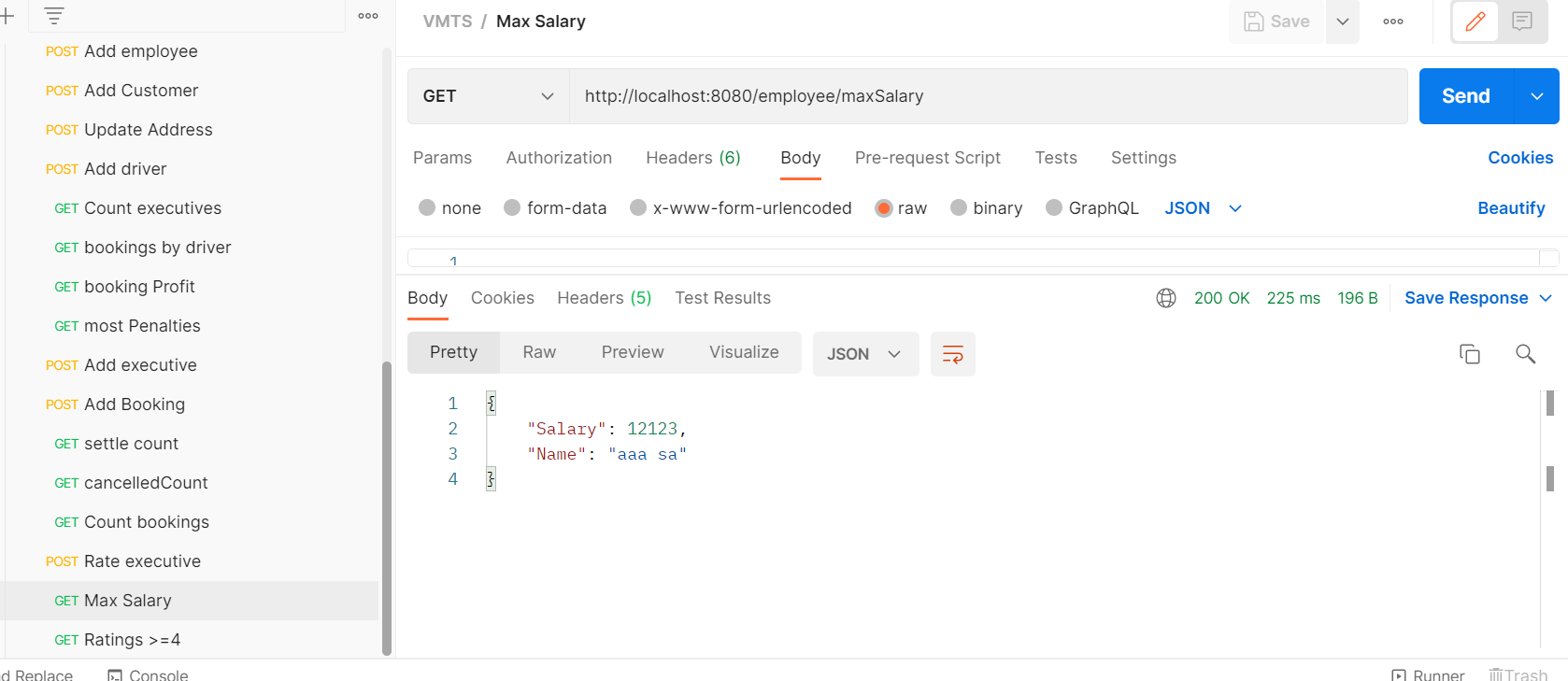
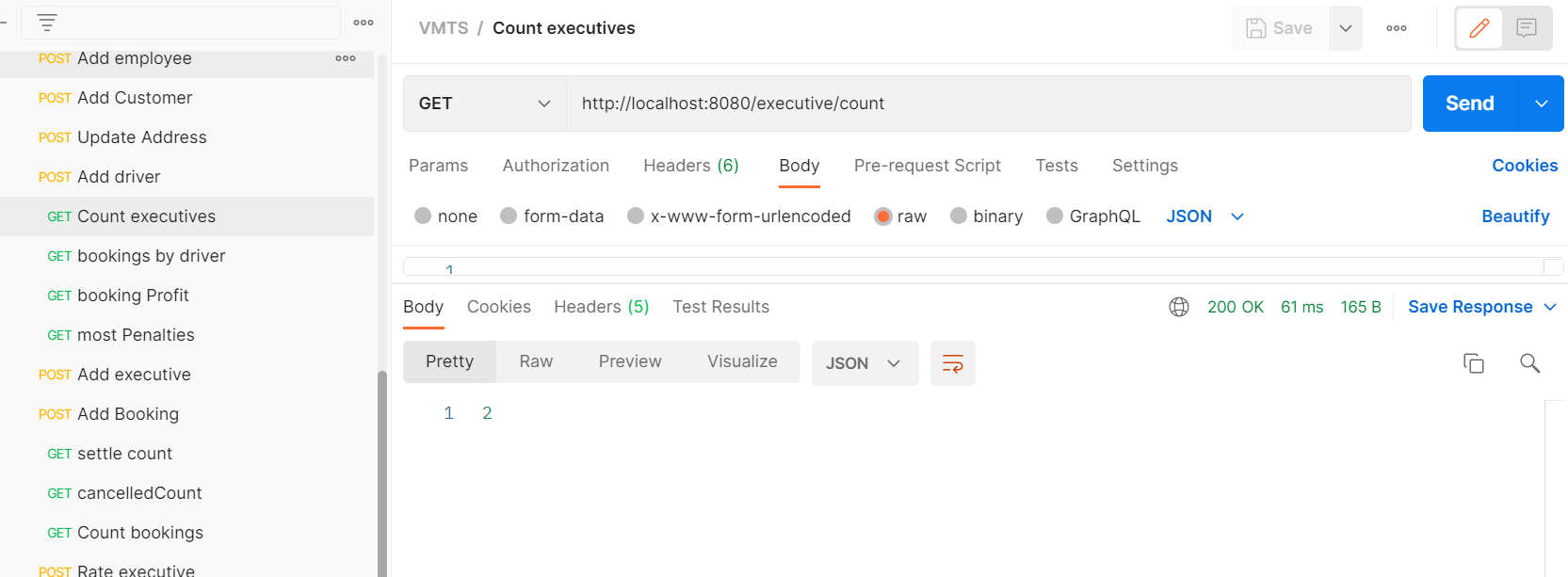
1. Name of emp with the highest salary

Select \* from EMPLOYEE where SALARY =( select MAX(SALARY) from EMPLOYEE );



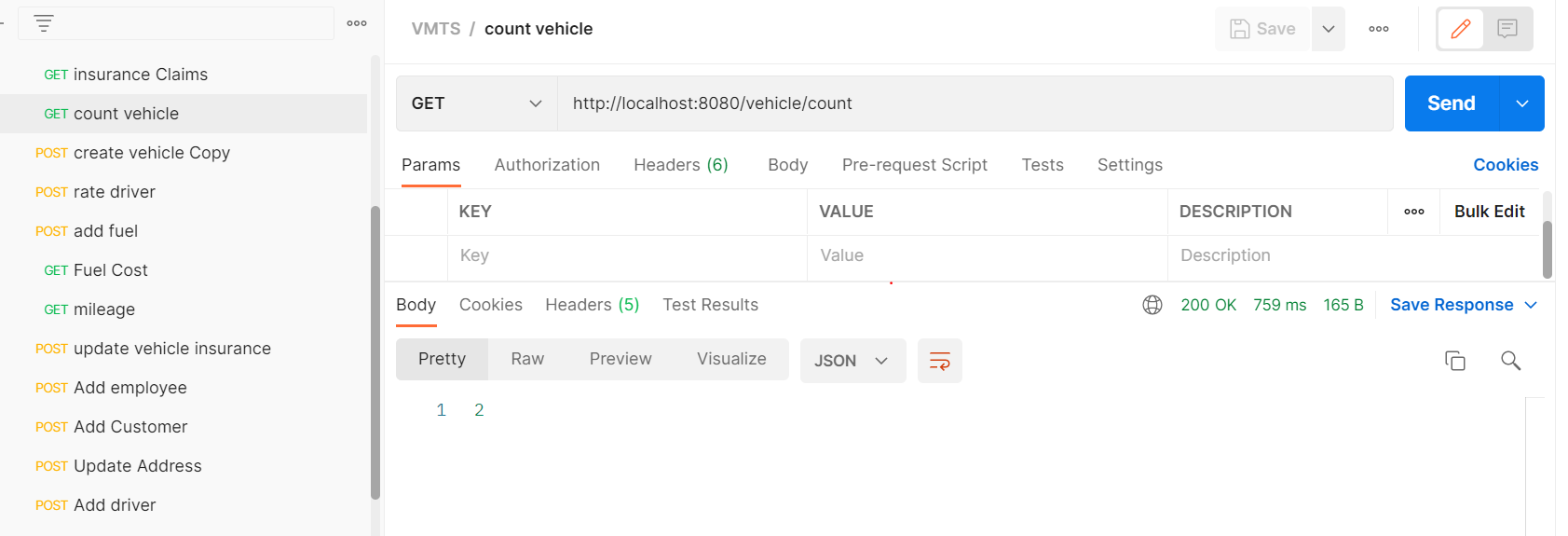
2. Number of executives in the company

SELECT COUNT(\*) FROM VMTS.EXECUTIVE;



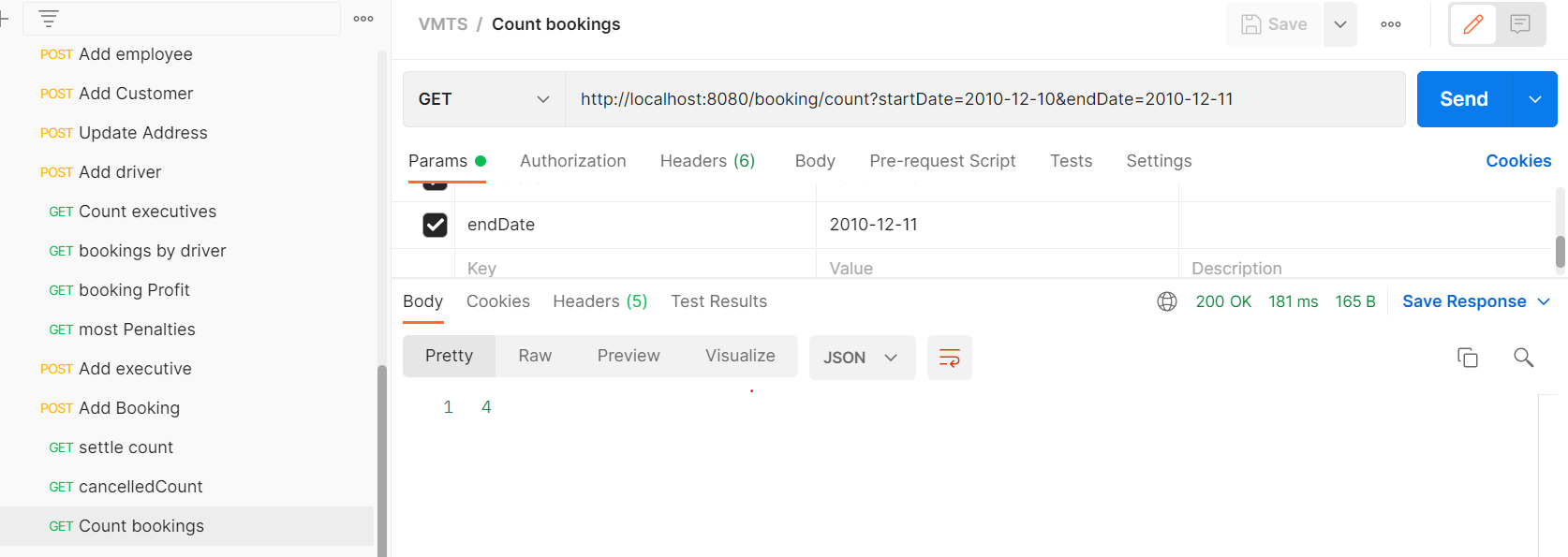
3.Number of vehicles in the company

SELECT COUNT(\*) FROM VMTS.VEHICLE;



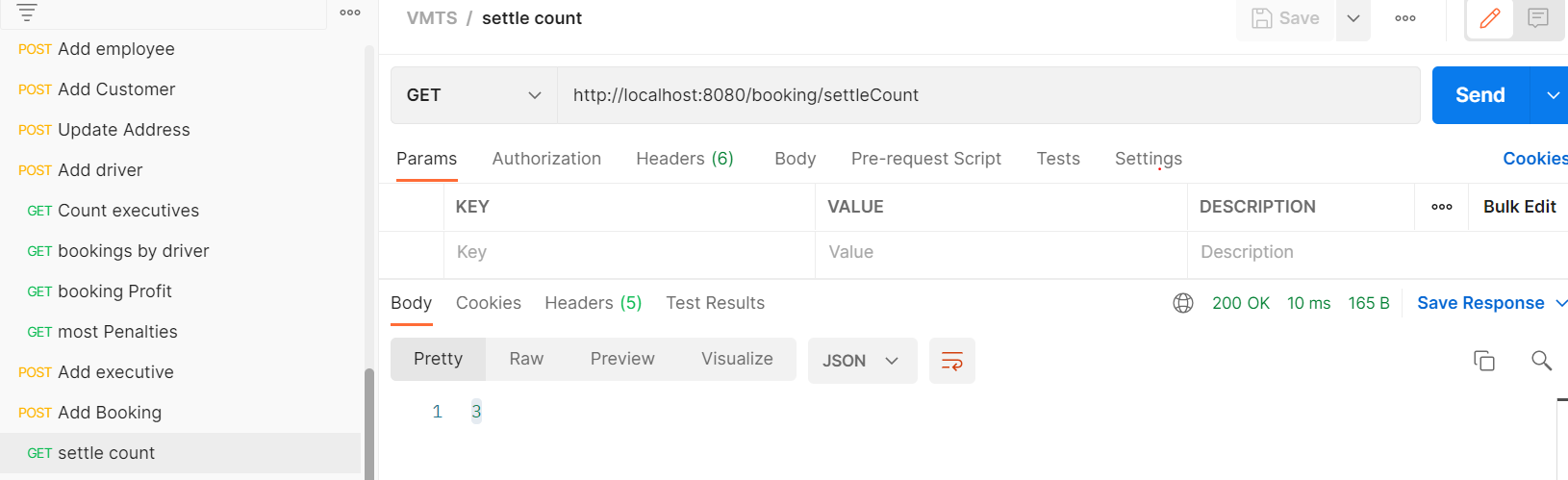
4. Number of total bookings made between specific dates

SELECT COUNT(\*) FROM VMTS.BOOKING WHERE START\_DATE BETWEEN <from\_date> AND <to\_date>;



5. Number of bookings that need to be settled

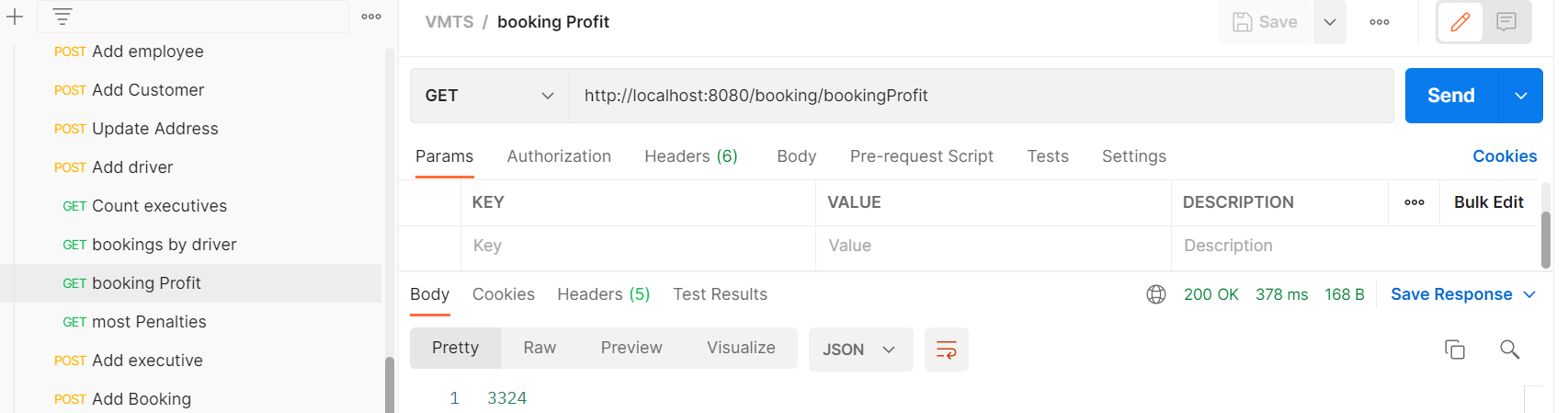
SELECT COUNT(\*) FROM VMTS.BOOKING WHERE BOOKING\_STATUS = 'completed';



6. Profit from bookings

select (sum(V.RENT\_PER\_DAY \* ((B.END\_DATE - B.START\_DATE) + 1)) - sum(B.penalty\_charge) - sum(F.fuel\_filled\_amount)) AS 'Total PROFIT'

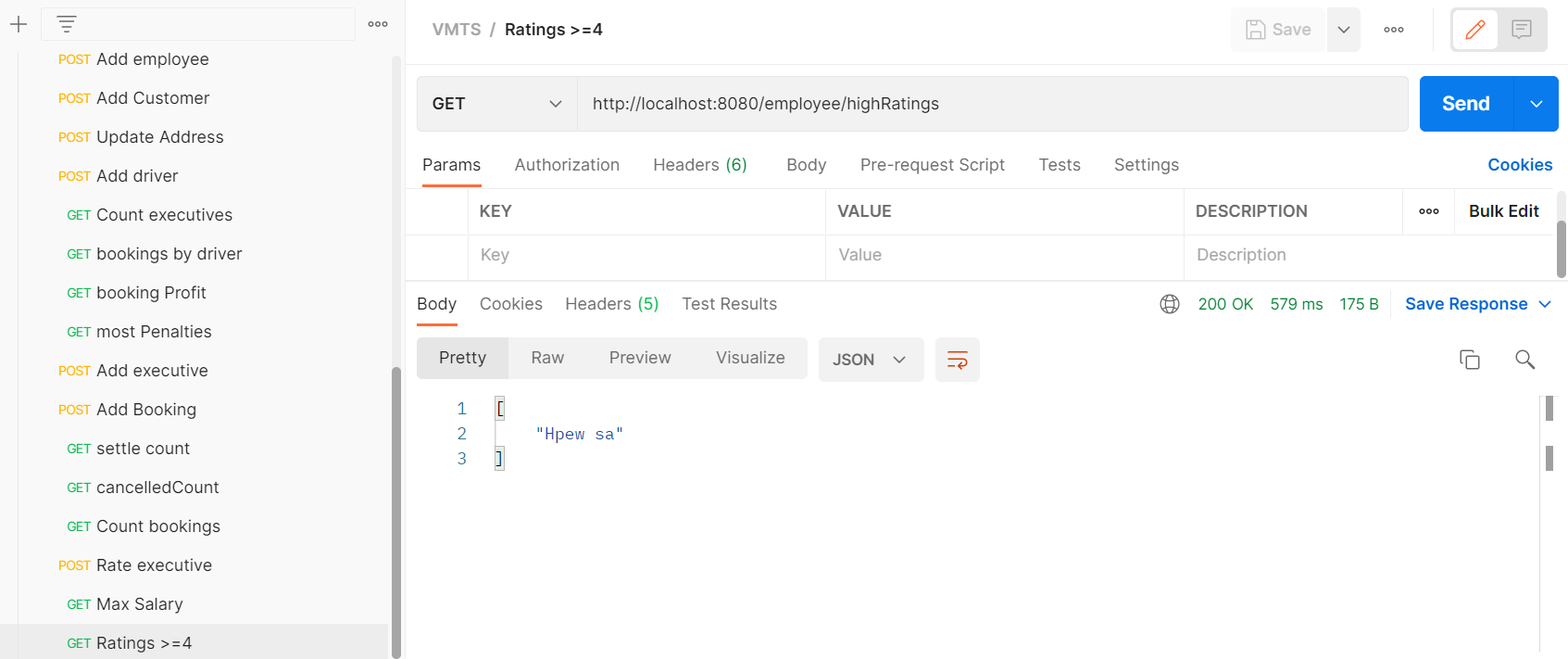
from VMTS.VEHICLE V, VMTS.Booking B, VMTS.fuel\_dtls F \where V.reg\_no = B.vehicle\_id AND B.booking\_status = 'PAID';



7. Get driver name/executive name with rating '4'

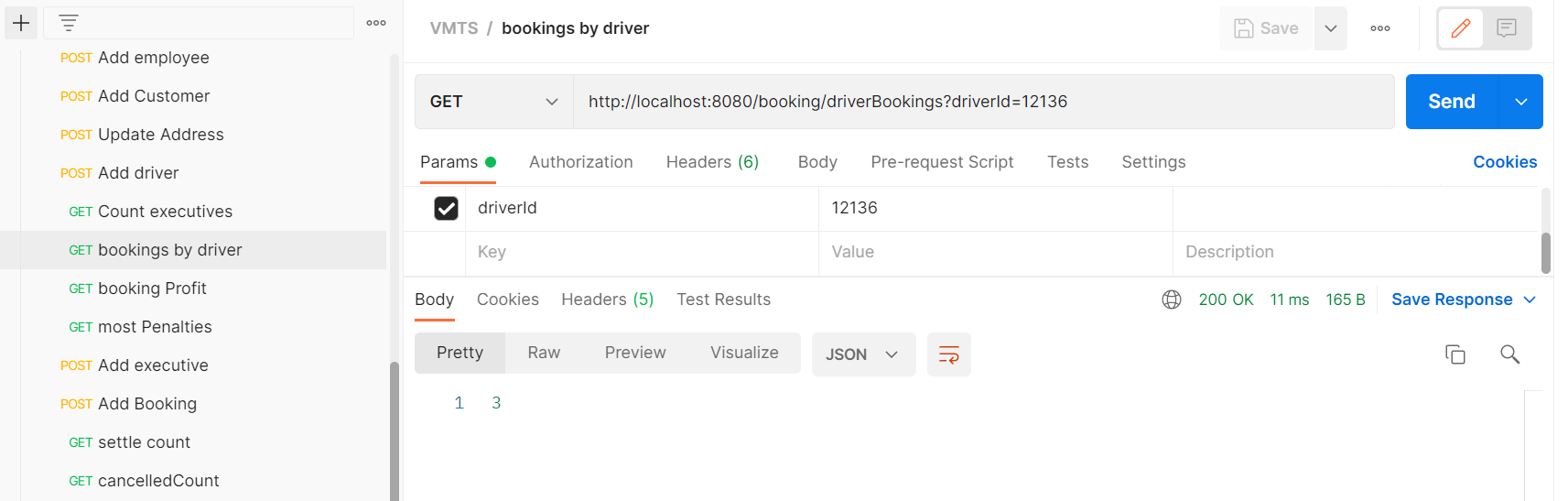
SELECT distinct CONCAT(A.FNAME,' ', A.LNAME) AS Highest\_rated\_employee FROM VMTS.EMPLOYEE A, VMTS.DRIVER B, VMTS.EXECUTIVE E

WHERE (A.EMP\_ID = E.EMP\_ID AND E.RATING >= '4') OR (A.EMP\_ID = B.EMP\_ID AND B.RATING >= '4');



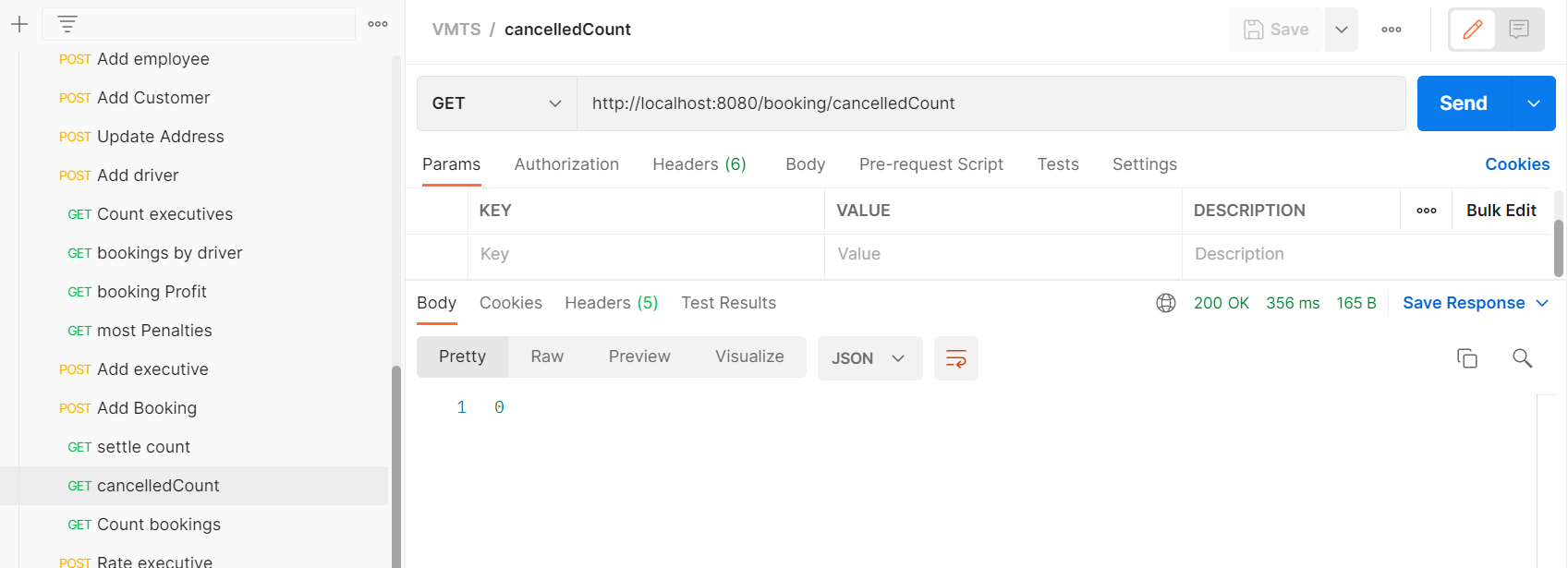
8. No.of bookings made by specific driver

SELECT COUNT(BOOKING\_ID) FROM VMTS.BOOKING WHERE DRIVER\_ID = '<ID >'



9. Number of cancelled bookings

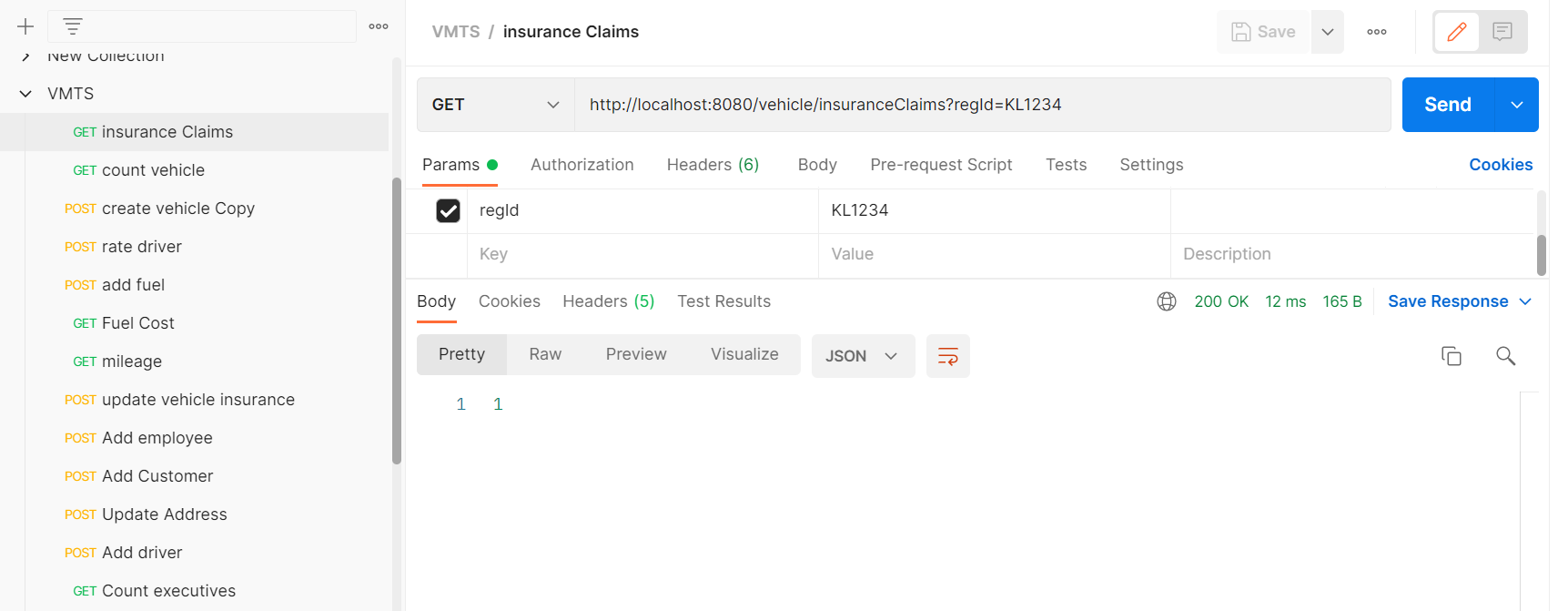
SELECT COUNT(\*) FROM VMTS.BOOKING WHERE BOOKING\_STATUS = 'cancelled';



10. No.of insurance claims made by vehicle with Reg.No

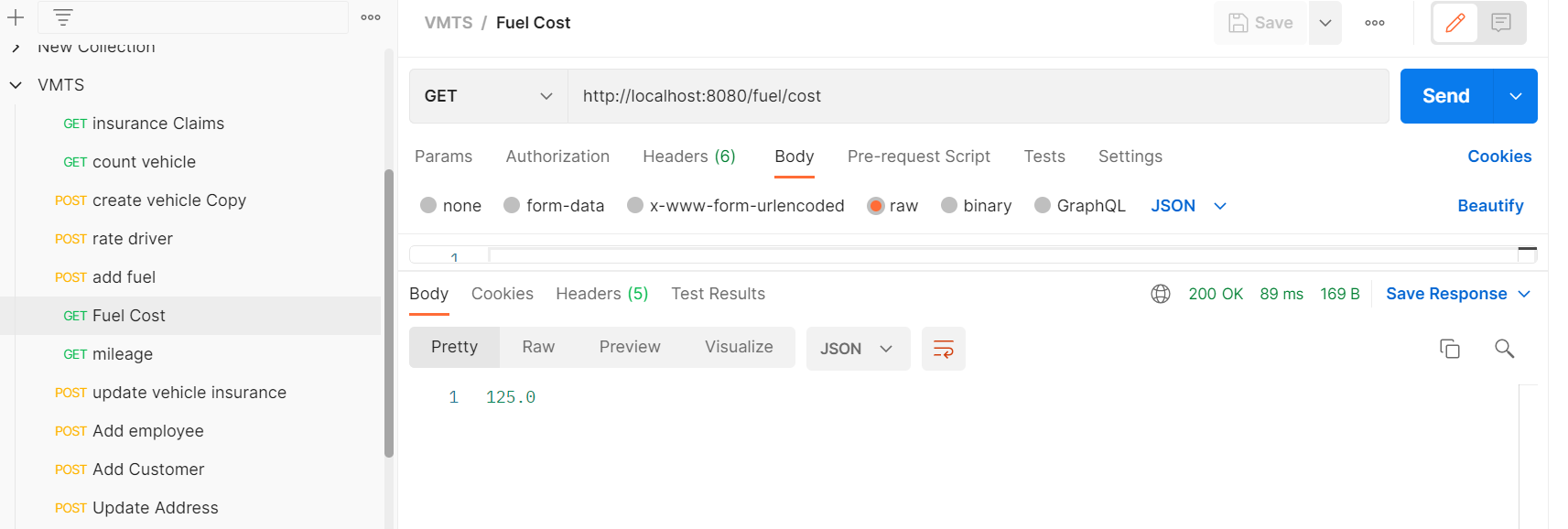
SELECT A.NO\_OF\_CLAIMS FROM VMTS.INSURANCE\_DTLS A,VMTS.VEHICLE B WHERE B.INSURANCE\_NO = A.INSURANCE\_ID AND

B.REG\_NO = <regNo>;



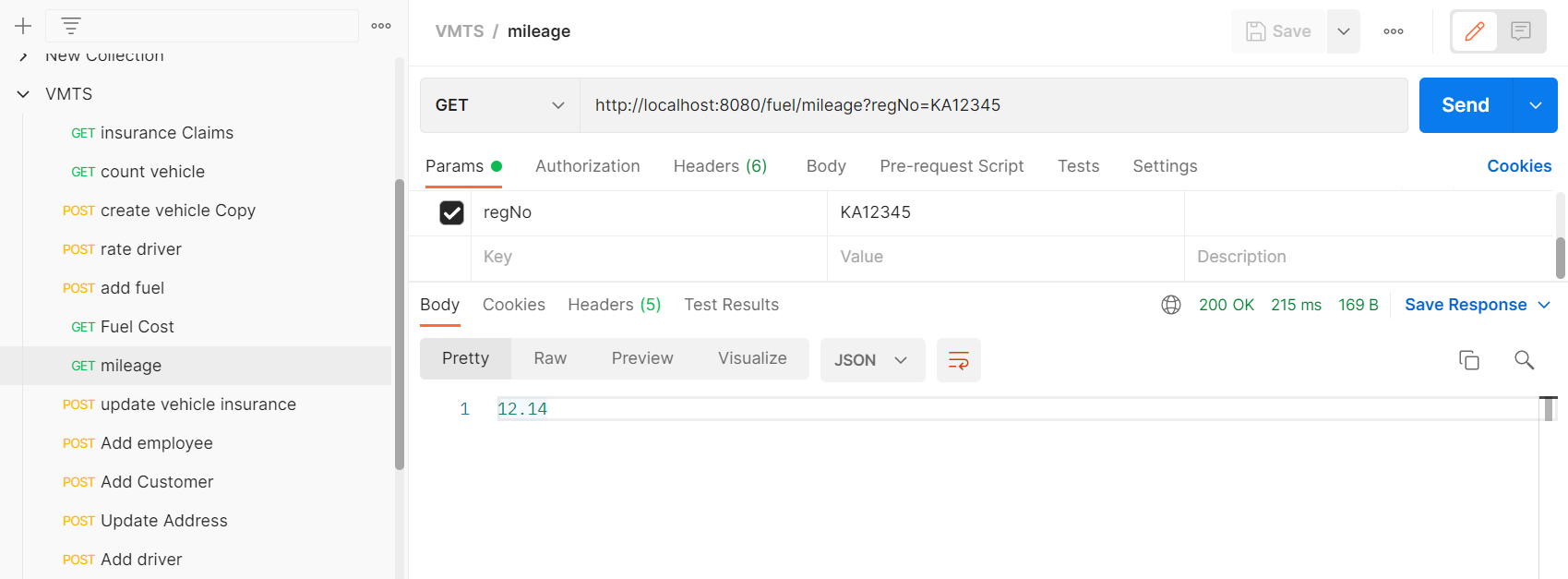
11. Total fuel cost in last month

SELECT SUM(FUEL\_FILLED\_AMOUNT) FROM VMTS.FUEL\_DTLS WHERE DATE\_OF\_FILLING < current\_date() AND DATE\_OF\_FILLING > current\_date()-30;



12. Mileage for a vehicle with reg no (new column added fuel\_filled)

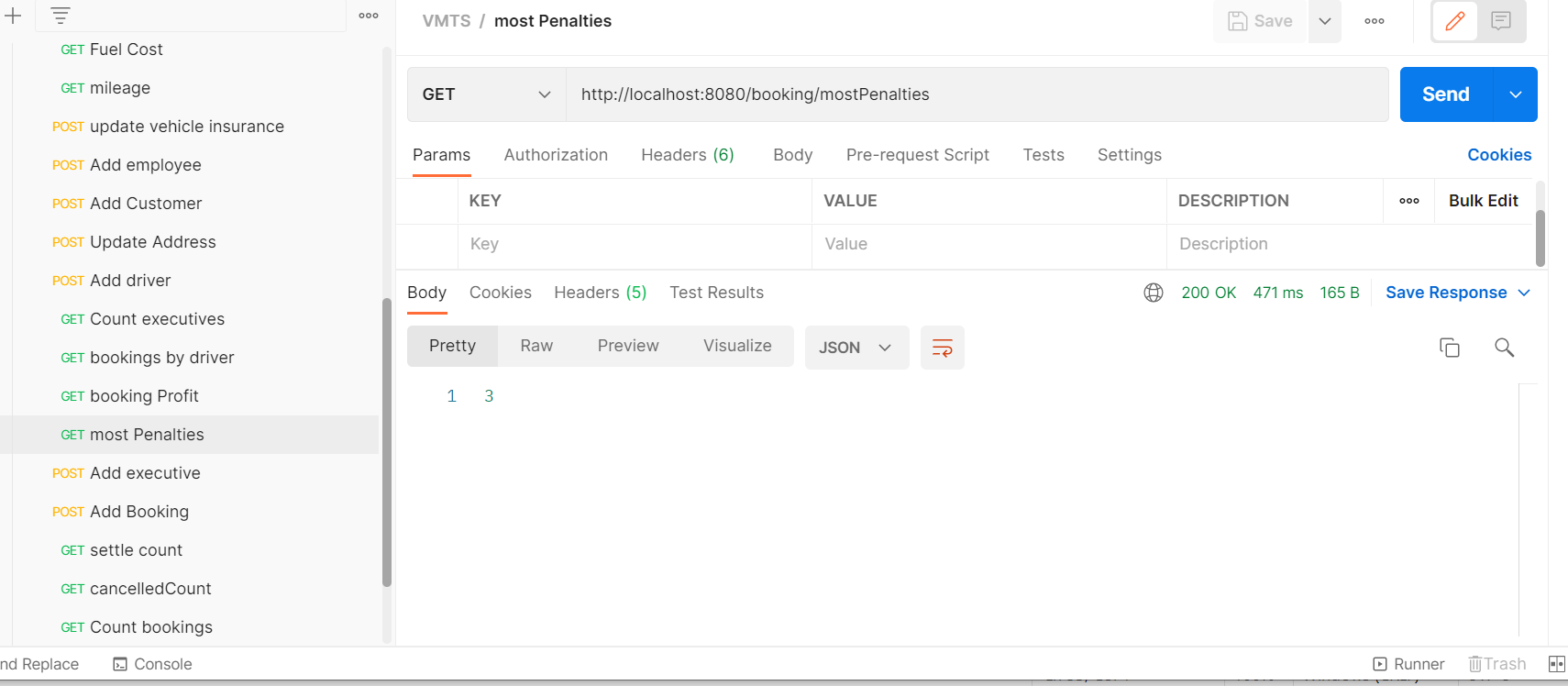
select round((SUM(b.km\_driven)/SUM(f.fuel\_filled)),2) AS mileage from vmts.fuel\_dtls f, vmts.booking b where f.vehicle\_id=b.vehicle\_id AND f.vehicle\_id = <vehicleId>;



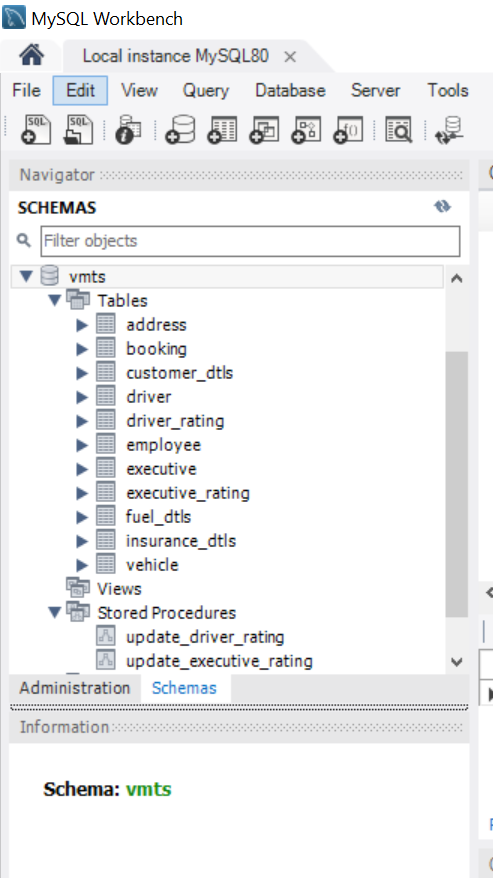
13. Driver involved with more number of penalties

select MAX(count) from (select driver\_id, count(penalty\_charge) count from vmts.booking

where penalty\_charge <> 0 group by driver\_id) AS Penalty\_Count;



Tables:



**Stored Procedures**

DELIMITER //

CREATE PROCEDURE vmts.update\_driver\_rating ()

BEGIN

update vmts.driver d set d.rating = round((select avg(r.rating)

from vmts.driver\_rating r where r.driver\_id=d.emp\_id group by r.driver\_id ),2) ;

END

DELIMITER //

CREATE PROCEDURE vmts.update\_executive\_rating ()

BEGIN

update vmts.executive e set e.rating = round((select avg(r.rating)

from vmts.executive\_rating r where r.executive\_id=e.emp\_id group by r.executive\_id ),2) ;

END

**Triggers**

DELIMITER //

CREATE TRIGGER vmts.update\_driver

AFTER INSERT

ON vmts.driver\_rating FOR EACH ROW

BEGIN

call vmts.update\_driver\_rating();

END

DELIMITER //

CREATE TRIGGER vmts.update\_executive

AFTER INSERT

ON vmts.executive\_rating FOR EACH ROW

BEGIN

call vmts.update\_executive\_rating();

END