```sql
-- Chris Crown
--MSBA 630, Modern Data Management
-- Dr. Zara Hatami
-- Assignment 4 - Due Dec 6


-- Part 1

-- Q1

--drop table emp_1

--select *
--from emp_1

create table emp_1
(
emp_num char (3) not null,
emp_lname varchar (15) not null,
emp_fname varchar (15) not null,
emp_initial char (1),
emp_hiredate date,
job_code char (3)
)

alter table emp_1
add constraint pk_emp_1 primary key (emp_num),
constraint fk_emp_1_job foreign key (job_code) references job
-- this code is struggling because job_code in our existing table JOB is not char (3), but I believe the code is otherwise correct


-- Q2

insert into emp_1 values ('101', 'News', 'John', 'G', '08-Nov-00', '502') -- because these are char we have to put quotes around even tho...
insert into emp_1 values ('102', 'Senior', 'David', 'H', '12-Jul-89', '501') -- ...some are just integers

--select *
--from emp_1

--Q5


-- I do not have this data (was not instructed to enter) in my table, but here is the code to do this
update emp_1
set job_code ='501' -- these have to be treated as strings bc they are char in Q1
where emp_num = '107'

--Q6

--to do this we have to use the delete function

delete from emp_1
where emp_lname = 'Smithfield' and emp_fname = 'William'
and emp_hiredate = '22-Jun-04' and job_code = '500'
--stringing together these 'ands' makes sure we target the right row fpr deletion


--Q23

-- use the alter function then type in the new column ecaxtly as it is in te prompt then specify the data type
alter table customer
add CUST_DOB date

--Q24

update customer
set cust_dob = '15-Mar-89'
where cust_num = '1000' -- I am assuming cust_num is a char or varchar. I do not have this information. If it was int, I would not put ' '
around it


--Q25

--same question and assumptions here (on the data types)
update customer
set cust_dob = '22-Dec-88'
where cust_num = '1001'


--Part 2

--Q35
--first I need to add the new column
alter table model
add mod_wait_chg numeric (7,2) -- I am choosing to go up to the ten thousands with 2 decimals
```

```sql
--Q36
--then since we have not yet (11/27) learned how to update multiple values at once, I add them one at a time
update model
set mod_wait_chg = 100 -- i know this is numeric so i don't need quotes
where mod_code = 'C-90A' -- this has to be char or varchar so I have to use quotes

update model
set mod_wait_chg =50
where mod_code = 'PA23-250'

update model
set mod_wait_chg = 75
where mod_code = 'PA31-350'

--Q37
-----------------------------------------------------------------------------------------------------------------------------------------------------------

alter table charter
add char_wait_chg numeric (7,2), -- using 10 and 2 for these just to capture everything
add char_flt_chg_hr numeric (7,2),
add char_flt_chg numeric (7,2),
add char_tax_chg numeric (7,2),
add char_tot_chg numeric (7,2), -- not adding 'not null' for any of these bc I am assuming the pk already existed in the table b4 I added these
add char_pymt numeric (7,2),
add char_balance numeric (7,2)

alter table charter
add constraint fk_charter_model foreign key (char_wait_chag) references model(mod_wait_chg), -- using fk's to enforce referential integrity
between the copied...
constraint fk_charter_model_2 foreign key (char_flt_chg_hr) references model(mod_chg_mile) --...values from the model table (in the prompt)

update charter
set char_flt_chg = (char_hours_flown * char_flt_chg_hr)  -- updating the columns to include the equation given in the prompt

update charter
set char_tax_chg = (char_flt_chg * 0.08)

update charter
set char_tot_chg = (char_flt_chg + char_tax_chg)

update charter
set char_balance = (char_tot_chg - char_pymt) -- taking the total charge minus what was paid to create a remainder value for their balance


--Q38

-- I built my new attributes AS IF they were a copied foreign key FROM the model table, just with a different name
-- here is the code, however, to insert the new values

--it is very hard to do this without the actual tables but I think this should bridge the 3 tables. Bc aircraift has ac_number and mod_code,
--model has mod_code and mod_wait_chg and charter has ac_number
update charter
set char_wait_chg =
(select model.mod_wait_chg
from aircraft join model on aircraft.mod_code = model.mod_code
where aircraft.ac_number = charter.ac_number)

--Q39
-- I ignored the hints on these becuase I am not strongwith views or procedures yet. I will need more practice on these once the assignments
--are done and I can solely study for the final.
update charter
set char_flt_chg_hr =
(select model.mod_chg_mile
from aircraft join model on aircraft.mod_code = model.mod_code
where aircraft.ac_number = charter.ac_number)

--Q40

--I have already done this in my answer to Q37, here is the code again
update charter
set char_flt_chg = (char_hours_flown * char_flt_chg_hr)  -- updating the columns to include the equation given in the prompt

--Q41

-- same, above, here it is again
update charter
set char_tax_chg = (char_flt_chg * 0.08)

--Q42
-- same, above, here it is again
update charter
set char_tot_chg = (char_flt_chg + char_tax_chg)

--Q43

alter table pilot
```

```sql
add pil_pic_hrs numeric (7,2)

update pilot
set pil_pic_hrs = (select sum(charter.char_hours_flown)
from charter join crew on charter.char_trip = crew.char_trip
where crew.emp_num = pilot.emp_num and crew.crew_job = 'Pilot' )
-- I'm a join guy - hope this works

--Part 3

--Write a procedure to delete an invoice, giving the invice number as a parameter. Name the procedure prc_inv_delete

create table invoice_test
(
invoice_num int not null, -- first create a test table so I can mess around with it
amount int
)

insert into invoice_test values (8004, 123)
insert into invoice_test values (8005, 456) -- then put in some random values so we can test our procedure
insert into invoice_test values (8006, 789)

--select * from invoice_test -- make sure it all shows up

create procedure prc_inv_delete @invoice_num int -- shamelessly create a procedure using nearly the ecaxt same code as we were given in class for...
as                                                                                --...selecting a vendor
delete from invoice_test where invoice_num=@invoice_num


--test the proceudre by deleting invoices 8005 and 8006

execute prc_inv_delete @invoice_num=8005
execute prc_inv_delete @invoice_num = 8006

--select * from invoice_test -- make sure that 8004 still exists as the only invoice left
```