|  |
| --- |
| Entity |
| // @Entity  @Data  public class KhachHang      implements EntityInterface<KhachHang, Long>  {      private Long userId;      @Override      public void updateNotId(KhachHang o2) {          // không làm gì cả      }      public KhachHang() {      }      public KhachHang(Long userId) {          this.userId = userId;      }      } |

|  |
| --- |
| RowMapper |
| public class KhachHangRowMapper      implements RowMapper<KhachHang>  {      @Override      @Nullable      public KhachHang mapRow(ResultSet ketQuaTraVe, int sttHang) throws SQLException {          Long userId = ketQuaTraVe.getLong("Userid");          KhachHang kh1 = new KhachHang(userId);          return kh1;      }    } |

|  |
| --- |
| KhachHangRepo |
| @Repository  public class KhachHangRepo      implements RepoInterface<KhachHang, Long>  {      @Autowired      private JdbcTemplate jdbcTemplate;      private KhachHangRowMapper khRowMapper = new KhachHangRowMapper();      @Override      public KhachHang findById(Long tKey) {          String sql =  "SELECT \* FROM KhachHang WHERE Userid= ? ";          PreparedStatementCreator psc = con -> {              PreparedStatement ps = con.prepareStatement(sql);              ps.setLong(1, tKey);              return ps;          };          return this.jdbcTemplate.query(psc, this.khRowMapper).stream().findFirst().orElse(null);      }      @Override      public List<KhachHang> getList(Integer sttPage, Integer sizePage) {          String sql =              "SELECT "              + " DISTINCT "              + " \* "              + " FROM KhachHang "              + " WHERE 1=1 "              + " ORDER BY Userid DESC "              + " OFFSET ? ROW "              + " FETCH NEXT ? ROW ONLY ";          PreparedStatementCreator psc = con-> {              PreparedStatement ps = con.prepareStatement(sql);              ps.setInt(1, ( (sizePage-1)\*sttPage) );              ps.setInt(2, sizePage);              return ps;          };          return this.jdbcTemplate.query(psc, this.khRowMapper);      }      @Override      public KhachHang create(KhachHang t) {          // PK của KhachHang là FK nên không được set null PK          String sql =              "INSERT INTO KhachHang "              + " (Userid) "              + " VALUES (?) ";          // PK của KhachHang là FK nên không cần get Key trả về          PreparedStatementCreator psc = con-> {              PreparedStatement ps = con.prepareStatement(sql);              // điền tham số vào dấu ?              ps.setLong(1, t.getUserId());              return ps;          };          jdbcTemplate.update(psc);          return this.findById(t.getUserId());      }      @Override      public KhachHang update(Long tKey, KhachHang t) {          t.setUserId(tKey);          String sql =              " UPDATE KhachHang "              + " SET "                  + " "              + " WHERE "                  + " Userid = ? ";          PreparedStatementCreator psc = con->{              PreparedStatement ps = con.prepareStatement(sql);              ps.setLong(1, t.getUserId());              return ps;          };          jdbcTemplate.update(psc);          return this.findById(t.getUserId());      }      @Override      public Boolean delete(Long tKey) {          String sql =              " DELETE FROM KhachHang "              + " WHERE Userid = ? ";          PreparedStatementCreator psc = con->{              PreparedStatement ps = con.prepareStatement(sql);              ps.setLong(1, tKey);              return ps;          };          jdbcTemplate.update(psc);          return true;      }    } |

|  |
| --- |
| KhachHangService |
| @Service  public class KhachHangService      implements ServiceInterface<KhachHang, Long>  {      @Autowired      private KhachHangRepo khRepo;      @Override      public KhachHang findById(Long tKey) {          return this.khRepo.findById(tKey);      }      @Override      public List<KhachHang> getList(Integer sttPage, Integer sizePage) {          return this.khRepo.getList(sttPage, sizePage);      }      @Override      public KhachHang create(KhachHang t) {          // PK của khachhang là FK nên không được set null          // KhachHang kh1 = this.findById(t.getUserId());          // // nếu đã tồn tại thì không create          // if(kh1 != null){          //     return null;          // }          return this.khRepo.create(t);      }      @Override      public KhachHang update(Long tKey, KhachHang t) {          KhachHang kh1 = this.findById(tKey);          // nếu không tồn tại thì không update          if(kh1 == null){              return null;          }          return this.khRepo.update(tKey, t);      }      @Override      public Boolean delete(Long tKey) {          return this.khRepo.delete(tKey);      }    } |