

CIS 430: Lab Assignment 3

Name: Yuvaraj Vagula

ID: 2862494

Object: Querying a Relational Database COMPANY Database

1. Added myself inside EMPLOYEE, DEPENDENT, WORKS_ON table then used a single SELECT to get my entries.

```
-- Insert myself into the EMPLOYEE table
INSERT INTO EMPLOYEE (FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SEX, SALARY, SUPERSSN, DNO)
VALUES ('Yuvaraj', 'N', 'Vagula', '777777777', '2006-05-01', '2121 Euclid Ave, Cleveland, OH',
'M', 50000, '123456789', 5);

-- Insert a dependent, a Bird named Timmy, with the same ESSN value
INSERT INTO DEPENDENT (ESSN, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP)
VALUES ('777777777', 'Timmy', 'M', '2022-05-01', 'Bird');

-- Insert my data into WORKS_ON with the same ESSN value
INSERT INTO WORKS_ON VALUES ('777777777', 3, 20);
INSERT INTO WORKS_ON VALUES ('777777777', 4, 20);

-- One SELECT to get my entries I just entered
SELECT E.FNAME, E.MINIT, E.LNAME, E.SSN, E.BDATE, E.ADDRESS, E.SEX, E.SALARY, E.SUPERSSN,
E.DNO,
      D.DEPENDENT_NAME, D.SEX AS DEP_SEX, D.BDATE AS DEP_BDATE, D.RELATIONSHIP,
      W.PNO, W.HOURS
FROM EMPLOYEE E
LEFT JOIN DEPENDENT D ON E.SSN = D.ESSN
LEFT JOIN WORKS_ON W ON E.SSN = W.ESSN
```

WHERE E.SSN = '77777777';

Lab 3.sql - LAPTOP-LSLNUDIE\COMPANY_yVagula (LAPTOP-LSLNUDIE\Yuvraj (66)) - Microsoft SQL Server Management Studio

Object Explorer: LAPTOP-LSLNUDIE (SQL Server 15.0.2116) > Databases > COMPANY_yVagula > Tables > EMPLOYEE

```

171 -- Insert myself into the EMPLOYEE table
172 INSERT INTO EMPLOYEE (FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SEX, SALARY, SUPERSRN, DNO)
173 VALUES ('Yuvraj', 'M', 'Vagula', '77777777', '2006-05-01', '2121 Euclid Ave, Cleveland, OH', 'M', 50000, '123456789', 5);
174
175 -- Insert a dependent, a Bird named Timmy, with the same SSN value
176 INSERT INTO DEPENDENT (ESSN, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP)
177 VALUES ('77777777', 'Timmy', 'M', '2022-05-01', 'Bird');
178
179 -- Insert my data into WORKS_ON with the same SSN value
180 INSERT INTO WORKS_ON VALUES ('77777777', 3, 20);
181
182 INSERT INTO WORKS_ON VALUES ('77777777', 4, 20);
183
184
185 -- One SELECT to get my entries I just entered
186 SELECT E.FNAME, E.MINIT, E.LNAME, E.SSN, E.BDATE, E.ADDRESS, E.SEX, E.SALARY, E.SUPERSRN, E.DNO,
187        D.DEPENDENT_NAME, D.SEX AS DEP_SEX, D.BDATE AS DEP_BDATE, D.RELATIONSHIP,
188        W.PNO, W.HOURS
189 FROM EMPLOYEE E
190 LEFT JOIN DEPENDENT D ON E.SSN = D.ESSN
191 LEFT JOIN WORKS_ON W ON E.SSN = W.ESSN
192 WHERE E.SSN = '77777777';
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Results: Messages

	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSRN	DNO	DEPENDENT_NAME	DEP_SEX	DEP_BDATE	RELATIONSHIP	PNO	HOURS
1	Yuvraj	N	Vagula	77777777	2006-05-01	2121 Euclid Ave, Cleveland, OH	M	50000.00	123456789	5	Timmy	M	2022-05-01	Bird	1	20.0
2	Yuvraj	N	Vagula	77777777	2006-05-01	2121 Euclid Ave, Cleveland, OH	M	50000.00	123456789	5	Timmy	M	2022-05-01	Bird	3	20.0

Query executed successfully.

Q1) Getting female managers names & addresses

```

-- Q1: Retrieve the name and address of all the female managers
SELECT E.FNAME, E.LNAME, E.ADDRESS
FROM EMPLOYEE E
JOIN DEPARTMENT D ON E.SSN = D.MGRSSN
WHERE E.SEX = 'F';

```

Lab 3.sql - LAPTOP-LSLNUDIE\COMPANY_yVagula (LAPTOP-LSLNUDIE\Yuvraj (66)) - Microsoft SQL Server Management Studio

Object Explorer: LAPTOP-LSLNUDIE (SQL Server 15.0.2116) > Databases > COMPANY_yVagula > Tables > EMPLOYEE

```

183
184
185 -- One SELECT to get my entries I just entered
186 SELECT E.FNAME, E.MINIT, E.LNAME, E.SSN, E.BDATE, E.ADDRESS, E.SEX, E.SALARY, E.SUPERSRN, E.DNO,
187        D.DEPENDENT_NAME, D.SEX AS DEP_SEX, D.BDATE AS DEP_BDATE, D.RELATIONSHIP,
188        W.PNO, W.HOURS
189 FROM EMPLOYEE E
190 LEFT JOIN DEPENDENT D ON E.SSN = D.ESSN
191 LEFT JOIN WORKS_ON W ON E.SSN = W.ESSN
192 WHERE E.SSN = '77777777';
193
194
195
196 -- Q1: Retrieve the name and address of all the female managers
197 SELECT E.FNAME, E.LNAME, E.ADDRESS
198 FROM EMPLOYEE E
199 JOIN DEPARTMENT D ON E.SSN = D.MGRSSN
200 WHERE E.SEX = 'F';
201
202 -- Q2: List all project numbers for projects involving an employee whose last name is 'Smith'
203 SELECT DISTINCT P.PNUMBER
204 FROM PROJECT P
205 JOIN DEPARTMENT D ON P.DNUM = D.DNUMBER
206 JOIN EMPLOYEE E1 ON D.MGRSSN = E1.SSN
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Results: Messages

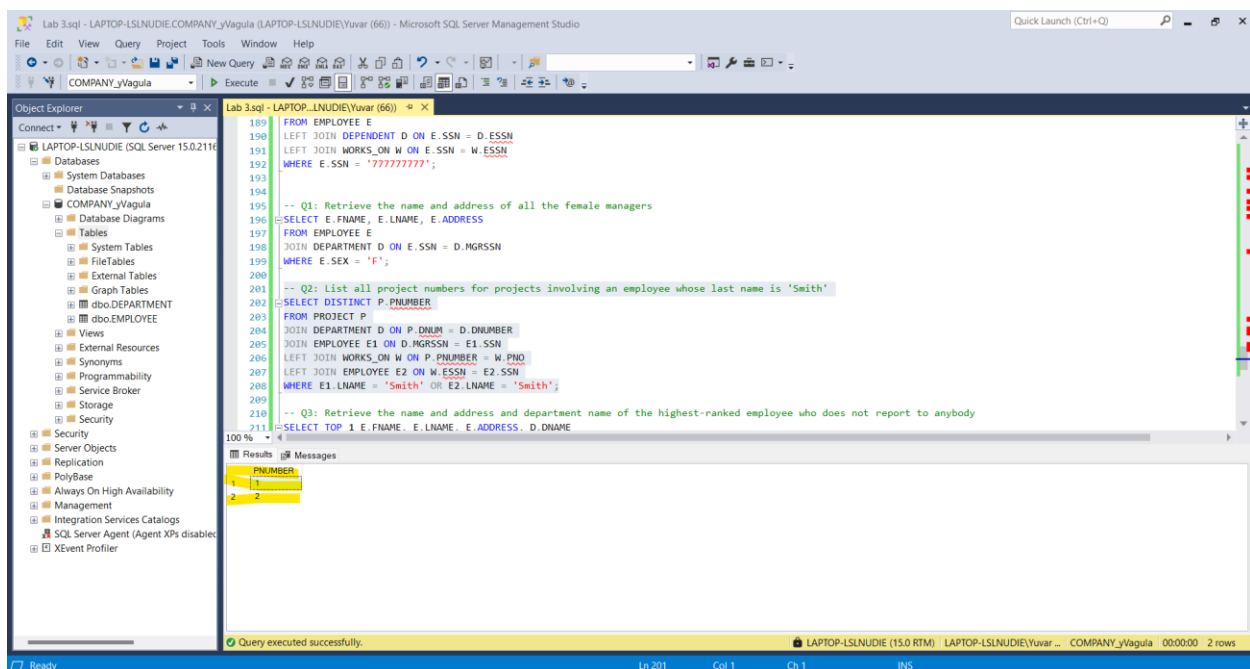
	FNAME	LNAME	ADDRESS
1	Jennifer	Wallace	291 Berry, Bellaire, TX

Query executed successfully.

Q2) A list of all project numbers for projects that involve an employee whose last name is 'Smith', either as a worker or as a manager of the department that controls the project.

-- Q2: List all project numbers for projects involving an employee whose last name is 'Smith'

```
SELECT DISTINCT P.PNUMBER
FROM PROJECT P
JOIN DEPARTMENT D ON P.DNUM = D.DNUMBER
JOIN EMPLOYEE E1 ON D.MGRSSN = E1.SSN
LEFT JOIN WORKS_ON W ON P.PNUMBER = W.PNO
LEFT JOIN EMPLOYEE E2 ON W.ESSN = E2.SSN
WHERE E1.LNAME = 'Smith' OR E2.LNAME = 'Smith';
```



Q3) Retrieve the name and address and his/her department name of the highest ranked employee who does not report to anybody in the company.

-- Q3: Retrieve the name and address and department name of the highest-ranked employee who does not report to anybody

```
SELECT TOP 1 E.FNAME, E.LNAME, E.ADDRESS, D.DNAME
FROM EMPLOYEE E
JOIN DEPARTMENT D ON E.DNO = D.DNUMBER
WHERE E.SUPERSSN IS NULL
ORDER BY E.SALARY DESC;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The query window displays the following SQL code:

```
-- Q3: Retrieve the name and address and department name of the highest-ranked employee who does not report to anybody
SELECT TOP 1 E.FNAME, E.LNAME, E.ADDRESS, D.DNAME
FROM EMPLOYEE E
JOIN DEPARTMENT D ON E.DNO = D.DNUMBER
WHERE E.SUPERSSN IS NULL
ORDER BY E.SALARY DESC;
```

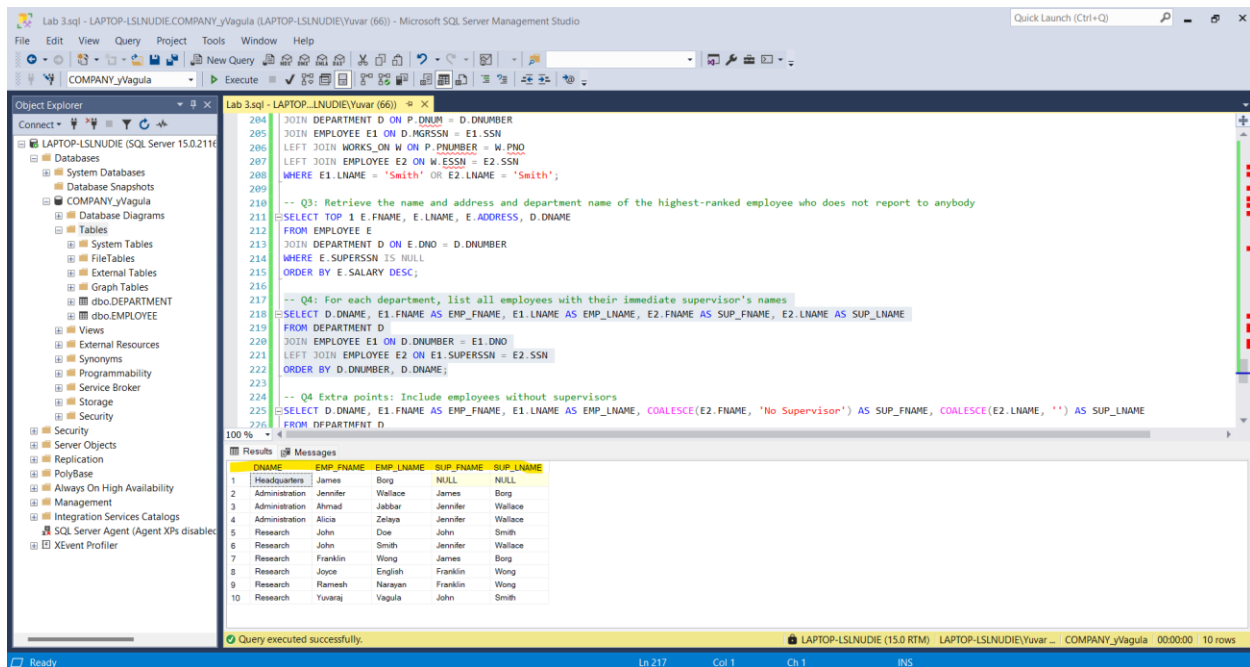
The Results pane shows the following data:

	FNAME	LNAME	ADDRESS	DNAME
1	James	Burg	450 Stone, Houston, TX	Headquarters

The status bar at the bottom indicates: Query executed successfully. LAPTOP-LSLNUDIE (15.0 RTM) LAPTOP-LSLNUDIE\Yuv... COMPANY_yVagula 00:00:00 1 rows

Q4) For each department, list all the employees who are working in the department with the employee's first and last name and first and last name of his or her immediate supervisor. List the result in the order of each department number and department name.

```
-- Q4: For each department, list all employees with their immediate supervisor's names
SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, E2.FNAME AS SUP_FNAME,
E2.LNAME AS SUP_LNAME
FROM DEPARTMENT D
JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
ORDER BY D.DNUMBER, D.DNAME;
```



The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The query window displays the following SQL code:

```
-- Q4: For each department, list all employees with their immediate supervisor's names
SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, E2.FNAME AS SUP_FNAME,
E2.LNAME AS SUP_LNAME
FROM DEPARTMENT D
JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
ORDER BY D.DNUMBER, D.DNAME;
```

The Results pane shows the following data:

	DNAME	EMP_FNAME	EMP_LNAME	SUP_FNAME	SUP_LNAME
1	Headquarters	James	Borg	NULL	NULL
2	Administration	Jennifer	Wallace	James	Borg
3	Administration	Ahmad	Jabbar	Jennifer	Wallace
4	Administration	Alicia	Zelaya	Jennifer	Wallace
5	Research	John	Due	John	Smith
6	Research	John	Smith	Jennifer	Wallace
7	Research	Franklin	Wong	James	Borg
8	Research	Joyce	English	Franklin	Wong
9	Research	Ramesh	Narayan	Franklin	Wong
10	Research	Yuvraj	Vagula	John	Smith

The status bar at the bottom indicates: Query executed successfully. LAPTOP-LSLNUDIE (15.0 RTM) LAPTOP-LSLNUDIE\Yuvraj... COMPANY_yVagula 00:00:00 10 rows

EXTRA POINTS) Include all the employees who do not have any supervisor as well in the list.

```
-- Extra points: Include employees without supervisors
SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, COALESCE(E2.FNAME, 'No
Supervisor') AS SUP_FNAME, COALESCE(E2.LNAME, '') AS SUP_LNAME
FROM DEPARTMENT D
JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
ORDER BY D.DNUMBER, D.DNAME;
```

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the database structure of 'COMPANY_yVagula'. The right pane shows a query window with the following SQL code:

```
-- Q3: Retrieve the name and address of the highest-ranked employee who does not report to anybody
SELECT TOP 1 E.FNAME, E.LNAME, E.ADDRESS, D.DNAME
FROM EMPLOYEE E
JOIN DEPARTMENT D ON E.DNO = D.DNUMBER
WHERE E.SUPERSSN IS NULL
ORDER BY E.SALARY DESC;

-- Q4: For each department, list all employees with their immediate supervisor's names
SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, E2.FNAME AS SUP_FNAME, E2.LNAME AS SUP_LNAME
FROM DEPARTMENT D
JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
ORDER BY D.DNUMBER, D.DNAME;

-- Extra points: Include employees without supervisors
SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, COALESCE(E2.FNAME, 'No Supervisor') AS SUP_FNAME, COALESCE(E2.LNAME, '') AS SUP_LNAME
FROM DEPARTMENT D
JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
ORDER BY D.DNUMBER, D.DNAME;
```

The bottom pane shows the results of the query, displaying a table with 10 rows and 5 columns: DNAME, EMP_FNAME, EMP_LNAME, SUP_FNAME, and SUP_LNAME.

	DNAME	EMP_FNAME	EMP_LNAME	SUP_FNAME	SUP_LNAME
1	Headquarters	James	Borg	No Supervisor	
2	Administration	Jennifer	Wallace	James	Borg
3	Administration	Abmad	Jabbar	Jennifer	Wallace
4	Administration	Alicia	Zelaya	Jennifer	Wallace
5	Research	John	Doe	John	Smith
6	Research	John	Smith	Jennifer	Wallace
7	Research	Franklin	Wong	James	Borg
8	Research	Joyce	English	Franklin	Wong
9	Research	Ramesh	Narayan	Franklin	Wong
10	Research	Yuvraj	Vagula	John	Smith

The status bar at the bottom indicates: Query executed successfully. LAPTOP-LSLNUDIE (15.0 RTM) LAPTOP-LSLNUDIEYuvraj... COMPANY_yVagula 00:00:00 10 rows

PROOF OF PATH AND FILENAME

The screenshot displays the Microsoft SQL Server Management Studio interface. The left pane shows the Object Explorer with the database structure of 'COMPANY_yVagula'. The central pane shows a SQL query being executed. The query consists of two parts: Q1, which selects the top 1 employee by salary, and Q4, which lists all employees with their immediate supervisor's names. The results pane at the bottom shows the output of the query, displaying a table with columns DNAME, EMP_FNAME, EMP_LNAME, SUP_FNAME, and SUP_LNAME. The status bar at the bottom indicates that the query was executed successfully.

```

210 -- Q1: Retrieve the name and address and department name of the highest-salaried employee who does not report to anybody.
211 SELECT TOP 1
212 FROM EMPLOYEE E
213 JOIN DEPARTMENT D ON E.DNO = D.DNUMBER
214 WHERE E.SUPERSSN IS NULL
215 ORDER BY E.SALARY DESC;
216
217 -- Q4: For each department, list all employees with their immediate supervisor's names
218 SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, E2.FNAME AS SUP_FNAME, E2.LNAME AS SUP_LNAME
219 FROM DEPARTMENT D
220 JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
221 LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
222 ORDER BY D.DNUMBER, D.DNAME;
223
224 -- Extra points: Include employees without supervisors
225 SELECT D.DNAME, E1.FNAME AS EMP_FNAME, E1.LNAME AS EMP_LNAME, COALESCE(E2.FNAME, 'No Supervisor') AS SUP_FNAME, COALESCE(E2.LNAME, '') AS SUP_LNAME
226 FROM DEPARTMENT D
227 JOIN EMPLOYEE E1 ON D.DNUMBER = E1.DNO
228 LEFT JOIN EMPLOYEE E2 ON E1.SUPERSSN = E2.SSN
229 ORDER BY D.DNUMBER, D.DNAME;
230
231

```

	DNAME	EMP_FNAME	EMP_LNAME	SUP_FNAME	SUP_LNAME
1	Headquarters	James	Borg	No Supervisor	
2	Administration	Jennifer	Wallace	James	Borg
3	Administration	Alfred	Jalbar	Jennifer	Wallace
4	Administration	Alicia	Zelma	Jennifer	Wallace
5	Research	John	Doe	John	Smith
6	Research	John	Smith	Jennifer	Wallace
7	Research	Franklin	Wong	James	Borg
8	Research	Joyce	English	Franklin	Wong
9	Research	Ramesh	Narayan	Franklin	Wong
10	Research	Yusang	Vagula	John	Smith

Query executed successfully.