-- Table: Role

CREATE TABLE Role (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) UNIQUE NOT NULL,

description TEXT

);

-- Table: User

CREATE TABLE User (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255),

email VARCHAR(255) UNIQUE NOT NULL,

password\_hash VARCHAR(255) NOT NULL,

role\_id INT,

is\_active BOOLEAN DEFAULT TRUE,

created\_at DATETIME,

updated\_at DATETIME,

FOREIGN KEY (role\_id) REFERENCES Role(id)

);

-- Table: Employer

CREATE TABLE Employer (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

role\_id INT,

company\_name VARCHAR(255),

address VARCHAR(255),

website VARCHAR(255),

contact\_number VARCHAR(20),

logo VARCHAR(255),

is\_verified BOOLEAN DEFAULT FALSE,

created\_at DATETIME,

updated\_at DATETIME,

FOREIGN KEY (user\_id) REFERENCES User(id),

FOREIGN KEY (role\_id) REFERENCES Role(id)

);

-- Table: Candidates

CREATE TABLE Candidates (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

role\_id INT,

resume VARCHAR(255),

bio TEXT,

experience\_years INT,

education TEXT,

profile\_photo VARCHAR(255),

created\_at DATETIME,

updated\_at DATETIME,

FOREIGN KEY (user\_id) REFERENCES User(id),

FOREIGN KEY (role\_id) REFERENCES Role(id)

);

-- Table: Job\_Category

CREATE TABLE Job\_Category (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255),

description TEXT,

is\_active BOOLEAN

);

-- Table: Resumes

CREATE TABLE Resumes (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

full\_name VARCHAR(100) NOT NULL,

email VARCHAR(100) NOT NULL,

phone\_number VARCHAR(20),

summary TEXT,

skills TEXT,

experience TEXT,

education TEXT,

certifications TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES User(id)

);

-- Table: Job

CREATE TABLE Job (

id INT AUTO\_INCREMENT PRIMARY KEY,

employer\_id INT,

category\_id INT,

resume\_id INT,

title VARCHAR(255),

description TEXT,

job\_type VARCHAR(100),

salary\_min DECIMAL(10,2),

salary\_max DECIMAL(10,2),

posted\_date DATETIME,

application\_deadline DATETIME,

num\_vacancies INT,

status VARCHAR(50),

FOREIGN KEY (employer\_id) REFERENCES Employer(id),

FOREIGN KEY (category\_id) REFERENCES Job\_Category(id),

FOREIGN KEY (resume\_id) REFERENCES Resumes(id)

);

-- Table: Skill

CREATE TABLE Skill (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) UNIQUE NOT NULL

);

-- Table: Job\_Skill (Many-to-Many)

CREATE TABLE Job\_Skill (

job\_id INT,

skill\_id INT,

PRIMARY KEY (job\_id, skill\_id),

FOREIGN KEY (job\_id) REFERENCES Job(id),

FOREIGN KEY (skill\_id) REFERENCES Skill(id)

);

-- Table: Seeker\_Skill (Many-to-Many)

CREATE TABLE Seeker\_Skill (

seeker\_id INT,

skill\_id INT,

PRIMARY KEY (seeker\_id, skill\_id),

FOREIGN KEY (seeker\_id) REFERENCES Candidates(id),

FOREIGN KEY (skill\_id) REFERENCES Skill(id)

);

-- Table: Application

CREATE TABLE Application (

id INT AUTO\_INCREMENT PRIMARY KEY,

job\_id INT,

seeker\_id INT,

applied\_date DATETIME,

status VARCHAR(50),

resume\_submitted VARCHAR(255),

cover\_letter TEXT,

created\_at DATETIME,

updated\_at DATETIME,

FOREIGN KEY (job\_id) REFERENCES Job(id),

FOREIGN KEY (seeker\_id) REFERENCES Candidates(id)

);

-- Table: Interview

CREATE TABLE Interview (

id INT AUTO\_INCREMENT PRIMARY KEY,

application\_id INT,

scheduled\_date DATETIME,

mode VARCHAR(50),

interviewer\_details TEXT,

result VARCHAR(50),

notes TEXT,

FOREIGN KEY (application\_id) REFERENCES Application(id)

);

-- Table: Employer\_Package

CREATE TABLE Employer\_Package (

id INT AUTO\_INCREMENT PRIMARY KEY,

employer\_id INT,

name VARCHAR(255),

price DECIMAL(10,2),

duration\_days INT,

features TEXT,

start\_date DATETIME,

end\_date DATETIME,

status VARCHAR(50),

FOREIGN KEY (employer\_id) REFERENCES Employer(id)

);

-- Table: Notification

CREATE TABLE Notification (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

type VARCHAR(50),

message TEXT,

is\_read BOOLEAN DEFAULT FALSE,

created\_at DATETIME,

FOREIGN KEY (user\_id) REFERENCES User(id)

);