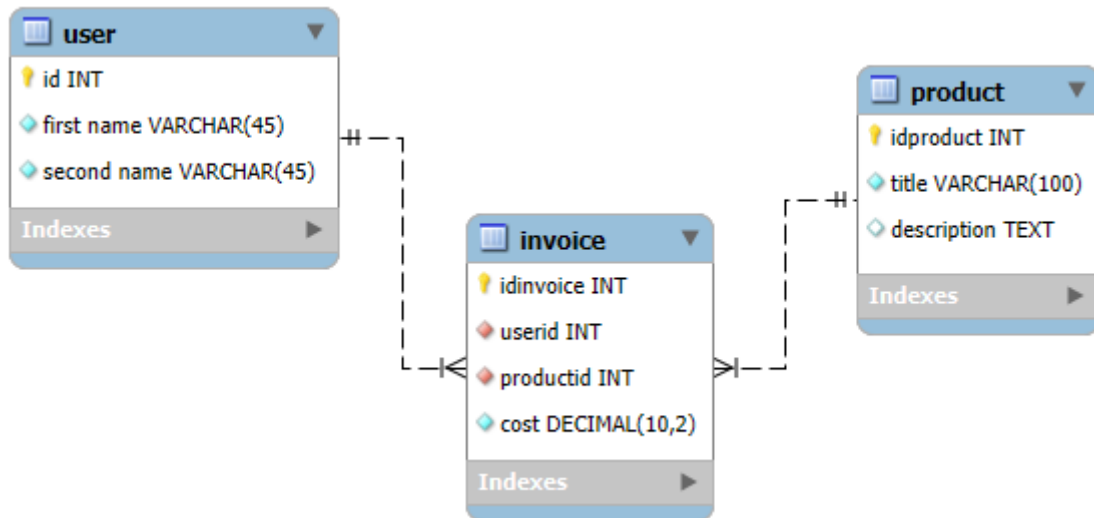


## Отчет Лаб 2

### Задание 1.



Ссылка на sql файл.

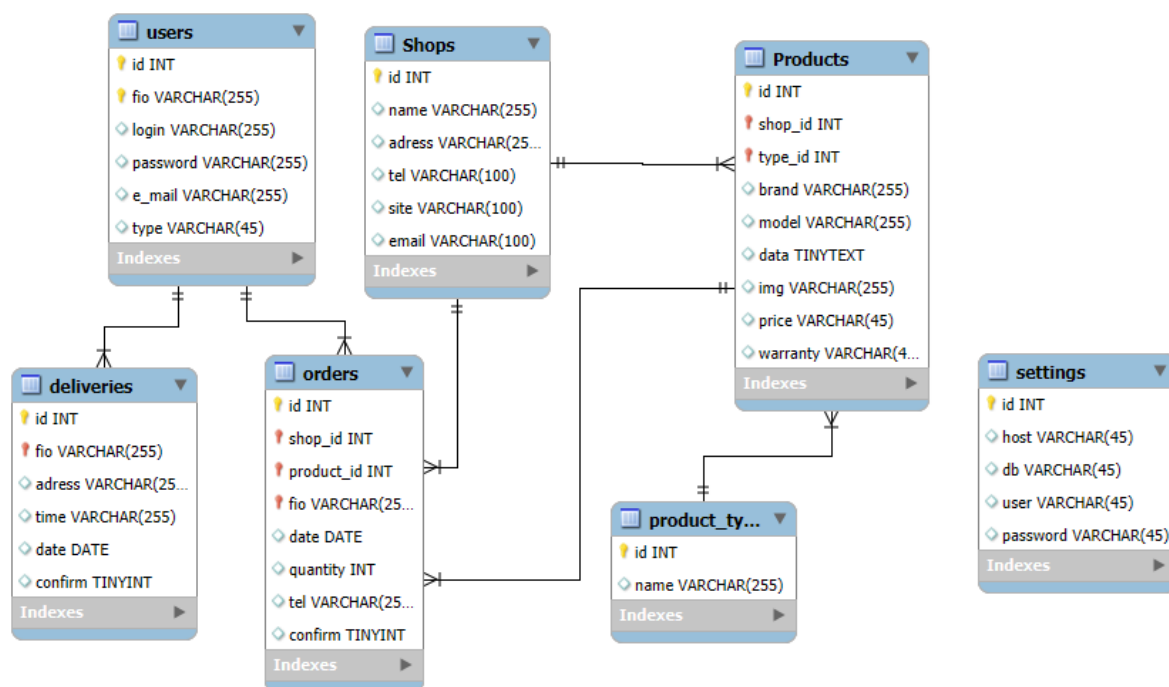
Скрипт таблицы инвойс.

-----  
-- Table `firstModel`.`invoice`  
-----

```
CREATE TABLE IF NOT EXISTS `firstModel`.`invoice` (  
  `idinvoice` INT NOT NULL AUTO_INCREMENT,  
  `userid` INT NOT NULL,  
  `productid` INT NOT NULL,  
  `cost` DECIMAL(10,2) NOT NULL,  
  PRIMARY KEY (`idinvoice`),  
  INDEX `user_idx` (`userid` ASC) VISIBLE,  
  INDEX `product_idx` (`productid` ASC) VISIBLE,  
  CONSTRAINT `user`  
    FOREIGN KEY (`userid`)  
      REFERENCES `firstModel`.`user` (`id`)  
      ON DELETE CASCADE  
      ON UPDATE CASCADE,  
  CONSTRAINT `product`  
    FOREIGN KEY (`productid`)  
      REFERENCES `firstModel`.`product` (`idproduct`)  
      ON DELETE CASCADE  
      ON UPDATE CASCADE)
```

ENGINE = InnoDB;

### Задание 2.



Ссылка на sql файл

Скипт таблицы ордерс

-----  
 -- Table `market`.`orders`  
 -----

```

CREATE TABLE IF NOT EXISTS `market`.`orders` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `shop_id` INT NOT NULL,
  `product_id` INT NOT NULL,
  `fio` VARCHAR(255) NOT NULL,
  `date` DATE NULL,
  `quantity` INT NULL,
  `tel` VARCHAR(255) NULL,
  `confirm` TINYINT NULL,
  PRIMARY KEY (`id`, `shop_id`, `product_id`, `fio`),
  UNIQUE INDEX `id_UNIQUE` (`id` ASC) VISIBLE,
  INDEX `fio_id_idx` (`fio` ASC) VISIBLE,
  INDEX `product_id_idx` (`product_id` ASC) VISIBLE,
  INDEX `shop_id_idx` (`shop_id` ASC) VISIBLE,
  CONSTRAINT `fio_id`
    FOREIGN KEY (`fio`)
      REFERENCES `market`.`users` (`fio`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `product_id`
    FOREIGN KEY (`product_id`)
      REFERENCES `market`.`Products` (`id`)
      ON DELETE NO ACTION
  
```

```
ON UPDATE NO ACTION,  
CONSTRAINT `shop_id`  
FOREIGN KEY (`shop_id`)  
REFERENCES `market`.`Shops` (`id`)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Задание 3.

#### Forward Engineering Progress

The following tasks will now be executed. Please monitor the execution.  
Press Show Logs to see the execution logs.

- ☒ Connect to DBMS
- ☒ Execute Forward Engineered Script
- ☒ Read Back Changes Made by Server
- ☒ Save Synchronization State

Forward Engineer Finished Successfully

Задание 4.

Как и в предыдущей лабораторной работе по удалению данных, связанных с данными в другой таблице, таким образом что при удалении связанные данные каскадно удаляются или обновляются связанные данные были каскадно удалены.