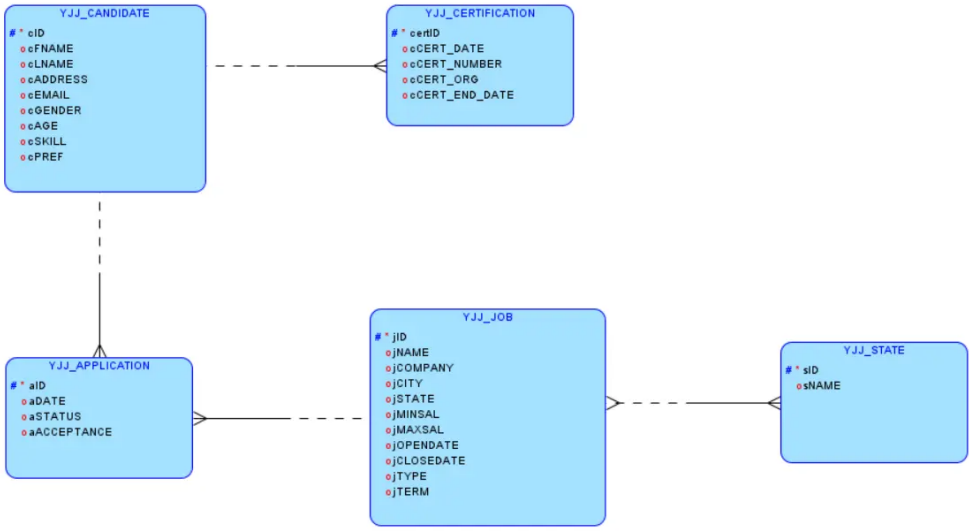
YJJ– Assignment

jh8824 Jiawei He CS–GY_6083_1_B

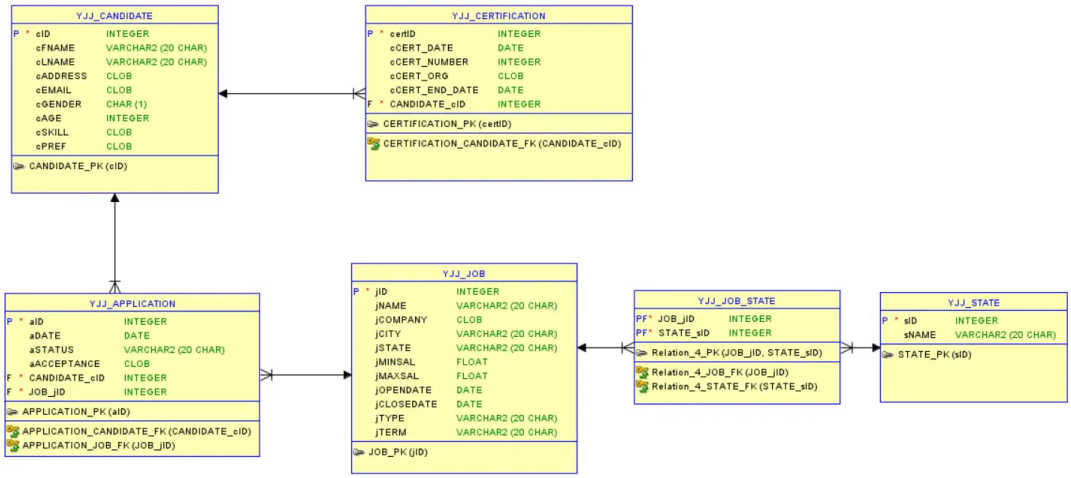
Submission Date: 3.2.2024

Problem 1:

1) Logical Model



2) Relational Model



3) Any valid assumptions made

- A single candidate can have multiple certifications.
- Certifications are specific to a candidate and not shared among candidates.
- A job can be associated with multiple states, and a state can have multiple jobs.

4) DDL code

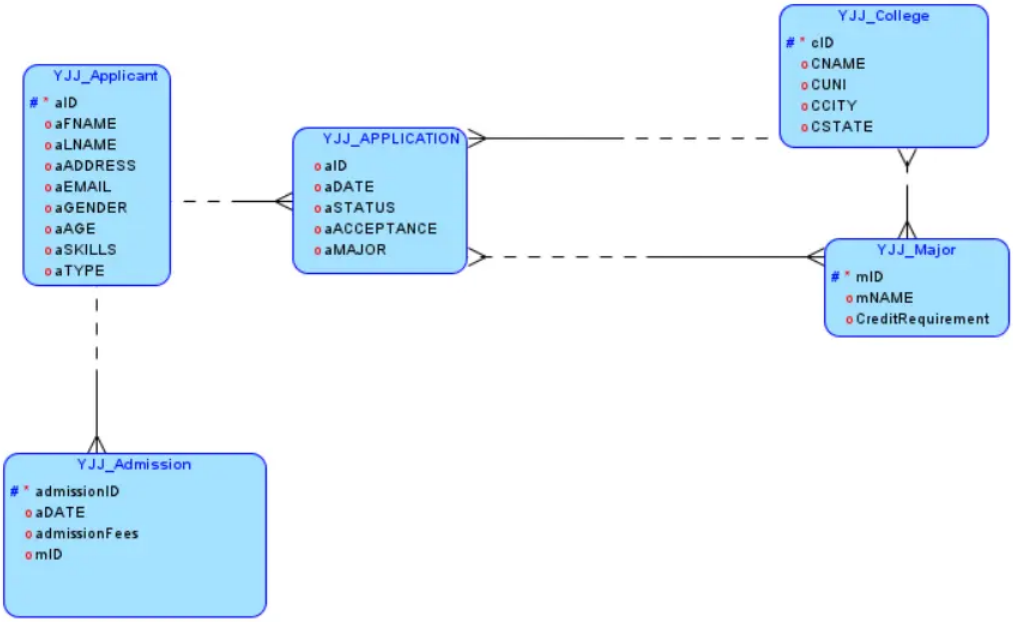
CREATE TABLE yjj_application (

```
    aid             INTEGER NOT NULL,
    adate           DATE,
    astatus         VARCHAR2(20 CHAR),
    aacceptance     CLOB,
    candidate_cid INTEGER NOT NULL,
    job_jid         INTEGER NOT NULL
);

ALTER TABLE yjj_application ADD CONSTRAINT application_pk PRIMARY KEY ( aid );

CREATE TABLE yjj_candidate (
    cid       INTEGER NOT NULL,
    cfname    VARCHAR2(20 CHAR),
    clname    VARCHAR2(20 CHAR),
    caddress CLOB,
    cemail    CLOB,
    cgender  CHAR(1),
    cage      INTEGER,
    cskill    CLOB,
    cpref     CLOB
);

ALTER TABLE yjj_candidate ADD CONSTRAINT candidate_pk PRIMARY KEY ( cid );

CREATE TABLE yjj_certification (
    certid          INTEGER NOT NULL,
    ccert_date      DATE,
    ccert_number    INTEGER,
    ccert_org       CLOB,
    ccert_end_date DATE,
    candidate_cid  INTEGER NOT NULL
);

ALTER TABLE yjj_certification ADD CONSTRAINT certification_pk PRIMARY KEY ( certid
);

CREATE TABLE yjj_job (
    jid         INTEGER NOT NULL,
    jname       VARCHAR2(20 CHAR),
    jcompany    CLOB,
    jcity       VARCHAR2(20 CHAR),
    jstate      VARCHAR2(20 CHAR),
    jminsal     FLOAT,
    jmaxsal     FLOAT,
    jopendate  DATE,
    jclosedate DATE,
    jtype       VARCHAR2(20 CHAR),
```

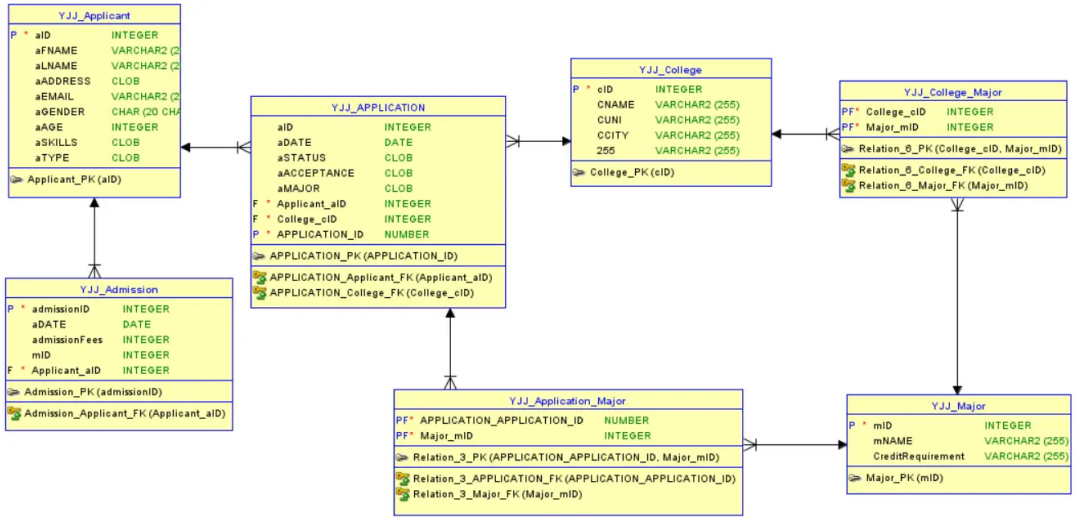
```
    jterm       VARCHAR2(20 CHAR)
);

ALTER TABLE yjj_job ADD CONSTRAINT job_pk PRIMARY KEY ( jid );

CREATE TABLE yjj_job_state (
    job_jid    INTEGER NOT NULL,
    state_sid INTEGER NOT NULL
);

ALTER TABLE yjj_job_state ADD CONSTRAINT relation_4_pk PRIMARY KEY ( job_jid,
                                                                    state_sid );

CREATE TABLE yjj_state (
    sid    INTEGER NOT NULL,
    sname VARCHAR2(20 CHAR)
);

ALTER TABLE yjj_state ADD CONSTRAINT state_pk PRIMARY KEY ( sid );

ALTER TABLE yjj_application
    ADD CONSTRAINT application_candidate_fk FOREIGN KEY ( candidate_cid )
        REFERENCES yjj_candidate ( cid );

ALTER TABLE yjj_application
    ADD CONSTRAINT application_job_fk FOREIGN KEY ( job_jid )
        REFERENCES yjj_job ( jid );

ALTER TABLE yjj_certification
    ADD CONSTRAINT certification_candidate_fk FOREIGN KEY ( candidate_cid )
        REFERENCES yjj_candidate ( cid );

ALTER TABLE yjj_job_state
    ADD CONSTRAINT relation_4_job_fk FOREIGN KEY ( job_jid )
        REFERENCES yjj_job ( jid );

ALTER TABLE yjj_job_state
    ADD CONSTRAINT relation_4_state_fk FOREIGN KEY ( state_sid )
        REFERENCES yjj_state ( sid );
```

Problem 2:

1) Logical Model



2) Relational Model



3) Any valid assumptions made

- An application can apply for multiple majors, but for simplicity, we've associated aMAJOR directly with APPLICATION. In a more complex model, a separate associative entity between APPLICATION and MAJOR could be considered.
- aSKILLS are stored in a single field, simplifying the model.

4) DDL code

```
CREATE TABLE yjj_admission (
    admissionid    INTEGER NOT NULL,
    adate          DATE,
    admissionfees INTEGER,
    mid            INTEGER,
    applicant_aid INTEGER NOT NULL
);

ALTER TABLE yjj_admission ADD CONSTRAINT admission_pk PRIMARY KEY (
admissionid );
```

```
CREATE TABLE yjj_applicant (
    aid       INTEGER NOT NULL,
    afname    VARCHAR2(255),
    alname    VARCHAR2(255),
    aaddress CLOB,
    aemail    VARCHAR2(255),
    agender  CHAR(20 CHAR),
    aage      INTEGER,
    askills  CLOB,
    atype     CLOB
);

ALTER TABLE yjj_applicant ADD CONSTRAINT applicant_pk PRIMARY KEY ( aid );

CREATE TABLE yjj_application (
    aid              INTEGER,
    adate            DATE,
    astatus          CLOB,
    aacceptance     CLOB,
    amajor           CLOB,
    applicant_aid  INTEGER NOT NULL,
    college_cid     INTEGER NOT NULL,
    application_id NUMBER NOT NULL
);

ALTER TABLE yjj_application ADD CONSTRAINT application_pk PRIMARY KEY (
application_id );

CREATE TABLE yjj_application_major (
    application_application_id NUMBER NOT NULL,
    major_mid                  INTEGER NOT NULL
);

ALTER TABLE yjj_application_major ADD CONSTRAINT relation_3_pk PRIMARY KEY (
application_application_id,
                                                        major_mid );

CREATE TABLE yjj_college (
    cid    INTEGER NOT NULL,
    cname VARCHAR2(255),
    cuni  VARCHAR2(255),
    ccity VARCHAR2(255),
    ”255” VARCHAR2(255)
);

ALTER TABLE yjj_college ADD CONSTRAINT college_pk PRIMARY KEY ( cid );
```

```
CREATE TABLE yjj_college_major (
    college_cid INTEGER NOT NULL,
    major_mid   INTEGER NOT NULL
);

ALTER TABLE yjj_college_major ADD CONSTRAINT relation_6_pk PRIMARY KEY (
college_cid,
                                                        major_mid );

CREATE TABLE yjj_major (
    mid               INTEGER NOT NULL,
    mname               VARCHAR2(255),
    creditrequirement VARCHAR2(255)
);

ALTER TABLE yjj_major ADD CONSTRAINT major_pk PRIMARY KEY ( mid );

ALTER TABLE yjj_admission
    ADD CONSTRAINT admission_applicant_fk FOREIGN KEY ( applicant_aid )
        REFERENCES yjj_applicant ( aid );

ALTER TABLE yjj_application
    ADD CONSTRAINT application_applicant_fk FOREIGN KEY ( applicant_aid )
        REFERENCES yjj_applicant ( aid );

ALTER TABLE yjj_application
    ADD CONSTRAINT application_college_fk FOREIGN KEY ( college_cid )
        REFERENCES yjj_college ( cid );

ALTER TABLE yjj_application_major
    ADD CONSTRAINT relation_3_application_fk FOREIGN KEY (
application_application_id )
        REFERENCES yjj_application ( application_id );

ALTER TABLE yjj_application_major
    ADD CONSTRAINT relation_3_major_fk FOREIGN KEY ( major_mid )
        REFERENCES yjj_major ( mid );

ALTER TABLE yjj_college_major
    ADD CONSTRAINT relation_6_college_fk FOREIGN KEY ( college_cid )
        REFERENCES yjj_college ( cid );

ALTER TABLE yjj_college_major
    ADD CONSTRAINT relation_6_major_fk FOREIGN KEY ( major_mid )
        REFERENCES yjj_major ( mid );

CREATE SEQUENCE yjj_application_application_id START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER yjj_application_application_id BEFORE
    INSERT ON yjj_application
    FOR EACH ROW
    WHEN ( new.application_id IS NULL )
BEGIN
    :new.application_id := yjj_application_application_id.nextval;
END;
/
```