

--1.1  
  
  
CREATE DATABASE university\_main  
 TEMPLATE = template0  
 ENCODING 'UTF8';  
  
  
  
CREATE DATABASE = university\_archive  
 CONNECTION LIMIT 50  
 TEMPLATE template0;  
  
  
  
CREATE DATABASE university\_test  
 IS\_TEMPLATE true  
 CONNECTION LIMIT 10;

--1.2  
  
CREATE TABLESPACE student\_data LOCATION 'C:/temp/students';  
CREATE TABLESPACE course\_data LOCATION 'C:/temp/courses';  
  
  
CREATE DATABASE university\_distributed  
 TEMPLATE = template0  
 TABLESPACE = student\_data  
 ENCODING = 'UTF8'  
 LC\_CTYPE = 'ru\_RU.UTF8'  
 LC\_COLLATE = 'ru\_RU.UTF8';  
  
--windows не поддерживает латин поэтому пришлось так сделать

--2.1  
  
\c university\_main  
  
  
  
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 DECIMAL(3,2) DEFAULT 0.00,  
 is\_active BOOLEAN DEFAULT TRUE,  
 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,  
 salary DECIMAL(10,2),  
 is\_tenured BOOLEAN DEFAULT FALSE,  
 years\_experience INTEGER  
);  
  
  
CREATE TABLE courses (  
 course\_id SERIAL PRIMARY KEY,  
 course\_code CHAR(8) NOT NULL,  
 course\_title VARCHAR(100) NOT NULL,  
 description TEXT,  
 credits SMALLINT NOT NULL,  
 max\_enrollment INTEGER,  
 course\_fee DECIMAL(8,2),  
 is\_online BOOLEAN DEFAULT FALSE,  
 created\_at TIMESTAMP WITHOUT TIME ZONE DEFAULT *CURRENT\_TIMESTAMP*);

--2.2  
  
CREATE TABLE class\_schedule (  
 schedule\_id SERIAL PRIMARY KEY,  
 course\_id INTEGER,  
 professor\_id INTEGER,  
 classroom VARCHAR(20),  
 class\_date DATE,  
 start\_time TIME,  
 end\_time TIME,  
 duration INTERVAL  
);  
  
CREATE TABLE student\_records (  
 record\_id SERIAL PRIMARY KEY,  
 student\_id INTEGER,  
 course\_id INTEGER,  
 semester VARCHAR(20),  
 year INTEGER,  
 grade CHAR(2),  
 attendance\_percentage DECIMAL(3,1),  
 submission\_timestamp TIMESTAMPTZ,  
 last\_updated TIMESTAMPTZ  
);  
  
  
--3.1  
  
ALTER TABLE students  
 ADD COLUMN middle\_name VARCHAR(30),  
 ADD COLUMN student\_status VARCHAR(20) DEFAULT 'ACTIVE',  
 ALTER COLUMN phone TYPE VARCHAR(20),  
 ALTER COLUMN gpa SET DEFAULT 0.00;  
  
ALTER TABLE professors  
 ADD COLUMN department\_code CHAR(5),  
 ADD COLUMN research\_area TEXT,  
 ALTER COLUMN years\_experience TYPE SMALLINT,  
 ALTER COLUMN is\_tenured SET DEFAULT false,  
 ADD COLUMN last\_promotion\_date DATE;  
  
  
  
ALTER TABLE courses  
 ADD COLUMN prerequisite\_course\_id INTEGER,  
 ADD COLUMN difficulty\_level SMALLINT,  
 ALTER COLUMN course\_code TYPE VARCHAR(10),  
 ALTER COLUMN credits SET DEFAULT 3,  
 ADD COLUMN lab\_required BOOLEAN DEFAULT false;  
  
  
--3.2  
ALTER TABLE class\_schedule  
 ADD COLUMN room\_capacity INTEGER,  
 DROP COLUMN duration,  
 ADD COLUMN session\_type VARCHAR(15),  
 ALTER COLUMN classroom TYPE VARCHAR(30),  
 ADD COLUMN equipment\_needed TEXT;  
  
  
ALTER TABLE student\_records  
 ADD COLUMN extra\_credit\_points DECIMAL(3,1) DEFAULT 0.0,  
 ALTER COLUMN grade TYPE VARCHAR(5),  
 ADD COLUMN final\_exam\_date DATE,  
 DROP COLUMN last\_updated;  
  
  
--4.1  
CREATE TABLE departments (  
 department\_id SERIAL PRIMARY KEY,  
 department\_name VARCHAR(100),  
 department\_code CHAR(5),  
 building VARCHAR(50),  
 phone VARCHAR(15),  
 budget DECIMAL(15,2),  
 established\_year INTEGER  
);  
  
  
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 DECIMAL(8,2),  
 is\_available BOOLEAN,  
 acquisition\_timestamp TIMESTAMP  
);  
  
  
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 DECIMAL(8,2),  
 loan\_status VARCHAR(20)  
);  
  
  
--4.2  
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 DECIMAL(4,1),  
 max\_percentage DECIMAL(4,1),  
 gpa\_points DECIMAL(3,2)  
);  
  
CREATE TABLE semester\_calendar (  
 semester\_id SERIAL PRIMARY KEY,  
 semester\_name VARCHAR(20),  
 academic\_year INTEGER,  
 start\_date DATE,  
 end\_date DATE,  
 registration\_deadline TIMESTAMPTZ,  
 is\_current BOOLEAN  
);  
  
--5.1  
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 DECIMAL(4,1),  
 max\_percentage DECIMAL(4,1),  
 gpa\_points DECIMAL(3,2),  
 description TEXT  
);  
  
DROP TABLE semester\_calendar CASCADE;  
  
  
CREATE TABLE semester\_calendar (  
 semester\_id SERIAL PRIMARY KEY,  
 semester\_name VARCHAR(20),  
 academic\_year INTEGER,  
 start\_date DATE,  
 end\_date DATE,  
 registration\_deadline TIMESTAMPTZ,  
 is\_current BOOLEAN  
);  
  
  
--5.2  
DROP DATABASE IF EXISTS university\_test;  
DROP DATABASE IF EXISTS university\_distributed;  
CREATE DATABASE university\_backup TEMPLATE university\_main;