```sql
create database university_main
    with owner = postgres
    template = template0
    encoding = 'UTF8';

create database university_archive
    with template = template0
    connection limit = 50;

create database university_archive
    with template = template0
    connection limit = 50;

UPDATE pg_database
SET datistemplate = TRUE
WHERE datname = 'university_test';
```

```sql
create tablespace student_data location '/data/students';
create tablespace course_data owner postgre location '/data/courses';
create database university_distributed
    with tablespace = student_data
    template = template0
    encoding = 'LATIN9';
```

```sql
create table students(
    student_id      serial primary key,
    first_name      varchar(50) not null,
    last_name       varchar(50) not null,
    email           varchar(100),
    phone           char(15),
    date_of_birth   date,
    enrollment_date date,
    gpa             numeric(4,2),
    is_active       bool,
    graduation_year smallint
);

create table professors(
    professor_id    serial primary key,
    first_name      varchar(50) not null,
    last_name       varchar(50) not null,
    email           varchar(100),
    office_number   varchar(20),
    hire_date       date,
    salsry          numeric(12,2),
    is_tenured      bool,
    years_experience int
);

create table courses(
    course_id       serial primary key,
    course_code     varchar(8),
    course_title    varchar(100),
    description     text,
    credits         smallint,
    max_enrollment  int,
    course_fee      numeric(10,2),
    is_online       bool,
    created_at      timestamp without time zone
);
```

```sql
create table class_schedule(
    schedule_id     serial primary key,
    course_id       int,
    professor_id    int,
    classroom       varchar(20),
    class_date      date,
    start_time      time without time zone,
    end_time        time without time zone,
    duration        interval
);

create table student_records(
    record_id           serial primary key,
    student_id          int,
    course_id           int,
    semester            varchar(20),
    year                int,
    grade               char(2),
    attendance_persentage numeric(4,1),
    submission_timestamp  timestamp without time zone,
    last_update         timestamp without time zone
);
```

```sql
alter table students add column middle_name varchar(30);
alter table students add column student_status varchar(20);
alter table students alter column phone type varchar(20);
alter table students alter column student_status set default 'ACTIVE';
alter table students alter column gpa set default 0.00;

alter table professors add column department_code char(5);
alter table professors add column research_area text;
alter table professors alter column yers_experience type smallint using yers_experience::smallint ;
alter table professors alter column is_tenured set default false;
alter table professors add column last_promotion_date date;

alter table courses add column prerequisite_course_id int;
alter table courses add column difficulty_level smallint;
alter table courses alter column course_code type varchar(10) USING course_code::varchar;
alter table courses alter column credits set default 3;
alter table courses add column lab_required bool default false;
```

```sql
alter table class_schedule rename to class_schedual;

alter table class_schedual add column room_capacity int;
alter table class_schedual drop column duration;
alter table class_schedual add column session_type varchar(15);
alter table class_schedual alter column classroom type varchar(30) using classroom::varchar;
alter table class_schedual add column equipment_needed text;

alter table student_records add column extra_credit_points numeric(4,1);
alter table student_records alter column grade type varchar(5) using grade::varchar;
alter table student_records alter column extra_credit_points set default 0.0;
alter table student_records add column final_exam_date date;
alter table student_records drop column last_update;
```

```sql
create table departments(
    department_id       serial primary key,
    department_name     varchar(100),
    department_code     char(5),
    building            varchar(50),
    phone               varchar(15),
    budget              numeric(14,2),
    established_year    int
);

create table library_books(
    book_id              serial PRIMARY KEY,
    isbn                 char(13),
    title                varchar(200),
    author               varchar(100),
    publisher            varchar(100),
    publication_date     date,
    price                numeric(8,2),
    is_available         bool,
    acquisition_timestamp timestamp without time zone
);
```

```sql
create table student_book_loans(
    loan_id     serial PRIMARY KEY,
    student_id  integer,
    book_id     integer,
    loan_date   date,
    due_date    date,
    return_date date,
    fine_amount numeric(8,2),
    loan_status varchar(20)
);
```

```sql
alter table professors add column department_id integer;
alter table students add column advisor_id integer;
alter table courses add column department_id integer;

create table grade_scale(
    grade_id       serial primary key,
    letter_grade   char(2),
    min_percentage numeric(4,1),
    max_percentage numeric(4,1),
    gpa_points     numeric(3,2)
);

create table semester_calendar(
    semester_id           serial primary key,
    semester_name         varchar(20),
    academic_year         int,
    start_date            date,
    end_date              date,
    registration_deadline timestamp with time zone,
    is_current            bool
);
```

```
drop table if exists student_book_loans;
drop table if exists library_books;
drop table if exists grade_scale;

create table grade_scale(
    grade_id        serial primary key,
    letter_grade    char(2),
    min_percentage numeric(4,1),
    max_percentage numeric(4,1),
    gpa_points      numeric(3,2),
    description     text
);

drop table if exists semester_calendar cascade;

create table semester_calendar(
    semester_id             serial primary key,
    semester_name           varchar(20),
    academic_year           int,
    start_date              date,
    end_date                date,
    registration_deadline timestamp with time zone,
    is_current              bool
);
```

```
drop database if exists university_test;
drop database if exists university_distributed;
create database university_backup with template = university_main;
```