}, "grade": "A", "score": 9}], "name": "Kosher Island", "restaurant\_id": "40356442"}

{"address": {"building": "7114", "coord": [-73.9068506, 40.6199034], "street": "Avenue U", "zipcode": "11234"}, "borough": "Brooklyn", "cuisine": "Delicatessen", "grades": [{"date": {"$date": 1401321600000}, "grade": "A", "score": 10}, {"date": {"$date": 1389657600000}, "grade": "A", "score": 10}, {"date": {"$date": 1375488000000}, "grade": "A", "score": 8}, {"date": {"$date": 1342569600000}, "grade": "A", "score": 10}, {"date": {"$date": 1331251200000}, "grade": "A", "score": 13}, {"date": {"$date": 1318550400000}, "grade": "A", "score": 9}], "name": "Wilken'S Fine Food", "restaurant\_id": "40356483"}

{"address": {"building": "6409", "coord": [-74.00528899999999, 40.628886], "street": "11 Avenue", "zipcode": "11219"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1405641600000}, "grade": "A", "score": 12}, {"date": {"$date": 1375142400000}, "grade": "A", "score": 12}, {"date": {"$date": 1360713600000}, "grade": "A", "score": 11}, {"date": {"$date": 1345075200000}, "grade": "A", "score": 2}, {"date": {"$date": 1313539200000}, "grade": "A", "score": 11}], "name": "Regina Caterers", "restaurant\_id": "40356649"}

{"address": {"building": "1839", "coord": [-73.9482609, 40.6408271], "street": "Nostrand Avenue", "zipcode": "11226"}, "borough": "Brooklyn", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [{"date": {"$date": 1405296000000}, "grade": "A", "score": 12}, {"date": {"$date": 1373414400000}, "grade": "A", "score": 8}, {"date": {"$date": 1341964800000}, "grade": "A", "score": 5}, {"date": {"$date": 1329955200000}, "grade": "A", "score": 8}], "name": "Taste The Tropics Ice Cream", "restaurant\_id": "40356731"}

{"address": {"building": "2300", "coord": [-73.8786113, 40.8502883], "street": "Southern Boulevard", "zipcode": "10460"}, "borough": "Bronx", "cuisine": "American ", "grades": [{"date": {"$date": 1401235200000}, "grade": "A", "score": 11}, {"date": {"$date": 1371600000000}, "grade": "A", "score": 4}, {"date": {"$date": 1339718400000}, "grade": "A", "score": 3}], "name": "Wild Asia", "restaurant\_id": "40357217"}

{"address": {"building": "7715", "coord": [-73.9973325, 40.61174889999999], "street": "18 Avenue", "zipcode": "11214"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1397606400000}, "grade": "A", "score": 5}, {"date": {"$date": 1366675200000}, "grade": "A", "score": 2}, {"date": {"$date": 1335225600000}, "grade": "A", "score": 5}, {"date": {"$date": 1323993600000}, "grade": "A", "score": 2}], "name": "C & C Catering Service", "restaurant\_id": "40357437"}

{"address": {"building": "1269", "coord": [-73.871194, 40.6730975], "street": "Sutter Avenue", "zipcode": "11208"}, "borough": "Brooklyn", "cuisine": "Chinese", "grades": [{"date": {"$date": 1410825600000}, "grade": "B", "score": 21}, {"date": {"$date": 1377648000000}, "grade": "A", "score": 7}, {"date": {"$date": 1364860800000}, "grade": "C", "score": 56}, {"date": {"$date": 1344988800000}, "grade": "B", "score": 27}, {"date": {"$date": 1332892800000}, "grade": "B", "score": 27}], "name": "May May Kitchen", "restaurant\_id": "40358429"}

{"address": {"building": "1", "coord": [-73.96926909999999, 40.7685235], "street": "East   66 Street", "zipcode": "10065"}, "borough": "Manhattan", "cuisine": "American ", "grades": [{"date": {"$date": 1399420800000}, "grade": "A", "score": 3}, {"date": {"$date": 1367539200000}, "grade": "A", "score": 4}, {"date": {"$date": 1335744000000}, "grade": "A", "score": 6}, {"date": {"$date": 1324944000000}, "grade": "A", "score": 0}], "name": "1 East 66Th Street Kitchen", "restaurant\_id": "40359480"}

{"address": {"building": "705", "coord": [-73.9653967, 40.6064339], "street": "Kings Highway", "zipcode": "11223"}, "borough": "Brooklyn", "cuisine": "Jewish/Kosher", "grades": [{"date": {"$date": 1415577600000}, "grade": "A", "score": 11}, {"date": {"$date": 1381363200000}, "grade": "A", "score": 13}, {"date": {"$date": 1349308800000}, "grade": "A", "score": 7}, {"date": {"$date": 1337558400000}, "grade": "A", "score": 9}, {"date": {"$date": 1325203200000}, "grade": "B", "score": 19}], "name": "Seuda Foods", "restaurant\_id": "40360045"}

{"address": {"building": "203", "coord": [-73.97822040000001, 40.6435254], "street": "Church Avenue", "zipcode": "11218"}, "borough": "Brooklyn", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [{"date": {"$date": 1391990400000}, "grade": "A", "score": 2}, {"date": {"$date": 1357084800000}, "grade": "A", "score": 13}, {"date": {"$date": 1326067200000}, "grade": "A", "score": 3}, {"date": {"$date": 1320624000000}, "grade": "P", "score": 12}, {"date": {"$date": 1311206400000}, "grade": "A", "score": 13}], "name": "Carvel Ice Cream", "restaurant\_id": "40360076"}

{"address": {"building": "265-15", "coord": [-73.7032601, 40.7386417], "street": "Hillside Avenue", "zipcode": "11004"}, "borough": "Queens", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [{"date": {"$date": 1414454400000}, "grade": "A", "score": 9}, {"date": {"$date": 1379462400000}, "grade": "A", "score": 10}, {"date": {"$date": 1348099200000}, "grade": "A", "score": 13}], "name": "Carvel Ice Cream", "restaurant\_id": "40361322"}

{"address": {"building": "6909", "coord": [-74.0259567, 40.6353674], "street": "3 Avenue", "zipcode": "11209"}, "borough": "Brooklyn", "cuisine": "Delicatessen", "grades": [{"date": {"$date": 1408579200000}, "grade": "A", "score": 4}, {"date": {"$date": 1393977600000}, "grade": "A", "score": 3}, {"date": {"$date": 1357776000000}, "grade": "A", "score": 10}], "name": "Nordic Delicacies", "restaurant\_id": "40361390"}

{"address": {"building": "522", "coord": [-73.95171, 40.767461], "street": "East   74 Street", "zipcode": "10021"}, "borough": "Manhattan", "cuisine": "American ", "grades": [{"date": {"$date": 1409616000000}, "grade": "A", "score": 12}, {"date": {"$date": 1387411200000}, "grade": "B", "score": 16}, {"date": {"$date": 1369699200000}, "grade": "A", "score": 9}, {"date": {"$date": 1354838400000}, "grade": "A", "score": 13}, {"date": {"$date": 1332979200000}, "grade": "A", "score": 11}], "name": "Glorious Food", "restaurant\_id": "40361521"}

{"address": {"building": "284", "coord": [-73.9829239, 40.6580753], "street": "Prospect Park West", "zipcode": "11215"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1416355200000}, "grade": "A", "score": 11}, {"date": {"$date": 1384387200000}, "grade": "A", "score": 2}, {"date": {"$date": 1354665600000}, "grade": "A", "score": 13}, {"date": {"$date": 1337212800000}, "grade": "A", "score": 11}], "name": "The Movable Feast", "restaurant\_id": "40361606"}

{"address": {"building": "129-08", "coord": [-73.839297, 40.78147], "street": "20 Avenue", "zipcode": "11356"}, "borough": "Queens", "cuisine": "Delicatessen", "grades": [{"date": {"$date": 1408147200000}, "grade": "A", "score": 12}, {"date": {"$date": 1377561600000}, "grade": "A", "score": 9}, {"date": {"$date": 1348099200000}, "grade": "A", "score": 7}, {"date": {"$date": 1317254400000}, "grade": "A", "score": 10}], "name": "Sal'S Deli", "restaurant\_id": "40361618"}

{"address": {"building": "759", "coord": [-73.9925306, 40.7309346], "street": "Broadway", "zipcode": "10003"}, "borough": "Manhattan", "cuisine": "Delicatessen", "grades": [{"date": {"$date": 1390262400000}, "grade": "A", "score": 12}, {"date": {"$date": 1357257600000}, "grade": "A", "score": 11}, {"date": {"$date": 1339027200000}, "grade": "A", "score": 6}, {"date": {"$date": 1326758400000}, "grade": "A", "score": 8}], "name": "Bully'S Deli", "restaurant\_id": "40361708"}

{"address": {"building": "3406", "coord": [-73.94024739999999, 40.7623288], "street": "10 Street", "zipcode": "11106"}, "borough": "Queens", "cuisine": "Delicatessen", "grades": [{"date": {"$date": 1395187200000}, "grade": "A", "score": 3}, {"date": {"$date": 1363132800000}, "grade": "A", "score": 12}, {"date": {"$date": 1332806400000}, "grade": "A", "score": 8}, {"date": {"$date": 1301961600000}, "grade": "A", "score": 7}], "name": "Steve Chu'S Deli & Grocery", "restaurant\_id": "40361998"}

{"address": {"building": "502", "coord": [-73.976112, 40.786714], "street": "Amsterdam Avenue", "zipcode": "10024"}, "borough": "Manhattan", "cuisine": "Chicken", "grades": [{"date": {"$date": 1410739200000}, "grade": "A", "score": 10}, {"date": {"$date": 1393891200000}, "grade": "A", "score": 13}, {"date": {"$date": 1374105600000}, "grade": "A", "score": 13}, {"date": {"$date": 1357689600000}, "grade": "A", "score": 11}, {"date": {"$date": 1334016000000}, "grade": "A", "score": 10}, {"date": {"$date": 1321315200000}, "grade": "A", "score": 7}], "name": "Harriet'S Kitchen", "restaurant\_id": "40362098"}

{"address": {"building": "730", "coord": [-73.96805719999999, 40.7925587], "street": "Columbus Avenue", "zipcode": "10025"}, "borough": "Manhattan", "cuisine": "American ", "grades": [{"date": {"$date": 1410480000000}, "grade": "B", "score": 26}, {"date": {"$date": 1377648000000}, "grade": "A", "score": 9}, {"date": {"$date": 1364169600000}, "grade": "B", "score": 20}, {"date": {"$date": 1329177600000}, "grade": "A", "score": 12}], "name": "P & S Deli Grocery", "restaurant\_id": "40362264"}

**Design the Mongodb solutions for the below given queries**

1. select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.
2. find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name
3. find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan.
4. find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.
5. find the restaurants that have a grade with a score of 2 or a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.
6. find the average score for each restaurant.
7. find the count of restaurants for each cuisine.
8. find the count of restaurants that received a grade of 'A' for each cuisine.
9. find the number of restaurants that have been graded in each month of the year.
10. find the name and address of the restaurants that received a grade of 'A' on a specific date.