I Silas Davis declare that I have completed this assignment completely and entirely on my own, without any consultation with others. I understand that any breach of the UAB Academic Honor Code may result in severe penalties.

Silas Davis

1. D -> DCA. CE -> CEAD. AE -> AEDC.
   No closure determines all attributes, no key found.
   Since attributes B and E are not present in the RHS of any functional dependencies, they will be apart of candidate key.
   BDE->BDECA. BCE->BCEAD. ABE->ABECD. These will be the candidate keys.

2. Firstly we can conclude that any functional dependencies which contain E attribute will be ignored since the given relation S does not have the attribute E present. That means at AB->C and BC->D are the FDs that remain, and they will hold. AB->C can translate to A->C and B->C. BC->D can translate to B->D and C->D. So using transitive dependency, A->D is true. FDs like ABC->D, BD->C, and AC->B will hold in S.

3. A. The candidate key will be AB, AC, BD.
   BD->C = BDCA. AB->D = ABDC. AC->B = ACBD.
   A->B will be in BCNF if A is super key, therefore a is in BCNF.

   B. The candidate key will be AC, D.
   AC->D, AC super key. D->A, D super key. D->C, D super key. D->B, D super key.
   Therefore, relation is in BCNF.

   C. The candidate key will be A, C, D.
   C->D = CBDA. D->A = DACB. C->D = CBDA. A->C = ACDB.
   This relation is already in BCNF.

   D. The candidate key is AD, CD, BD.
   BC->A = BCA, not super key. AD->C = ADCB, super key. CD->B = CDBA, super key. BD->C = BDCA. Loss less join, relation not in BCNF.

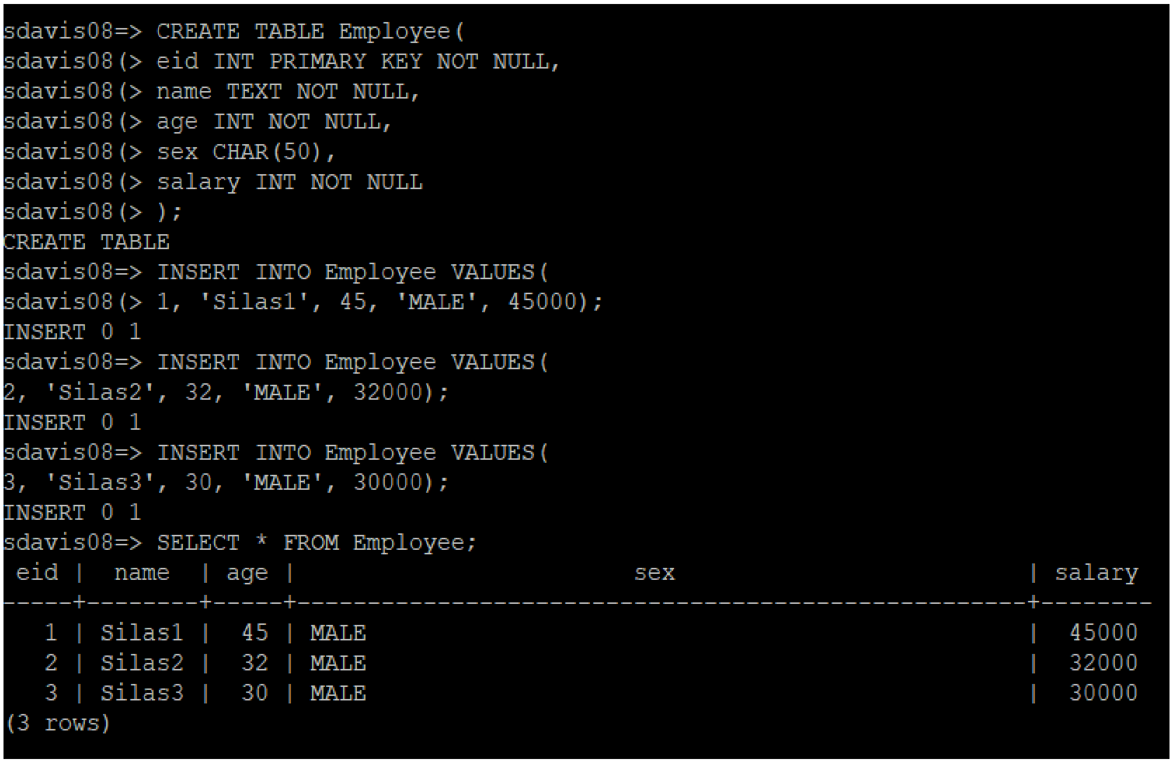
4. Since all table are not dependent on one key, which is the primary key, then it is not in 3NF.
   A->BCDE

A->EF

A->C

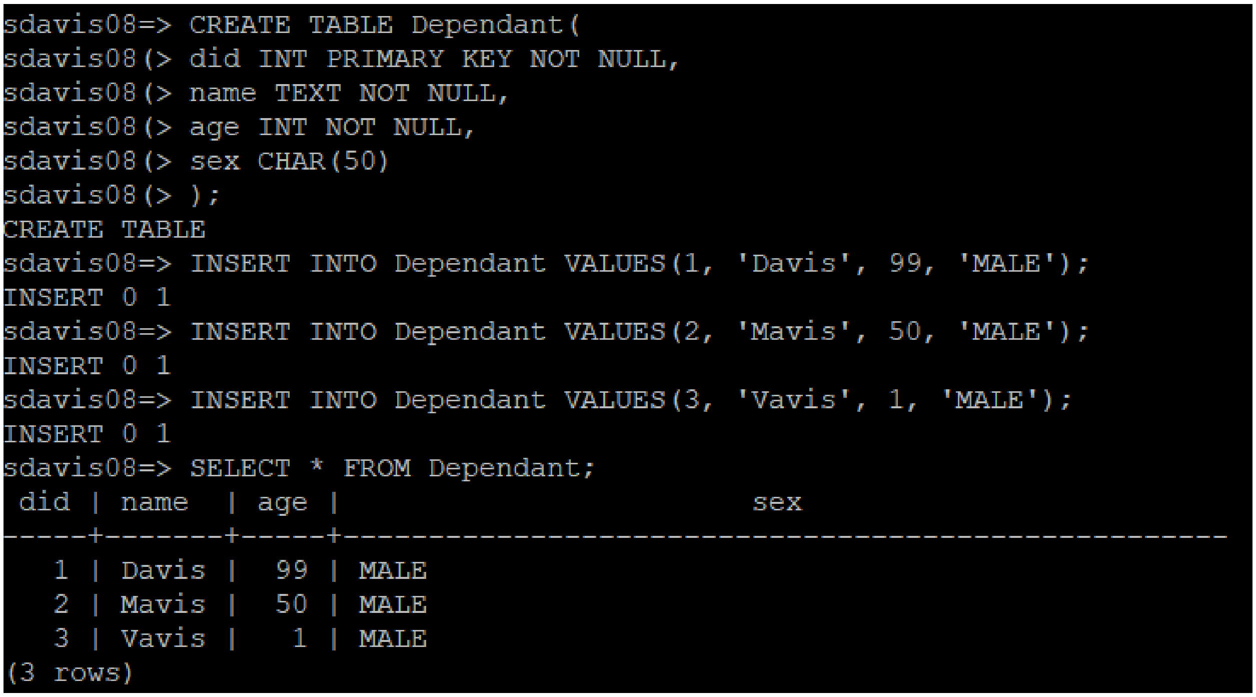
Since all table are dependent on A, the primary key, we can fetch all table data with “A”.

5. A.



```
sdavis08=> CREATE TABLE Employee(
sdavis08(> eid INT PRIMARY KEY NOT NULL,
sdavis08(> name TEXT NOT NULL,
sdavis08(> age INT NOT NULL,
sdavis08(> sex CHAR(50),
sdavis08(> salary INT NOT NULL
sdavis08(> );
CREATE TABLE
sdavis08=> INSERT INTO Employee VALUES(
sdavis08(> 1, 'Silas1', 45, 'MALE', 45000);
INSERT 0 1
sdavis08=> INSERT INTO Employee VALUES(
2, 'Silas2', 32, 'MALE', 32000);
INSERT 0 1
sdavis08=> INSERT INTO Employee VALUES(
3, 'Silas3', 30, 'MALE', 30000);
INSERT 0 1
sdavis08=> SELECT * FROM Employee;
 eid |  name  | age |                        sex                         | salary
-----+--------+-----+----------------------------------------------------+--------
   1 | Silas1 |  45 | MALE                                               |  45000
   2 | Silas2 |  32 | MALE                                               |  32000
   3 | Silas3 |  30 | MALE                                               |  30000
(3 rows)
```

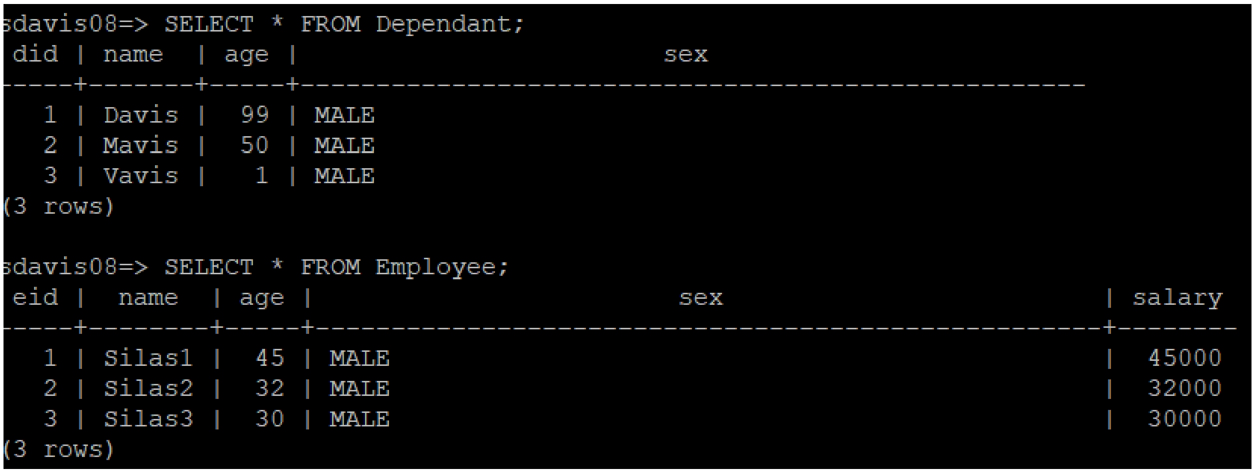
B.



```
sdavis08=> CREATE TABLE Dependant(
sdavis08(> did INT PRIMARY KEY NOT NULL,
sdavis08(> name TEXT NOT NULL,
sdavis08(> age INT NOT NULL,
sdavis08(> sex CHAR(50)
sdavis08(> );
CREATE TABLE
sdavis08=> INSERT INTO Dependant VALUES(1, 'Davis', 99, 'MALE');
INSERT 0 1
sdavis08=> INSERT INTO Dependant VALUES(2, 'Mavis', 50, 'MALE');
INSERT 0 1
sdavis08=> INSERT INTO Dependant VALUES(3, 'Vavis', 1, 'MALE');
INSERT 0 1
sdavis08=> SELECT * FROM Dependant;
 did | name  | age |                        sex
-----+-------+-----+----------------------------------------------------
   1 | Davis |  99 | MALE
   2 | Mavis |  50 | MALE
   3 | Vavis |   1 | MALE
(3 rows)
```

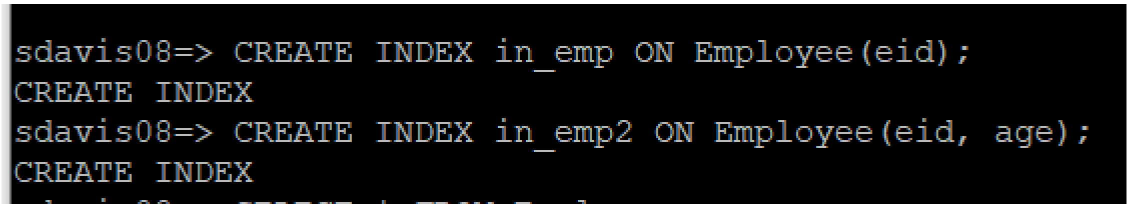
C.

D. In the Employee table, the eid column is the primary key and the did column is the primary key for the Dependant table.



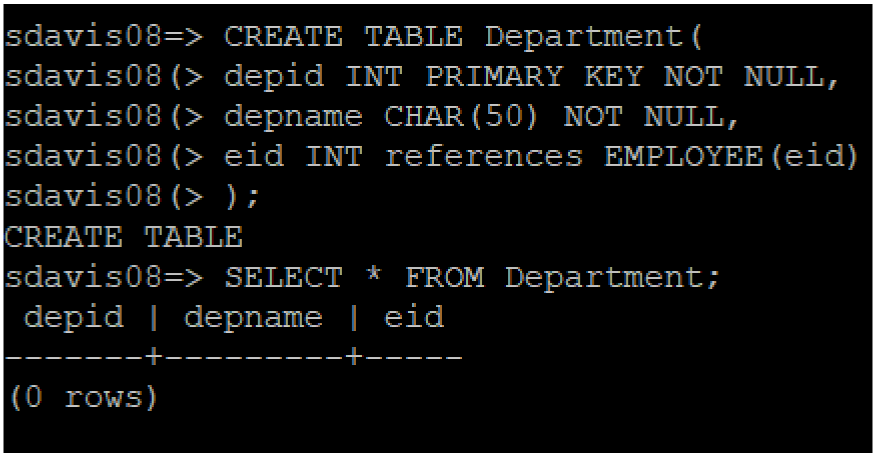
```
sdavis08=> SELECT * FROM Dependant;
 did | name  | age |                        sex
-----+-------+-----+----------------------------------------------------
   1 | Davis |  99 | MALE
   2 | Mavis |  50 | MALE
   3 | Vavis |   1 | MALE
(3 rows)

sdavis08=> SELECT * FROM Employee;
 eid |  name  | age |                        sex                         | salary
-----+--------+-----+----------------------------------------------------+--------
   1 | Silas1 |  45 | MALE                                               |  45000
   2 | Silas2 |  32 | MALE                                               |  32000
   3 | Silas3 |  30 | MALE                                               |  30000
(3 rows)
```

E.



```
sdavis08=> CREATE INDEX in_emp ON Employee(eid);
CREATE INDEX
sdavis08=> CREATE INDEX in_emp2 ON Employee(eid, age);
CREATE INDEX
```

F.



```
sdavis08=> CREATE TABLE Department(
sdavis08(> depid INT PRIMARY KEY NOT NULL,
sdavis08(> depname CHAR(50) NOT NULL,
sdavis08(> eid INT references EMPLOYEE(eid)
sdavis08(> );
CREATE TABLE
sdavis08=> SELECT * FROM Department;
 depid | depname | eid
-------+---------+-----
(0 rows)
```