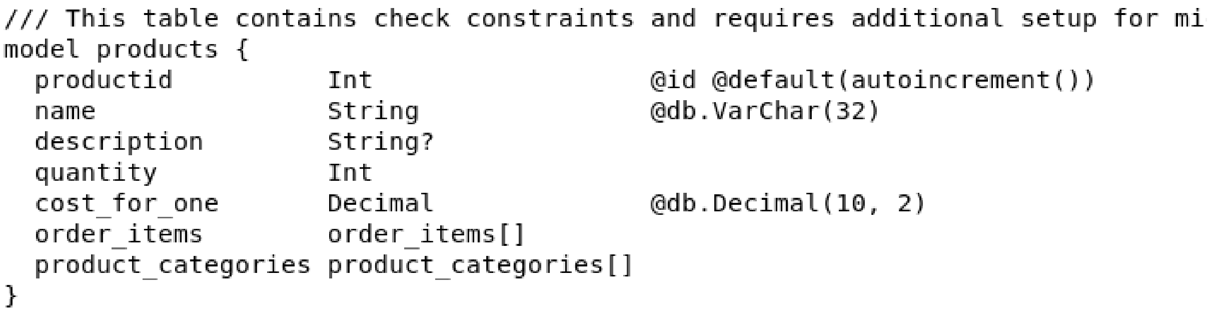
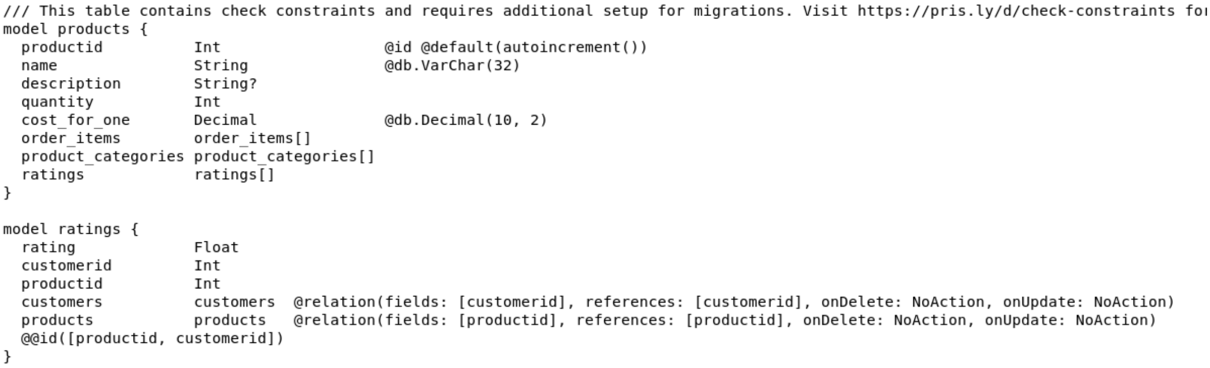
add-ratings-table migration: Додаю нову таблицю рейтингу, для якої визначаю колонку rating, для зберігання рейтингу, та два foreign keys: customers, products.

screenshot before:



```
/// This table contains check constraints and requires additional setup for mi
model products {
  productid          Int                  @id @default(autoincrement())
  name               String               @db.VarChar(32)
  description        String?
  quantity           Int
  cost_for_one       Decimal              @db.Decimal(10, 2)
  order_items        order_items[]
  product_categories product_categories[]
}
```

screenshot after:

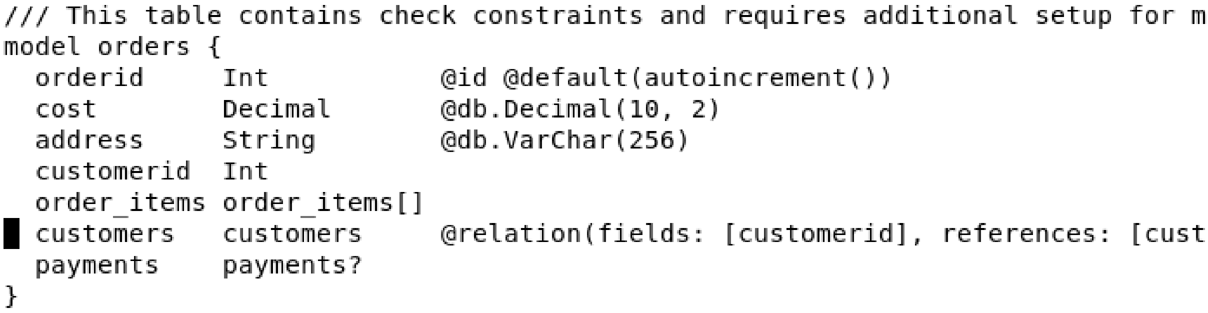


```
/// This table contains check constraints and requires additional setup for migrations. Visit https://pris.ly/d/check-constraints fo
model products {
  productid          Int                  @id @default(autoincrement())
  name               String               @db.VarChar(32)
  description        String?
  quantity           Int
  cost_for_one       Decimal              @db.Decimal(10, 2)
  order_items        order_items[]
  product_categories product_categories[]
  ratings            ratings[]
}

model ratings {
  rating             Float
  customerid         Int
  productid          Int
  customers          customers  @relation(fields: [customerid], references: [customerid], onDelete: NoAction, onUpdate: NoAction)
  products           products   @relation(fields: [productid], references: [productid], onDelete: NoAction, onUpdate: NoAction)
  @@id([productid, customerid])
}
```

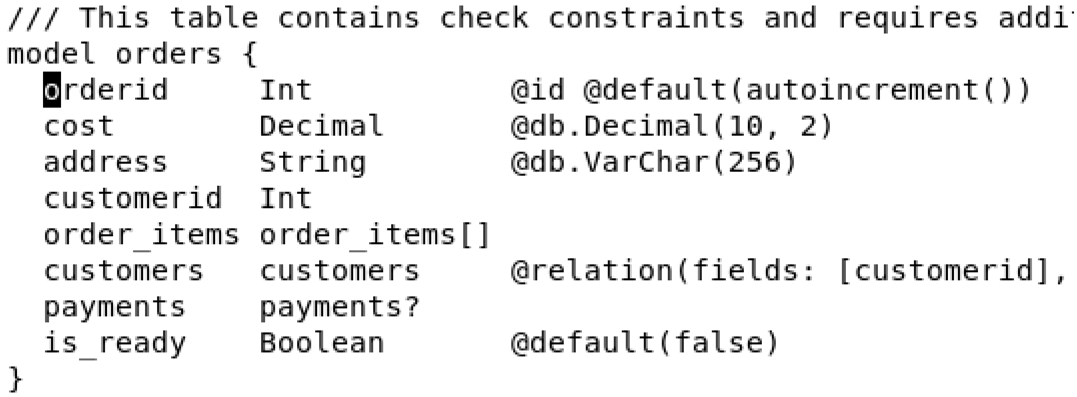
add-orders-column: Додаю нову колонку до таблиці orders - is_ready, яка визначає чи готове замовлення.

screenshot before:



```
/// This table contains check constraints and requires additional setup for m
model orders {
  orderid     Int           @id @default(autoincrement())
  cost        Decimal       @db.Decimal(10, 2)
  address     String        @db.VarChar(256)
  customerid  Int
  order_items order_items[]
  customers   customers     @relation(fields: [customerid], references: [cust
  payments    payments?
}
```

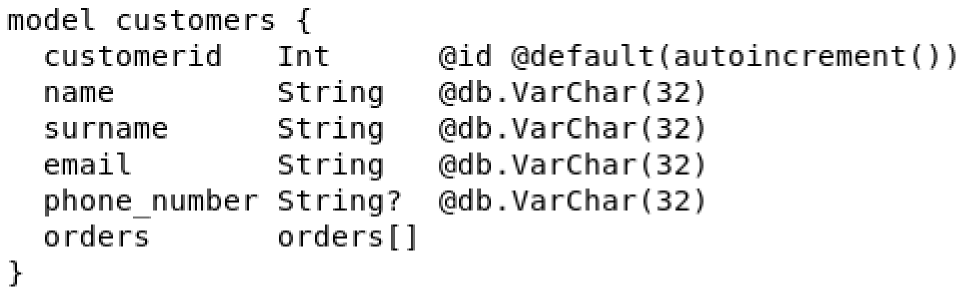
screenshot after:



```
/// This table contains check constraints and requires addi
model orders {
  orderid     Int           @id @default(autoincrement())
  cost        Decimal       @db.Decimal(10, 2)
  address     String        @db.VarChar(256)
  customerid  Int
  order_items order_items[]
  customers   customers     @relation(fields: [customerid],
  payments    payments?
  is_ready    Boolean       @default(false)
}
```

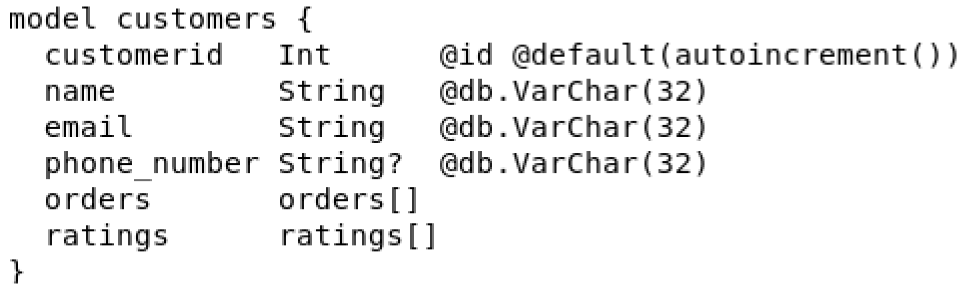
remove-customers-column: Видаляю колонку surname для customer. Для ім'я тепер використовується одна колонка name.

screenshot before:



```
model customers {
  customerid   Int      @id @default(autoincrement())
  name         String   @db.VarChar(32)
  surname      String   @db.VarChar(32)
  email        String   @db.VarChar(32)
  phone_number String?  @db.VarChar(32)
  orders       orders[]
}
```

screenshot after:



```
model customers {
  customerid   Int      @id @default(autoincrement())
  name         String   @db.VarChar(32)
  email        String   @db.VarChar(32)
  phone_number String?  @db.VarChar(32)
  orders       orders[]
  ratings      ratings[]
}
```