// Подключаем public/ и шаблонизатор

app.use(express.static(path.join(\_\_dirname, 'public')));

app.use(bodyParser.urlencoded({ extended: true }));

app.set('view engine', 'hbs');

app.set('views', path.join(\_\_dirname, 'demo-drive', 'views'));

app.use('/bootstrap', express.static(path.join(\_\_dirname, 'node\_modules', 'bootstrap', 'dist')));

// Регистрируем хелпер сравнения

hbs.registerHelper('ifEquals', function (arg1, arg2, options) {

  return arg1 === arg2 ? options.fn(this) : options.inverse(this);

});

/\* aboba \*/

app.use(session({

  secret: 'testdrive-secret',

  resave: false,

  saveUninitialized: true

}));

// Маршруты

app.get('/', (req, res) => res.redirect('/login'));

app.get('/register', (req, res) => res.render('register'));

app.post('/register', async (req, res) => {

  const { full\_name, phone, email, login, password } = req.body;

  const hash = await bcrypt.hash(password, 10);

  try {

    await pool.query('INSERT INTO users (full\_name, phone, email, login, password) VALUES ($1, $2, $3, $4, $5)',

      [full\_name, phone, email, login, hash]);

    res.redirect('/login');

  } catch (err) {

    res.send('Ошибка при регистрации: ' + err.message);

  }

});

app.get('/login', (req, res) => res.render('login'));

app.post('/login', async (req, res) => {

  const { login, password } = req.body;

  // Проверка администратора

  const adminResult = await pool.query('SELECT \* FROM admins WHERE login = $1', [login]);

  const admin = adminResult.rows[0];

  if (admin && password === admin.password) {

    req.session.admin = true;

    return res.redirect('/admin');

  }

  // Проверка пользователя

  const result = await pool.query('SELECT \* FROM users WHERE login = $1', [login]);

  const user = result.rows[0];

  if (user && await bcrypt.compare(password, user.password)) {

    req.session.user = user;

    res.redirect('/dashboard');

  } else {

    res.send('Неверный логин или пароль');

  }

});

app.get('/dashboard', async (req, res) => {

  if (!req.session.user) return res.redirect('/login');

  const apps = await pool.query(`

    SELECT tdr.\*, c.brand AS car\_brand, c.model AS car\_model

    FROM test\_drive\_requests tdr

    JOIN cars c ON tdr.car\_id = c.id

    WHERE tdr.user\_id = $1

    ORDER BY tdr.created\_at DESC

  `, [req.session.user.id]);

  res.render('dashboard', { apps: apps.rows });

});

app.get('/form', async (req, res) => {

  if (!req.session.user) return res.redirect('/login');

  const cars = await pool.query('SELECT \* FROM cars ORDER BY brand, model');

  res.render('form', { cars: cars.rows });

});

app.post('/form', async (req, res) => {

  const {

    address, phone, license\_series, license\_number, license\_date,

    car\_id, desired\_date, desired\_time, payment\_type

  } = req.body;

  await pool.query(`

    INSERT INTO test\_drive\_requests (

      user\_id, car\_id, address, contact\_phone, license\_series, license\_number, license\_issue\_date,

      desired\_date, desired\_time, payment\_type

    ) VALUES ($1,$2,$3,$4,$5,$6,$7,$8,$9,$10)`,

    [req.session.user.id, car\_id, address, phone, license\_series, license\_number, license\_date,

      desired\_date, desired\_time, payment\_type]);

  res.redirect('/dashboard');

});

app.get('/admin', async (req, res) => {

  if (!req.session.admin) return res.send('Доступ запрещен');

  const apps = await pool.query(`

    SELECT tdr.\*, u.full\_name, u.phone AS user\_phone, u.email, c.brand AS car\_brand, c.model AS car\_model

    FROM test\_drive\_requests tdr

    JOIN users u ON tdr.user\_id = u.id

    JOIN cars c ON tdr.car\_id = c.id

    ORDER BY tdr.created\_at DESC

  `);

  res.render('admin', { apps: apps.rows });

});

app.post('/admin/update', async (req, res) => {

  const { id, status, reason } = req.body;

  await pool.query('UPDATE test\_drive\_requests SET status = $1, rejection\_reason = $2 WHERE id = $3', [status, reason, id]);

  res.redirect('/admin');

});

app.get('/logout', (req, res) => {

  req.session.destroy(err => {

    if (err) {

      return res.send('Ошибка при выходе');

    }

    res.redirect('/login');

  });

});