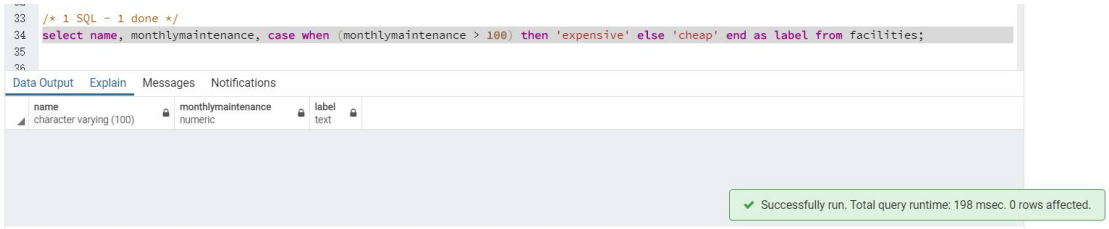
Question 1:

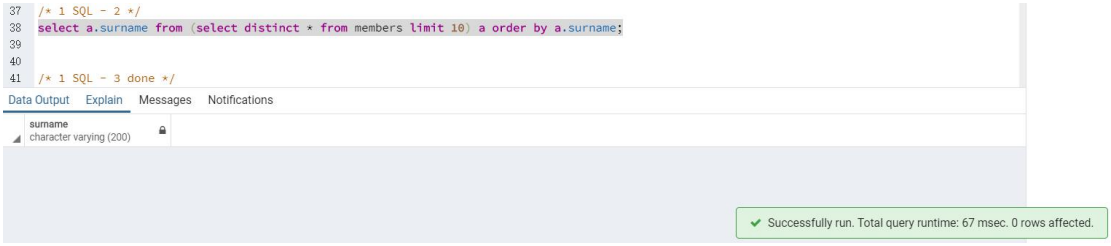
1-1:

select name, monthlymaintenance, case when (monthlymaintenance > 100) then 'expensive' else 'cheap' end as label from facilities;



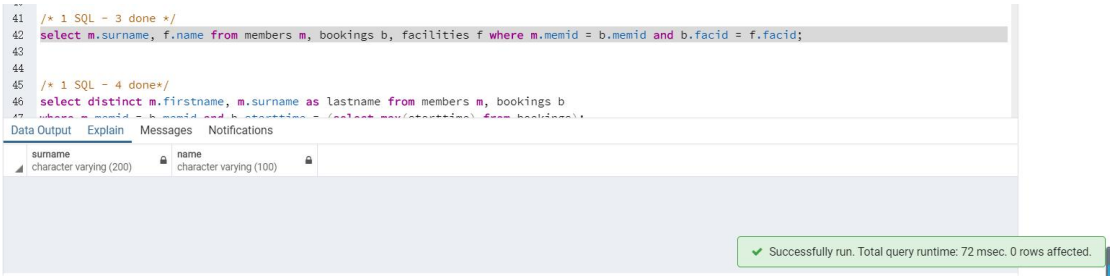
1-2:

select a.surname from (select distinct * from members limit 10) a order by a.surname;



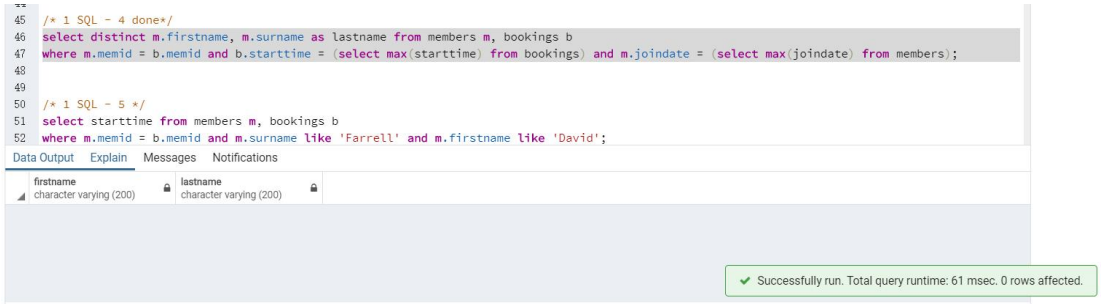
1-3:

select m.surname, f.name from members m, bookings b, facilities f where m.memid = b.memid and b.facid = f.facid;



1-4:

select distinct m.firstname, m.surname as lastname from members m, bookings b
where m.memid = b.memid and b.starttime = (select max(starttime) from bookings) and m.joindate = (select max(joindate) from members);



1-5:

select starttime from members m, bookings b
where m.memid = b.memid and m.surname like 'Farrell' and m.firstname like 'David';

```
/* 1 SQL - 5 */
select starttime from members m, bookings b
where m.memid = b.memid and m.surname like 'Farrell' and m.firstname like 'David';
```

Data Output | Explain | Messages | Notifications

starttime
timestamp without time zone

Successfully run. Total query runtime: 73 msec. 0 rows affected.

Question 2:

```
get.data <- function(ticker.name, time.interval){
   library(quantmod)
   stock.data <- getSymbols(as.character(ticker.name), from = "2017-01-01", to = "2017-12-31",
auto.assign = F)
   adj.close <- as.numeric(stock.data[, 6])
   if (length(adj.close) %% time.interval != 0){
      adj.close <- c(adj.close, rep(NA, time.interval - length(adj.close) %% time.interval))
   }
   origData <- as.data.frame(matrix(adj.close, ncol = time.interval,
                                                    nrow = ceiling(length(adj.close)/time.interval), byrow =
T))
   return(origData)
}

a <- get.data("AAPL", 9)
```

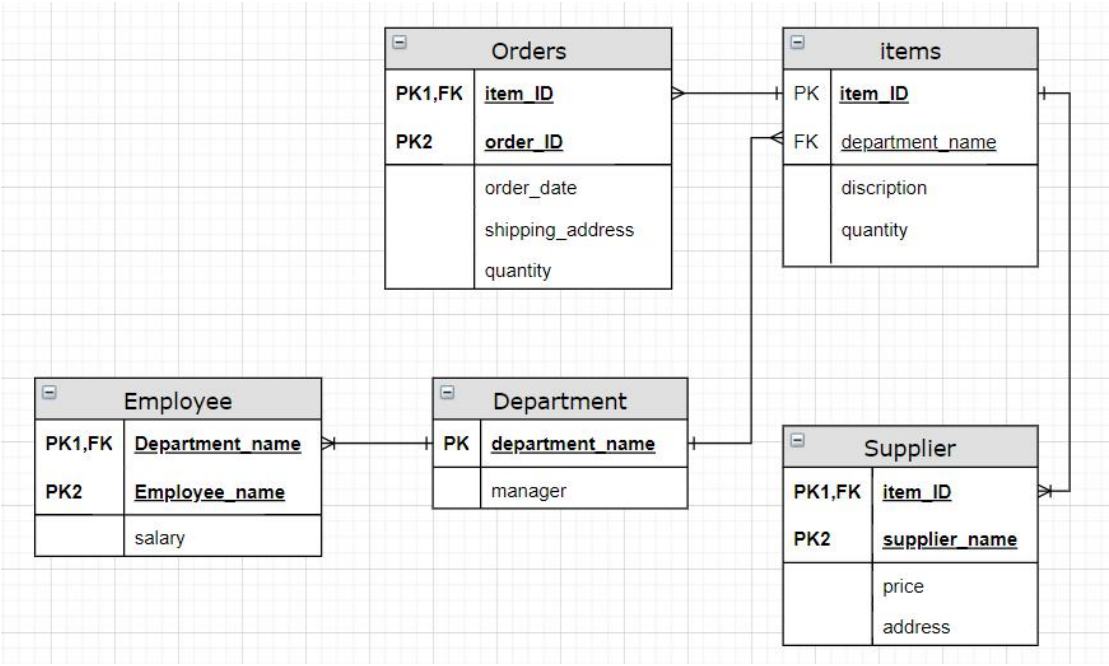
```
get.data <- function(ticker.name, time.interval){
  library(quantmod)
  stock.data <- getSymbols(as.character(ticker.name), from = "2017-01-01", to = "2017-12-31", auto.assign = F)
  adj.close <- as.numeric(stock.data[, 6])
  if (length(adj.close) %% time.interval != 0){
    adj.close <- c(adj.close, rep(NA, time.interval - length(adj.close) %% time.interval))
  }
  origData <- as.data.frame(matrix(adj.close, ncol = time.interval,
                                   nrow = ceiling(length(adj.close)/time.interval), byrow = T))
  return(origData)
}

a <- get.data("AAPL", 9)
a
```

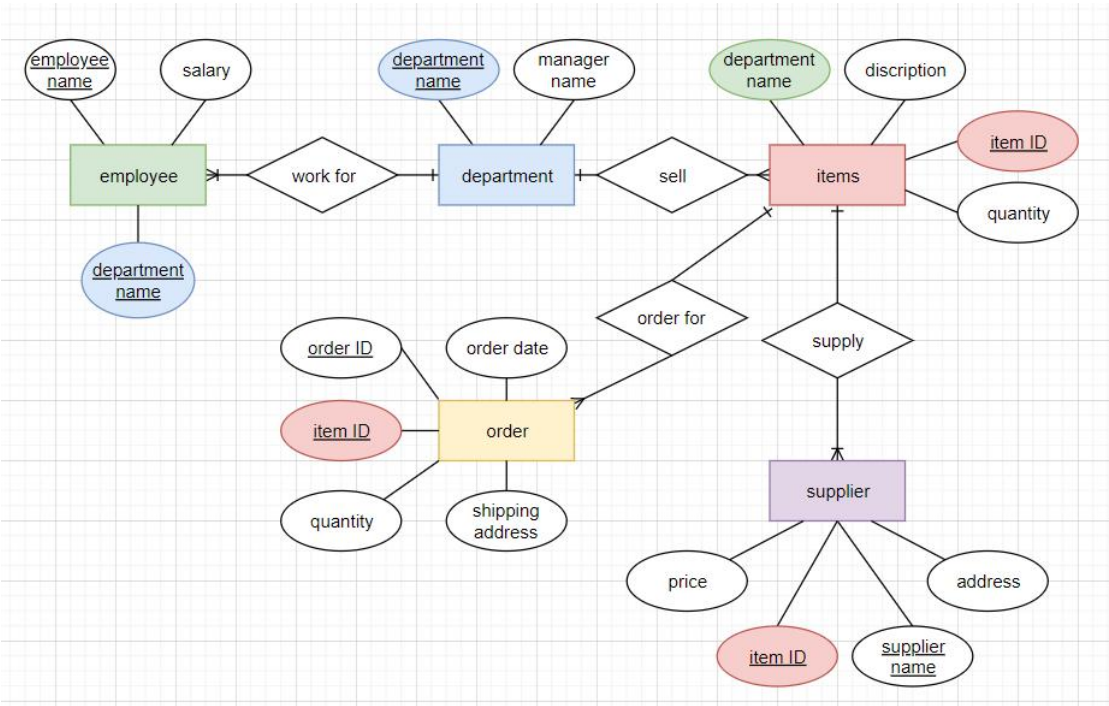
Console | Terminal

C:/Users/DELL/Desktop/FE513/Final/

```
> a <- get.data("AAPL", 9)
> a
        V1       V2       V3       V4       V5       V6       V7       V8       V9
1  110.9539 110.8297 111.3933 112.6351 113.6668 113.7815 114.3928 113.9152 113.7146
2  114.6317 114.6221 114.4215 114.6317 114.7081 114.6030 116.4275 116.4849 116.4944
3  116.1887 115.9212 122.9902 122.7800 123.3054 124.4613 125.6458 126.1330 127.0444
4  126.7566 127.8791 129.5389 130.0090 129.8555 130.2105 131.1507 131.5441 130.9876
5  131.1123 131.3714 131.4290 134.1153 133.3189 134.1057 133.6835 133.8562 133.3573
6  133.0503 133.4917 133.5492 133.3478 134.7581 134.9787 134.3072 135.7175 134.1632
7  135.6791 135.1994 134.9308 135.1610 137.9625 138.2695 138.0872 137.8282 137.8665
8  138.8931 138.1736 137.8282 137.5211 137.3581 135.8806 136.0437 135.3242 136.0724
9  135.4680 134.9691 136.6577 136.4946 137.8090 138.6629 137.8474 137.9529 137.8186
10 140.6296 141.5219 141.0901 140.5816 142.9130 146.7986 147.7388 147.0385 148.3101
11 150.3813 149.9960 149.7744 144.7456 146.9517 147.4527 148.3486 148.1656 147.7224
12 148.2330 147.9826 148.0404 147.1637 147.5683 149.7552 148.2908 148.7918 149.6781
13 149.3120 143.5221 140.0926 141.2197 139.8421 139.0040 137.0580 140.9789 139.6976
14 140.5261 140.2949 140.9211 140.4780 138.4645 140.4876 138.4163 138.7439 138.2429
15 138.8113 137.5011 138.8980 139.7458 140.1985 140.4009 142.3565 143.5800 144.0809
16 144.5819 145.4874 144.8323 144.7649 146.5182 147.1444 147.8381 145.0443 144.0231
17 143.2813 144.5530 151.3832 149.8707 150.6607 152.9921 154.2155 155.1596 150.2175
18 152.3065 154.5987 156.2912 155.6625 152.6741 152.3259 152.0454 154.5310 154.7244
19 154.0377 154.6084 156.1655 157.5582 157.9837 158.6123 158.6607 156.7554 156.5910
20 155.9623 153.4188 156.1945 155.5755 154.4053 153.0802 154.6277 153.4574 153.5155
21 150.9429 148.3509 146.9002 145.6042 148.1091 149.1633 148.2445 149.0569 148.7571
22 149.4051 148.4380 150.2852 150.1981 150.7204 150.7784 151.4071 150.8752 151.8326
23 154.6277 155.1983 154.5116 150.8558 151.1170 151.0396 151.9390 151.2717 152.2388
24 157.6935 161.2430 163.4868 161.4074 162.5873 166.8331 168.5256 169.0672 170.4503
25 170.1021 169.5391 168.8597 166.3069 164.1133 166.0740 165.1519 164.9869 168.0540
26 169.8206 169.8303 168.9761 167.9861 164.5016 166.8020 166.0254 164.8121 164.6568
27 164.0454 164.3463 164.3948 167.5979 166.6564 167.2096 167.1611 168.8597 171.2377
28 169.4129 169.2285 169.8691 169.8691 165.5596 165.5887 166.0546 164.2589       NA
>
```

Question 3:





Background color is used to show foreign keys.

```
drop table if exists orders;
drop table if exists supplier;
drop table if exists items;
drop table if exists employee;
drop table if exists department;

create table department(department_name text,
                        manager_name text,
                        primary key(department_name));

create table employee(employee_name text,
```

```
                                salary numeric(15, 2),
                                department_name text,
                                foreign key(department_name) references
department(department_name),
                                primary key(employee_name, department_name));

create table items(item_ID int primary key,
                                department_name text,
                                discription text,
                                quantity int,
                                foreign key(department_name) references department(department_name));

create table supplier(supplier_name text,
                                address text,
                                item_ID int,
                                price numeric(15, 2),
                                foreign key(item_ID) references items(item_ID),
                                primary key (supplier_name, item_ID));

create table orders(order_ID int,
                                item_ID int,
                                order_date date,
                                shipping_address text,
                                quantity int,
                                foreign key(item_ID) references items(item_ID),
                                primary key(order_ID, item_ID))
```

```
93  select * from department;
```

Data Output | Explain | Messages | Notification

department_name [PK] text	manager_name text

```
93  select * from department;
94  select * from orders;
95  select * from supplier;
96  select * from employee;
97  select * from store;
98
```

Data Output | Explain | Messages | Notifications

order_id [PK] integer	item_id [PK] integer	order_date date	shipping_address text	quantity integer

```
select * from supplier;
select * from employee;
select * from store;
```

Data Output | Explain | Messages | Notifications

supplier_name [PK] text	address text	item_id [PK] integer	price numeric (15,2)

```
select * from employee;
select * from store;
```

Data Output | Explain | Messages | Notifications

employee_name [PK] text	salary numeric (15,2)	department_name [PK] text

```
select * from items;
```

Data Output | Explain | Messages | Notifications

item_id [PK] integer	department_name text	discription text	quantity integer