МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №7

по дисциплине «Базы данных»

тема: «Создание графического интерфейса для взаимодействия с БД c помощью ORM»

Выполнил: ст. группы ВТ-32

Воскобойников И. С.

Проверил: Панченко М.В.

Белгород 2020 г.

**Цель работы**: научиться работать с базой данных при помощи ORM

**Задание**

Написать графический интерфейс, позволяющий работать с базой данных с использованием ORM

Код программы

models.py

**from** peewee **import** \*  
  
db = SqliteDatabase(**'BD.db'**)  
  
  
**class** BaseModel(Model):  
 **class** Meta:  
 database = db  
  
  
**class** Country(BaseModel):  
 county = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'county'**]  
  
  
**class** City(BaseModel):  
 city = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'city'**]  
  
  
**class** Street(BaseModel):  
 street = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'street'**]  
  
  
**class** Building(BaseModel):  
 building = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'building'**]  
  
  
**class** Address(BaseModel):  
 country = ForeignKeyField(Country, related\_name=**'address'**, on\_delete=**'CASCADE'**)  
 city = ForeignKeyField(City, related\_name=**'address'**, on\_delete=**'CASCADE'**)  
 street = ForeignKeyField(Street, related\_name=**'address'**, on\_delete=**'CASCADE'**)  
 building = ForeignKeyField(Building, related\_name=**'address'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'country'**, **'city'**, **'street'**, **'building'**]  
  
  
**class** NameProduct(BaseModel):  
 name = CharField(max\_length=250)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'neme'**]  
  
  
**class** Product(BaseModel):  
 name = ForeignKeyField(NameProduct, related\_name=**'product'**, on\_delete=**'CASCADE'**)  
 address = ForeignKeyField(Address, related\_name=**'product'**, on\_delete=**'CASCADE'**)  
 size = IntegerField()  
 weight = IntegerField()  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'name'**, **'address'**, **'size'**, **'weight'**]  
  
  
**class** Storage(BaseModel):  
 product = ForeignKeyField(Product, related\_name=**'storage'**, on\_delete=**'CASCADE'**)  
 address = ForeignKeyField(Address, related\_name=**'storage'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'product'**, **'address '**]  
  
  
**class** ConsumerContactDetails(BaseModel):  
 name = CharField(max\_length=50)  
 description = CharField(max\_length=50)  
 email = CharField(max\_length=50)  
 phone = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'name'**, **'description'**, **'email'**, **'phone'**]  
  
  
**class** TypeDelivery(BaseModel):  
 type = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'type'**]  
  
  
**class** DeliveryOrder(BaseModel):  
 product = ForeignKeyField(Product, related\_name=**'delivery\_order'**, on\_delete=**'CASCADE'**)  
 price = IntegerField()  
 start\_date = DateField()  
 end\_date = DateField()  
 type\_delivery = ForeignKeyField(TypeDelivery, related\_name=**'delivery\_order'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'product'**, **'price'**, **'start\_date'**, **'end\_date'**, **'type\_delivery'**]  
  
  
**class** Consumer(BaseModel):  
 delivery\_order = ForeignKeyField(DeliveryOrder, related\_name=**'consumer'**, on\_delete=**'CASCADE'**)  
 address = ForeignKeyField(Address, related\_name=**'consumer'**, on\_delete=**'CASCADE'**)  
 contact\_details = ForeignKeyField(ConsumerContactDetails, related\_name=**'consumer'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'delivery\_order'**, **'address'**, **'contact\_details'**]  
  
  
**class** ProviderContactDetails(BaseModel):  
 name = CharField(max\_length=50)  
 description = CharField(max\_length=50)  
 email = CharField(max\_length=50)  
 phone = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'name'**, **'description'**, **'email'**, **'phone'**]  
  
  
**class** Provider(BaseModel):  
 product = ForeignKeyField(Product, related\_name=**'provider'**, on\_delete=**'CASCADE'**)  
 address = ForeignKeyField(Address, related\_name=**'provider'**, on\_delete=**'CASCADE'**)  
 contact\_details = ForeignKeyField(ProviderContactDetails, related\_name=**'provider'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'product'**, **'address'**, **'contact\_details'**]  
  
  
**class** TypeTransport(BaseModel):  
 type = IntegerField()  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'type'**]  
  
  
**class** TypeSizeTransport(BaseModel):  
 size = IntegerField()  
 type = ForeignKeyField(TypeTransport, related\_name=**'type\_size\_transport'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'size'**, **'type'**]  
  
**class** ModelTransport(BaseModel):  
 model = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'model'**]  
  
**class** Transport(BaseModel):  
 type\_size = ForeignKeyField(TypeSizeTransport, related\_name=**'transport'**, on\_delete=**'CASCADE'**)  
 model = ForeignKeyField(ModelTransport, related\_name=**'transport'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'type\_size'**, **'model'**]  
  
  
**class** CourierContactDetails(BaseModel):  
 name = CharField(max\_length=50)  
 email = CharField(max\_length=50)  
 phone = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'name'**, **'email'**, **'phone'**]  
  
  
**class** Courier(BaseModel):  
 transport = ForeignKeyField(Transport, related\_name=**'courier'**, on\_delete=**'CASCADE'**)  
 location = DoubleField()  
 delivery\_orders = ForeignKeyField(DeliveryOrder, related\_name=**'courier'**, on\_delete=**'CASCADE'**)  
 contact\_details = ForeignKeyField(CourierContactDetails, related\_name=**'courier'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'transport'**, **'location'**, **'delivery\_orders'**, **'contact\_details'**]  
  
  
**class** TransportCompanyContactDetails(BaseModel):  
 name = CharField(max\_length=50)  
 description = CharField(max\_length=50)  
 email = CharField(max\_length=50)  
 phone = CharField(max\_length=50)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'name'**, **'description'**, **'email'**, **'phone'**]  
  
  
**class** TransportCompany(BaseModel):  
 courier = ForeignKeyField(Courier, related\_name=**'transport\_company'**, on\_delete=**'CASCADE'**)  
 delivery\_order = ForeignKeyField(DeliveryOrder, related\_name=**'transport\_company'**, on\_delete=**'CASCADE'**)  
 address = ForeignKeyField(Address, related\_name=**'transport\_company'**, on\_delete=**'CASCADE'**)  
 contact\_details = ForeignKeyField(TransportCompanyContactDetails, related\_name=**'transport\_company'**, on\_delete=**'CASCADE'**)  
  
 @staticmethod  
 **def** get\_names():  
 **return** [**'id'**, **'courier'**, **'delivery\_order'**, **'address'**, **'contact\_details'**]  
  
  
**for** m **in** [Country, City, Street, Building, Address, NameProduct, Product, Storage,  
 ConsumerContactDetails, TypeDelivery, DeliveryOrder, Consumer, ProviderContactDetails,  
 Provider, TypeTransport, TypeSizeTransport, ModelTransport, Transport,  
 CourierContactDetails, Courier, TransportCompanyContactDetails, TransportCompany]:  
 **if not** m.table\_exists():  
 m.create\_table()

ORM\_lab\_7.py

**from** tkinter **import** \*  
**from** tkinter **import** ttk  
**from** models **import** \*  
  
  
**class** BD: *# Класс "БД"  
 # Исполнение запроса* @staticmethod  
 **def** print\_table(model):  
 header = model.get\_names()  
 body = model.select()  
 max\_len = max([len(x) **for** x **in** header])  
 **for** i **in** body:  
 **for** j **in** i:  
 **if** len(str(j)) > max\_len:  
 max\_len = len(str(j))  
 max\_len += 2  
 **for** i **in** header:  
 print(**"|"**, (**"{:^"** + str(max\_len) + **"s}"**).format(i), **"|"**, end=**""**)  
 print(**"\n| "**, **"-"** \* (max\_len + 2) \* len(header), **" |"**)  
 **for** i **in** body:  
 **for** j **in** i:  
 print(**"|"**, (**"{:^"** + str(max\_len) + **"s}"**).format(str(j)), **"|"**, end=**""**)  
 print()  
  
 @staticmethod  
 **def** str\_table(model):  
 res = []  
 header = model.get\_names()  
 print(header)  
 body = model.select().dicts()  
 max\_len = 0 *# Найдем максимальную длину поля таблицы в строковом представлении* **for** i **in** header:  
 **if** len(i) > max\_len:  
 max\_len = len(i)  
 **for** i **in** body:  
 **for** j **in** i.values():  
 **if** len(str(j)) > max\_len:  
 max\_len = len(str(j))  
 max\_len += 2  
 bufStr = **""  
 for** i **in** header: *# Выведем таблицу* bufStr += **"|"** + (**"{:^"** + str(max\_len) + **"s}"**).format(i) + **"|"** res.append(bufStr)  
 res.append(**"| "** + **"-"** \* (max\_len + 2) \* len(header) + **" |"**)  
 **for** i **in** body:  
 bufStr = **""  
 for** j **in** i.values():  
 bufStr += **"|"** + (**"{:^"** + str(max\_len) + **"s}"**).format(str(j)) + **"|"** res.append(bufStr)  
 **return '\n'**.join(res)  
  
  
*# курьер***def** insert\_courier\_contact\_details():  
 CourierContactDetails.create(  
 id=text\_courier\_contact\_details\_id.get(),  
 name=text\_courier\_contact\_details\_fio.get(),  
 phone=text\_courier\_contact\_details\_phone.get(),  
 email=text\_courier\_contact\_details\_email.get()  
 )  
  
  
**def** output\_courier\_contact\_details():  
 textOutputCourier.delete(1.0, END)  
 textOutputCourier.insert(1.0, BD.str\_table(CourierContactDetails))  
  
**def** insert\_courier():  
 Courier.create(  
 id=text\_courier\_id.get(),  
 transport=text\_courier\_id\_transport.get(),  
 location=text\_courier\_location.get(),  
 delivery\_order=text\_courier\_id\_delivery\_order.get(),  
 contact\_details=text\_courier\_id\_contact\_details.get()  
 )  
  
**def** output\_courier():  
 textOutputCourier.delete(1.0, END)  
 textOutputCourier.insert(1.0, BD.str\_table(Courier))  
  
**def** insert\_type\_transport():  
 TypeTransport.create(  
 id=text\_type\_transport\_id.get(),  
 type=text\_type\_transport\_type.get(),  
 )  
  
**def** output\_type\_transport():  
 textOutputTransport.delete(1.0, END)  
 textOutputTransport.insert(1.0, BD.str\_table(TypeTransport))  
  
**def** insert\_size\_transport():  
 TypeSizeTransport(  
 id=text\_size\_transport\_id.get(),  
 size=text\_size\_transport\_size.get(),  
 type=text\_size\_transport\_type.get()  
 )  
  
**def** output\_size\_transport():  
 textOutputTransport.delete(1.0, END)  
 textOutputTransport.insert(1.0, BD.str\_table(TypeSizeTransport))  
  
**def** insert\_model\_transport():  
 ModelTransport.create(  
 id=text\_model\_transport\_id.get(),  
 model=text\_model\_transport\_model.get()  
 )  
  
**def** output\_model\_transport():  
 textOutputTransport.delete(1.0, END)  
 textOutputTransport.insert(1.0, BD.str\_table(ModelTransport))  
  
**def** insert\_transport():  
 Transport.create(  
 id=text\_transport\_id.get(),  
 type\_size=text\_transport\_type\_size.get(),  
 model=text\_transport\_model.get()  
 )  
  
**def** output\_transport():  
 textOutputTransport.delete(1.0, END)  
 textOutputTransport.insert(1.0, BD.str\_table(Transport))  
  
**def** insert\_transport\_company\_contact():  
 TransportCompanyContactDetails.create(  
 id=text\_transport\_company\_contact\_id.get(),  
 name=text\_transport\_company\_contact\_name.get(),  
 description=text\_transport\_company\_contact\_dscription.get(),  
 email=text\_transport\_company\_contact\_email.get(),  
 phone=text\_transport\_company\_contact\_phone.get()  
 )  
  
**def** output\_transport\_company\_contact():  
 textOutputTransportCompany.delete(1.0, END)  
 textOutputTransportCompany.insert(1.0, BD.str\_table(TransportCompanyContactDetails))  
  
**def** insert\_transport\_company():  
 TransportCompany.create(  
 id=text\_transport\_company\_id.get(),  
 courier=text\_transport\_company\_id\_courier.get(),  
 delivery\_order=text\_transport\_company\_id\_order.get(),  
 adress=text\_transport\_company\_id\_adress.get(),  
 contact\_details=text\_transport\_company\_id\_contact\_details.get()  
 )  
  
**def** output\_transport\_company():  
 textOutputTransportCompany.delete(1.0, END)  
 textOutputTransportCompany.insert(1.0, BD.str\_table(TransportCompany))  
  
**def** insert\_delivery\_order\_type():  
 TypeDelivery.create(  
 id=text\_delivery\_order\_type\_id.get(),  
 type=text\_delivery\_order\_type\_type.get()  
 )  
  
**def** output\_delivery\_order\_type():  
 textOutputDeliveryOrder.delete(1.0, END)  
 textOutputDeliveryOrder.insert(1.0, BD.str\_table(TypeDelivery))  
  
**def** insert\_delivery\_order():  
 DeliveryOrder.create(  
 id=text\_delivery\_order\_id.get(),  
 product=text\_delivery\_order\_id\_product.get(),  
 price=text\_delivery\_order\_price.get(),  
 start\_date=text\_delivery\_order\_start\_date.get(),  
 end\_date=text\_delivery\_order\_end\_date.get(),  
 type\_delivery=text\_delivery\_order\_type\_delivery.get()  
 )  
  
**def** output\_delivery\_order():  
 textOutputDeliveryOrder.delete(1.0, END)  
 textOutputDeliveryOrder.insert(1.0, BD.str\_table(DeliveryOrder))  
  
**def** insert\_provider\_contact():  
 ProviderContactDetails.create(  
 id=text\_provider\_contact\_details\_id.get(),  
 name=text\_provider\_contact\_details\_name.get(),  
 description=text\_provider\_contact\_details\_dscription.get(),  
 email=text\_provider\_contact\_details\_email.get(),  
 phone=text\_provider\_contact\_details\_phone.get()  
 )  
  
**def** output\_provider\_contact():  
 textOutputProvider.delete(1.0, END)  
 textOutputProvider.insert(1.0, BD.str\_table(ProviderContactDetails))  
  
**def** insert\_provider():  
 Provider.create(  
 id=text\_provider\_id.get(),  
 product=text\_provider\_id\_product.get(),  
 address=text\_provider\_id\_adress.get(),  
 contact\_details=text\_provider\_id\_contact\_details.get()  
 )  
  
**def** output\_provider():  
 textOutputProvider.delete(1.0, END)  
 textOutputProvider.insert(1.0, BD.str\_table(Provider))  
  
**def** insert\_name\_product():  
 NameProduct.create(  
 id=text\_name\_product\_id.get(),  
 name=text\_name\_product\_name.get()  
 )  
  
**def** output\_name\_product():  
 textOutputProduct.delete(1.0, END)  
 textOutputProduct.insert(1.0, BD.str\_table(NameProduct))  
  
**def** insert\_product():  
 Product.create(  
 id=text\_product\_id.get(),  
 size=text\_product\_size.get(),  
 weight=text\_product\_weight.get(),  
 name=text\_product\_id\_name.get(),  
 address=text\_product\_id\_adress.get()  
 )  
  
**def** output\_product():  
 textOutputProduct.delete(1.0, END)  
 textOutputProduct.insert(1.0, BD.str\_table(Product))  
  
**def** insert\_consumer\_contact():  
 ConsumerContactDetails.create(  
 id=text\_consumer\_contact\_id.get(),  
 name=text\_consumer\_contact\_name.get(),  
 description=text\_consumer\_contact\_dscription.get(),  
 email=text\_consumer\_contact\_email.get(),  
 phone=text\_consumer\_contact\_phone.get()  
 )  
**def** output\_consumer\_contact():  
 textOutputСonsumer.delete(1.0, END)  
 textOutputСonsumer.insert(1.0, BD.str\_table(CourierContactDetails))  
  
**def** insert\_consumer():  
 Consumer.create(  
 id=text\_consumer\_id.get(),  
 id\_delivery\_order=text\_consumer\_id\_order.get(),  
 id\_address=text\_consumer\_id\_adress.get(),  
 contact\_details=text\_consumer\_id\_contact\_details.get()  
 )  
**def** output\_consumer():  
 textOutputСonsumer.delete(1.0, END)  
 textOutputСonsumer.insert(1.0, BD.str\_table(Consumer))  
  
**def** insert\_stoc():  
 Storage.create(  
 id=text\_stoc\_id.get(),  
 id\_product=text\_size\_transport\_size.get(),  
 address\_stoc=text\_stoc\_id\_addres.get()  
 )  
  
**def** output\_stoc():  
 textOutputStoc.delete(1.0, END)  
 textOutputStoc.insert(1.0, BD.str\_table(Storage))  
  
**def** insert\_addres\_county():  
 Country.create(  
 id=text\_addres\_country\_id.get(),  
 county=text\_addres\_country\_name.get()  
 )  
  
**def** output\_addres\_county():  
 textOutputAddres.delete(1.0, END)  
 textOutputAddres.insert(1.0, BD.str\_table(Country))  
  
**def** insert\_addres\_sity():  
 City.create(  
 id=text\_addres\_city\_id.get(),  
 city=text\_addres\_city\_name.get()  
 )  
  
**def** output\_addres\_sity():  
 textOutputAddres.delete(1.0, END)  
 textOutputAddres.insert(1.0, BD.str\_table(City))  
  
**def** insert\_addres\_street():  
 Street.create(  
 id=text\_addres\_street\_id.get(),  
 street=text\_addres\_street\_name.get()  
 )  
  
**def** output\_addres\_street():  
 textOutputAddres.delete(1.0, END)  
 textOutputAddres.insert(1.0, BD.str\_table(Street))  
  
**def** insert\_addres\_bilding():  
 Building.create(  
 id=text\_addres\_bilding\_id.get(),  
 building=text\_addres\_bilding\_name.get()  
 )  
  
**def** output\_addres\_bilding():  
 textOutputAddres.delete(1.0, END)  
 textOutputAddres.insert(1.0, BD.str\_table(Building))  
  
**def** insert\_addres():  
 Address.create(  
 id=text\_addres\_id.get(),  
 country=text\_addres\_county.get(),  
 city=text\_addres\_city.get(),  
 street=text\_addres\_street.get(),  
 bilding=text\_addres\_bilding.get()  
 )  
  
**def** output\_addres():  
 textOutputAddres.delete(1.0, END)  
 textOutputAddres.insert(1.0, BD.str\_table(Address))  
**if** \_\_name\_\_ == **'\_\_main\_\_'**:  
  
  
  
 window = Tk()  
 window.title(**'База данных транспортной компании'**)  
 window.geometry(**'1200x800'**)  
 tables = ttk.Notebook(window)  
  
  
  
  
  
 *# курьер* frame\_courier = ttk.Frame(tables)  
 tables.add(frame\_courier, text=**'Курьер'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_courier, text=**"Контактные данные куьера"**)  
 lbl.grid(row=0,column=1)  
  
 label\_courier\_contact\_details\_id = Label(frame\_courier, text=**'id:'**)  
 label\_courier\_contact\_details\_id.grid(row=1, column=0)  
 text\_courier\_contact\_details\_id = Entry(frame\_courier, width=40)  
 text\_courier\_contact\_details\_id.grid(row=1, column=1)  
  
 label\_courier\_id = Label(frame\_courier, text=**'ФИО:'**)  
 label\_courier\_id.grid(row=2, column=0)  
 text\_courier\_contact\_details\_fio = Entry(frame\_courier, width=40)  
 text\_courier\_contact\_details\_fio.grid(row=2, column=1)  
  
 label\_courier\_contact\_details\_phone = Label(frame\_courier, text=**'телефон:'**)  
 label\_courier\_contact\_details\_phone.grid(row=3, column=0)  
 text\_courier\_contact\_details\_phone = Entry(frame\_courier, width=40)  
 text\_courier\_contact\_details\_phone.grid(row=3, column=1)  
  
 label\_courier\_contact\_details\_email = Label(frame\_courier, text=**'почта:'**)  
 label\_courier\_contact\_details\_email.grid(row=4, column=0)  
 text\_courier\_contact\_details\_email = Entry(frame\_courier, width=40)  
 text\_courier\_contact\_details\_email.grid(row=4, column=1)  
  
 btn = Button(frame\_courier, text=**'Добавить'**, command=insert\_courier\_contact\_details)  
 btn.grid(row=5, column=0)  
  
 btnOutput = Button(frame\_courier, text=**'Вывести таблицу'**, command=output\_courier\_contact\_details)  
 btnOutput.grid(row=5, column=1)  
  
 lbl = Label(frame\_courier, text=**"Курьер"**)  
 lbl.grid(row=6, column=1)  
  
 label\_courier\_id = Label(frame\_courier, text=**'id:'**)  
 label\_courier\_id.grid(row=7, column=0)  
 text\_courier\_id = Entry(frame\_courier, width=40)  
 text\_courier\_id.grid(row=7, column=1)  
  
 label\_courier\_id\_transport = Label(frame\_courier, text=**'id транспорта:'**)  
 label\_courier\_id\_transport.grid(row=8, column=0)  
 text\_courier\_id\_transport = Entry(frame\_courier, width=40)  
 text\_courier\_id\_transport.grid(row=8, column=1)  
  
 label\_courier\_location = Label(frame\_courier, text=**'местоположение:'**)  
 label\_courier\_location.grid(row=9, column=0)  
 text\_courier\_location = Entry(frame\_courier, width=40)  
 text\_courier\_location.grid(row=9, column=1)  
  
  
 label\_courier\_id\_delivery\_order = Label(frame\_courier, text=**'id заказа:'**)  
 label\_courier\_id\_delivery\_order.grid(row=10, column=0)  
 text\_courier\_id\_delivery\_order = Entry(frame\_courier, width=40)  
 text\_courier\_id\_delivery\_order.grid(row=10, column=1)  
  
  
 label\_courier\_id\_contact\_details = Label(frame\_courier, text=**'id контактных данных:'**)  
 label\_courier\_id\_contact\_details.grid(row=11, column=0)  
 text\_courier\_id\_contact\_details = Entry(frame\_courier, width=40)  
 text\_courier\_id\_contact\_details.grid(row=11, column=1)  
  
 btn = Button(frame\_courier, text=**'Добавить'**, command=insert\_courier)  
 btn.grid(row=12, column=0)  
  
 btnOutput = Button(frame\_courier, text=**'Вывести таблицу'**, command=output\_courier)  
 btnOutput.grid(row=12, column=1)  
  
 textOutputCourier = Text(frame\_courier, width=200, height=20, font=**'Consolas 10'**)  
 textOutputCourier.place(x=10, y=300)  
  
 *# транспорт* frame\_transport = ttk.Frame(tables)  
 tables.add(frame\_transport, text=**'Транспорт'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_transport, text=**"Тип транспорта"**)  
 lbl.grid(row=0, column=1)  
  
 label\_type\_transport\_id = Label(frame\_transport, text=**'id:'**)  
 label\_type\_transport\_id.grid(row=1, column=0)  
 text\_type\_transport\_id = Entry(frame\_transport, width=40)  
 text\_type\_transport\_id.grid(row=1, column=1)  
  
 label\_type\_transport\_type = Label(frame\_transport, text=**'тип:'**)  
 label\_type\_transport\_type.grid(row=2, column=0)  
 text\_type\_transport\_type = Entry(frame\_transport, width=40)  
 text\_type\_transport\_type.grid(row=2, column=1)  
  
 btn = Button(frame\_transport, text=**'Добавить'**, command=insert\_type\_transport)  
 btn.grid(row=3, column=0)  
  
 btnOutput = Button(frame\_transport, text=**'Вывести таблицу'**, command=output\_type\_transport)  
 btnOutput.grid(row=3, column=1)  
  
 lbl = Label(frame\_transport, text=**"Размер транспорта"**)  
 lbl.grid(row=0, column=3)  
  
 label\_size\_transport\_id = Label(frame\_transport, text=**'id:'**)  
 label\_size\_transport\_id.grid(row=1, column=2)  
 text\_size\_transport\_id = Entry(frame\_transport, width=40)  
 text\_size\_transport\_id.grid(row=1, column=3)  
  
 label\_size\_transport\_size = Label(frame\_transport, text=**'размер:'**)  
 label\_size\_transport\_size.grid(row=2, column=2)  
 text\_size\_transport\_size = Entry(frame\_transport, width=40)  
 text\_size\_transport\_size.grid(row=2, column=3)  
  
 label\_size\_transport\_type = Label(frame\_transport, text=**'тип транспорта:'**)  
 label\_size\_transport\_type.grid(row=3, column=2)  
 text\_size\_transport\_type = Entry(frame\_transport, width=40)  
 text\_size\_transport\_type.grid(row=3, column=3)  
  
 btn = Button(frame\_transport, text=**'Добавить'**, command=insert\_size\_transport)  
 btn.grid(row=4, column=2)  
  
 btnOutput = Button(frame\_transport, text=**'Вывести таблицу'**, command=output\_size\_transport)  
 btnOutput.grid(row=4, column=3)  
  
 lbl = Label(frame\_transport, text=**"Модель транспорта"**)  
 lbl.grid(row=0, column=5)  
  
 label\_model\_transport\_id = Label(frame\_transport, text=**'id:'**)  
 label\_model\_transport\_id.grid(row=1, column=4)  
 text\_model\_transport\_id = Entry(frame\_transport, width=40)  
 text\_model\_transport\_id.grid(row=1, column=5)  
  
 label\_model\_transport\_model = Label(frame\_transport, text=**'модель:'**)  
 label\_model\_transport\_model.grid(row=2, column=4)  
 text\_model\_transport\_model = Entry(frame\_transport, width=40)  
 text\_model\_transport\_model.grid(row=2, column=5)  
  
 btn = Button(frame\_transport, text=**'Добавить'**, command=insert\_model\_transport)  
 btn.grid(row=3, column=4)  
  
 btnOutput = Button(frame\_transport, text=**'Вывести таблицу'**, command=output\_model\_transport)  
 btnOutput.grid(row=3, column=5)  
  
 lbl = Label(frame\_transport, text=**"Транспорт"**)  
 lbl.grid(row=5, column=1)  
  
 label\_transport\_id = Label(frame\_transport, text=**'id:'**)  
 label\_transport\_id.grid(row=6, column=0)  
 text\_transport\_id = Entry(frame\_transport, width=40)  
 text\_transport\_id.grid(row=6, column=1)  
  
 label\_transport\_type\_size = Label(frame\_transport, text=**'id типа и размера:'**)  
 label\_transport\_type\_size.grid(row=7, column=0)  
 text\_transport\_type\_size = Entry(frame\_transport, width=40)  
 text\_transport\_type\_size.grid(row=7, column=1)  
  
 label\_transport\_model = Label(frame\_transport, text=**'модель:'**)  
 label\_transport\_model.grid(row=8, column=0)  
 text\_transport\_model = Entry(frame\_transport, width=40)  
 text\_transport\_model.grid(row=8, column=1)  
  
 btn = Button(frame\_transport, text=**'Добавить'**, command=insert\_transport)  
 btn.grid(row=9, column=0)  
  
 btnOutput = Button(frame\_transport, text=**'Вывести таблицу'**, command=output\_transport)  
 btnOutput.grid(row=9, column=1)  
  
 textOutputTransport = Text(frame\_transport, width=200, height=20, font=**'Consolas 10'**)  
 textOutputTransport.place(x=10, y=300)  
  
 *# транспортная компания* frame\_transport\_company = ttk.Frame(tables)  
 tables.add(frame\_transport\_company, text=**'Транспортная компания'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_transport\_company, text=**"контакты компании"**)  
 lbl.grid(row=0, column=1)  
  
 label\_transport\_company\_contact\_id = Label(frame\_transport\_company, text=**'id:'**)  
 label\_transport\_company\_contact\_id.grid(row=1, column=0)  
 text\_transport\_company\_contact\_id = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_contact\_id.grid(row=1, column=1)  
  
 label\_transport\_company\_contact\_name = Label(frame\_transport\_company, text=**'название:'**)  
 label\_transport\_company\_contact\_name.grid(row=2, column=0)  
 text\_transport\_company\_contact\_name = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_contact\_name.grid(row=2, column=1)  
  
 label\_transport\_company\_contact\_dscription = Label(frame\_transport\_company, text=**'описание:'**)  
 label\_transport\_company\_contact\_dscription.grid(row=3, column=0)  
 text\_transport\_company\_contact\_dscription = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_contact\_dscription.grid(row=3, column=1)  
  
 label\_transport\_company\_contact\_email = Label(frame\_transport\_company, text=**'email:'**)  
 label\_transport\_company\_contact\_email.grid(row=4, column=0)  
 text\_transport\_company\_contact\_email = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_contact\_email.grid(row=4, column=1)  
  
 label\_transport\_company\_contact\_phone = Label(frame\_transport\_company, text=**'телефон:'**)  
 label\_transport\_company\_contact\_phone.grid(row=5, column=0)  
 text\_transport\_company\_contact\_phone = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_contact\_phone.grid(row=5, column=1)  
  
 btn = Button(frame\_transport\_company, text=**'Добавить'**, command=insert\_transport\_company\_contact)  
 btn.grid(row=6, column=0)  
  
 btnOutput = Button(frame\_transport\_company, text=**'Вывести таблицу'**, command=output\_transport\_company\_contact)  
 btnOutput.grid(row=6, column=1)  
  
 lbl = Label(frame\_transport\_company, text=**"транспортная компания"**)  
 lbl.grid(row=0, column=3)  
  
 label\_transport\_company\_id = Label(frame\_transport\_company, text=**'id:'**)  
 label\_transport\_company\_id.grid(row=1, column=2)  
 text\_transport\_company\_id = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_id.grid(row=1, column=3)  
  
 label\_transport\_company\_id\_courier = Label(frame\_transport\_company, text=**'id курьера:'**)  
 label\_transport\_company\_id\_courier.grid(row=2, column=2)  
 text\_transport\_company\_id\_courier = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_id\_courier.grid(row=2, column=3)  
  
 label\_transport\_company\_id\_order = Label(frame\_transport\_company, text=**'id заказа:'**)  
 label\_transport\_company\_id\_order.grid(row=3, column=2)  
 text\_transport\_company\_id\_order = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_id\_order.grid(row=3, column=3)  
  
 label\_transport\_company\_id\_adress = Label(frame\_transport\_company, text=**'id адреса:'**)  
 label\_transport\_company\_id\_adress.grid(row=4, column=2)  
 text\_transport\_company\_id\_adress = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_id\_adress.grid(row=4, column=3)  
  
 label\_transport\_company\_id\_contact\_details = Label(frame\_transport\_company, text=**'id контактных данных:'**)  
 label\_transport\_company\_id\_contact\_details.grid(row=5, column=2)  
 text\_transport\_company\_id\_contact\_details = Entry(frame\_transport\_company, width=40)  
 text\_transport\_company\_id\_contact\_details.grid(row=5, column=3)  
  
 btn = Button(frame\_transport\_company, text=**'Добавить'**, command=insert\_transport\_company)  
 btn.grid(row=6, column=2)  
  
 btnOutput = Button(frame\_transport\_company, text=**'Вывести таблицу'**, command=output\_transport\_company)  
 btnOutput.grid(row=6, column=3)  
  
  
 textOutputTransportCompany = Text(frame\_transport\_company, width=200, height=20, font=**'Consolas 10'**)  
 textOutputTransportCompany.place(x=10, y=300)  
  
 *# заказ доставка* frame\_delivery\_order = ttk.Frame(tables)  
 tables.add(frame\_delivery\_order, text=**'Заказ доставки'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_delivery\_order, text=**"тип доставки"**)  
 lbl.grid(row=0, column=1)  
  
 label\_delivery\_order\_type\_id = Label(frame\_delivery\_order, text=**'id:'**)  
 label\_delivery\_order\_type\_id.grid(row=1, column=0)  
 text\_delivery\_order\_type\_id = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_type\_id.grid(row=1, column=1)  
  
 label\_delivery\_order\_type\_type = Label(frame\_delivery\_order, text=**'тип:'**)  
 label\_delivery\_order\_type\_type.grid(row=2, column=0)  
 text\_delivery\_order\_type\_type = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_type\_type.grid(row=2, column=1)  
  
 btn = Button(frame\_delivery\_order, text=**'Добавить'**, command=insert\_delivery\_order\_type)  
 btn.grid(row=3, column=0)  
  
 btnOutput = Button(frame\_delivery\_order, text=**'Вывести таблицу'**, command=output\_delivery\_order\_type)  
 btnOutput.grid(row=3, column=1)  
  
 lbl = Label(frame\_delivery\_order, text=**"заказ доставки"**)  
 lbl.grid(row=0, column=3)  
  
 label\_delivery\_order\_id = Label(frame\_delivery\_order, text=**'id:'**)  
 label\_delivery\_order\_id.grid(row=1, column=2)  
 text\_delivery\_order\_id = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_id.grid(row=1, column=3)  
  
 label\_delivery\_order\_id\_product = Label(frame\_delivery\_order, text=**'id товара:'**)  
 label\_delivery\_order\_id\_product.grid(row=2, column=2)  
 text\_delivery\_order\_id\_product = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_id\_product.grid(row=2, column=3)  
  
 label\_delivery\_order\_price = Label(frame\_delivery\_order, text=**'цена:'**)  
 label\_delivery\_order\_price.grid(row=3, column=2)  
 text\_delivery\_order\_price = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_price.grid(row=3, column=3)  
  
 label\_delivery\_order\_start\_date = Label(frame\_delivery\_order, text=**'дата поступления:'**)  
 label\_delivery\_order\_start\_date.grid(row=4, column=2)  
 text\_delivery\_order\_start\_date = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_start\_date.grid(row=4, column=3)  
  
 label\_delivery\_order\_end\_date = Label(frame\_delivery\_order, text=**'дата завершения:'**)  
 label\_delivery\_order\_end\_date.grid(row=5, column=2)  
 text\_delivery\_order\_end\_date = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_end\_date.grid(row=5, column=3)  
  
 label\_delivery\_order\_type\_delivery = Label(frame\_delivery\_order, text=**'id типа доставки:'**)  
 label\_delivery\_order\_type\_delivery.grid(row=6, column=2)  
 text\_delivery\_order\_type\_delivery = Entry(frame\_delivery\_order, width=40)  
 text\_delivery\_order\_type\_delivery.grid(row=6, column=3)  
  
 btn = Button(frame\_delivery\_order, text=**'Добавить'**, command=insert\_delivery\_order)  
 btn.grid(row=7, column=2)  
  
 btnOutput = Button(frame\_delivery\_order, text=**'Вывести таблицу'**, command=output\_delivery\_order)  
 btnOutput.grid(row=7, column=3)  
  
 textOutputDeliveryOrder = Text(frame\_delivery\_order, width=200, height=20, font=**'Consolas 10'**)  
 textOutputDeliveryOrder.place(x=10, y=300)  
  
 *# поставщик* frame\_provider\_contact\_details = ttk.Frame(tables)  
 tables.add(frame\_provider\_contact\_details, text=**'Поставщик'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_provider\_contact\_details, text=**"контакты поставщика"**)  
 lbl.grid(row=0, column=1)  
  
 label\_provider\_contact\_details\_id = Label(frame\_provider\_contact\_details, text=**'id:'**)  
 label\_provider\_contact\_details\_id.grid(row=1, column=0)  
 text\_provider\_contact\_details\_id = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_contact\_details\_id.grid(row=1, column=1)  
  
 label\_provider\_contact\_details\_name = Label(frame\_provider\_contact\_details, text=**'название:'**)  
 label\_provider\_contact\_details\_name.grid(row=2, column=0)  
 text\_provider\_contact\_details\_name = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_contact\_details\_name.grid(row=2, column=1)  
  
 label\_provider\_contact\_details\_dscription = Label(frame\_provider\_contact\_details, text=**'описание:'**)  
 label\_provider\_contact\_details\_dscription.grid(row=3, column=0)  
 text\_provider\_contact\_details\_dscription = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_contact\_details\_dscription.grid(row=3, column=1)  
  
 label\_provider\_contact\_details\_email = Label(frame\_provider\_contact\_details, text=**'email:'**)  
 label\_provider\_contact\_details\_email.grid(row=4, column=0)  
 text\_provider\_contact\_details\_email = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_contact\_details\_email.grid(row=4, column=1)  
  
 label\_provider\_contact\_details\_phone = Label(frame\_provider\_contact\_details, text=**'телефон:'**)  
 label\_provider\_contact\_details\_phone.grid(row=5, column=0)  
 text\_provider\_contact\_details\_phone = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_contact\_details\_phone.grid(row=5, column=1)  
  
 btn = Button(frame\_provider\_contact\_details, text=**'Добавить'**, command=insert\_provider\_contact)  
 btn.grid(row=6, column=0)  
  
 btnOutput = Button(frame\_provider\_contact\_details, text=**'Вывести таблицу'**, command=output\_provider\_contact)  
 btnOutput.grid(row=6, column=1)  
  
 lbl = Label(frame\_provider\_contact\_details, text=**"поставщик"**)  
 lbl.grid(row=0, column=3)  
  
 label\_provider\_id = Label(frame\_provider\_contact\_details, text=**'id:'**)  
 label\_provider\_id.grid(row=1, column=2)  
 text\_provider\_id = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_id.grid(row=1, column=3)  
  
 label\_provider\_id\_product = Label(frame\_provider\_contact\_details, text=**'id продукта:'**)  
 label\_provider\_id\_product.grid(row=2, column=2)  
 text\_provider\_id\_product = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_id\_product.grid(row=2, column=3)  
  
 label\_provider\_id\_adress = Label(frame\_provider\_contact\_details, text=**'id адреса:'**)  
 label\_provider\_id\_adress.grid(row=3, column=2)  
 text\_provider\_id\_adress = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_id\_adress.grid(row=3, column=3)  
  
 label\_provider\_id\_contact\_details = Label(frame\_provider\_contact\_details, text=**'id контактных данных:'**)  
 label\_provider\_id\_contact\_details.grid(row=4, column=2)  
 text\_provider\_id\_contact\_details = Entry(frame\_provider\_contact\_details, width=40)  
 text\_provider\_id\_contact\_details.grid(row=4, column=3)  
  
 btn = Button(frame\_provider\_contact\_details, text=**'Добавить'**, command=insert\_provider)  
 btn.grid(row=5, column=2)  
  
 btnOutput = Button(frame\_provider\_contact\_details, text=**'Вывести таблицу'**, command=output\_provider)  
 btnOutput.grid(row=5, column=3)  
  
 textOutputProvider = Text(frame\_provider\_contact\_details, width=200, height=20, font=**'Consolas 10'**)  
 textOutputProvider.place(x=10, y=300)  
  
 *# товар* frame\_product = ttk.Frame(tables)  
 tables.add(frame\_product, text=**'Товар'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_product, text=**"наименование товара"**)  
 lbl.grid(row=0, column=1)  
  
 label\_name\_product\_id = Label(frame\_product, text=**'id:'**)  
 label\_name\_product\_id.grid(row=1, column=0)  
 text\_name\_product\_id = Entry(frame\_product, width=40)  
 text\_name\_product\_id.grid(row=1, column=1)  
  
 label\_name\_product\_name = Label(frame\_product, text=**'название:'**)  
 label\_name\_product\_name.grid(row=2, column=0)  
 text\_name\_product\_name = Entry(frame\_product, width=40)  
 text\_name\_product\_name.grid(row=2, column=1)  
  
 btn = Button(frame\_product, text=**'Добавить'**, command=insert\_name\_product)  
 btn.grid(row=3, column=0)  
  
 btnOutput = Button(frame\_product, text=**'Вывести таблицу'**, command=output\_name\_product)  
 btnOutput.grid(row=3, column=1)  
  
 lbl = Label(frame\_product, text=**"товар"**)  
 lbl.grid(row=0, column=3)  
  
 label\_product\_id = Label(frame\_product, text=**'id:'**)  
 label\_product\_id.grid(row=1, column=2)  
 text\_product\_id = Entry(frame\_product, width=40)  
 text\_product\_id.grid(row=1, column=3)  
  
 label\_product\_size = Label(frame\_product, text=**'размер:'**)  
 label\_product\_size.grid(row=2, column=2)  
 text\_product\_size = Entry(frame\_product, width=40)  
 text\_product\_size.grid(row=2, column=3)  
  
 label\_product\_weight = Label(frame\_product, text=**'вес:'**)  
 label\_product\_weight.grid(row=3, column=2)  
 text\_product\_weight = Entry(frame\_product, width=40)  
 text\_product\_weight.grid(row=3, column=3)  
  
 label\_product\_id\_name = Label(frame\_product, text=**'id наименования:'**)  
 label\_product\_id\_name.grid(row=4, column=2)  
 text\_product\_id\_name = Entry(frame\_product, width=40)  
 text\_product\_id\_name.grid(row=4, column=3)  
  
 label\_product\_id\_adress = Label(frame\_product, text=**'адрес где товар:'**)  
 label\_product\_id\_adress.grid(row=5, column=2)  
 text\_product\_id\_adress = Entry(frame\_product, width=40)  
 text\_product\_id\_adress.grid(row=5, column=3)  
  
 btn = Button(frame\_product, text=**'Добавить'**, command=insert\_product)  
 btn.grid(row=6, column=2)  
  
 btnOutput = Button(frame\_product, text=**'Вывести таблицу'**, command=output\_product)  
 btnOutput.grid(row=6, column=3)  
  
 textOutputProduct = Text(frame\_product, width=200, height=20, font=**'Consolas 10'**)  
 textOutputProduct.place(x=10, y=300)  
  
 *# потребитель* frame\_consumer = ttk.Frame(tables)  
 tables.add(frame\_consumer, text=**'Потребитель'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_consumer, text=**"контакты потребителя"**)  
 lbl.grid(row=0, column=1)  
  
 label\_consumer\_contact\_id = Label(frame\_consumer, text=**'id:'**)  
 label\_consumer\_contact\_id.grid(row=1, column=0)  
 text\_consumer\_contact\_id = Entry(frame\_consumer, width=40)  
 text\_consumer\_contact\_id.grid(row=1, column=1)  
  
 label\_consumer\_contact\_name = Label(frame\_consumer, text=**'название:'**)  
 label\_consumer\_contact\_name.grid(row=2, column=0)  
 text\_consumer\_contact\_name = Entry(frame\_consumer, width=40)  
 text\_consumer\_contact\_name.grid(row=2, column=1)  
  
 label\_consumer\_contact\_dscription = Label(frame\_consumer, text=**'описание:'**)  
 label\_consumer\_contact\_dscription.grid(row=3, column=0)  
 text\_consumer\_contact\_dscription = Entry(frame\_consumer, width=40)  
 text\_consumer\_contact\_dscription.grid(row=3, column=1)  
  
 label\_consumer\_contact\_email = Label(frame\_consumer, text=**'email:'**)  
 label\_consumer\_contact\_email.grid(row=4, column=0)  
 text\_consumer\_contact\_email = Entry(frame\_consumer, width=40)  
 text\_consumer\_contact\_email.grid(row=4, column=1)  
  
 label\_consumer\_contact\_phone = Label(frame\_consumer, text=**'телефон:'**)  
 label\_consumer\_contact\_phone.grid(row=5, column=0)  
 text\_consumer\_contact\_phone = Entry(frame\_consumer, width=40)  
 text\_consumer\_contact\_phone.grid(row=5, column=1)  
  
 btn = Button(frame\_consumer, text=**'Добавить'**, command=insert\_consumer\_contact)  
 btn.grid(row=6, column=0)  
  
 btnOutput = Button(frame\_consumer, text=**'Вывести таблицу'**, command=output\_consumer\_contact)  
 btnOutput.grid(row=6, column=1)  
  
 lbl = Label(frame\_consumer, text=**"потребитель"**)  
 lbl.grid(row=0, column=3)  
  
 label\_consumer\_id = Label(frame\_consumer, text=**'id:'**)  
 label\_consumer\_id.grid(row=1, column=2)  
 text\_consumer\_id = Entry(frame\_consumer, width=40)  
 text\_consumer\_id.grid(row=1, column=3)  
  
 label\_consumer\_id\_order = Label(frame\_consumer, text=**'id заказа:'**)  
 label\_consumer\_id\_order.grid(row=2, column=2)  
 text\_consumer\_id\_order = Entry(frame\_consumer, width=40)  
 text\_consumer\_id\_order.grid(row=2, column=3)  
  
 label\_consumer\_id\_adress = Label(frame\_consumer, text=**'id адреса:'**)  
 label\_consumer\_id\_adress.grid(row=3, column=2)  
 text\_consumer\_id\_adress = Entry(frame\_consumer, width=40)  
 text\_consumer\_id\_adress.grid(row=3, column=3)  
  
 label\_consumer\_id\_contact\_details = Label(frame\_consumer, text=**'id контактных данных:'**)  
 label\_consumer\_id\_contact\_details.grid(row=4, column=2)  
 text\_consumer\_id\_contact\_details = Entry(frame\_consumer, width=40)  
 text\_consumer\_id\_contact\_details.grid(row=4, column=3)  
  
 btn = Button(frame\_consumer, text=**'Добавить'**, command=insert\_transport\_company)  
 btn.grid(row=5, column=2)  
  
 btnOutput = Button(frame\_consumer, text=**'Вывести таблицу'**, command=output\_consumer)  
 btnOutput.grid(row=5, column=3)  
  
 textOutputСonsumer = Text(frame\_consumer, width=200, height=20, font=**'Consolas 10'**)  
 textOutputСonsumer.place(x=10, y=200)  
  
 *# склад* frame\_stoc = ttk.Frame(tables)  
 tables.add(frame\_stoc, text=**'Склад'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_stoc, text=**"склад"**)  
 lbl.grid(row=0, column=1)  
  
 label\_stoc\_id = Label(frame\_stoc, text=**'id:'**)  
 label\_stoc\_id.grid(row=1, column=0)  
 text\_stoc\_id = Entry(frame\_stoc, width=40)  
 text\_stoc\_id.grid(row=1, column=1)  
  
 label\_stoc\_id\_product = Label(frame\_stoc, text=**'id продукта:'**)  
 label\_stoc\_id\_product.grid(row=2, column=0)  
 text\_stoc\_id\_product = Entry(frame\_stoc, width=40)  
 text\_stoc\_id\_product.grid(row=2, column=1)  
  
 label\_stoc\_id\_addres = Label(frame\_stoc, text=**'id адреса:'**)  
 label\_stoc\_id\_addres.grid(row=3, column=0)  
 text\_stoc\_id\_addres = Entry(frame\_stoc, width=40)  
 text\_stoc\_id\_addres.grid(row=3, column=1)  
  
 btn = Button(frame\_stoc, text=**'Добавить'**, command=insert\_stoc)  
 btn.grid(row=4, column=0)  
  
 btnOutput = Button(frame\_stoc, text=**'Вывести таблицу'**, command=output\_stoc)  
 btnOutput.grid(row=4, column=1)  
  
 textOutputStoc = Text(frame\_stoc, width=200, height=20, font=**'Consolas 10'**)  
 textOutputStoc.place(x=10, y=200)  
  
 *# адресс* frame\_addres = ttk.Frame(tables)  
 tables.add(frame\_addres, text=**'Адрес'**)  
 tables.pack(expand=2, fill=**'both'**)  
  
 lbl = Label(frame\_addres, text=**"страна"**)  
 lbl.grid(row=0, column=1)  
  
 label\_addres\_country\_id = Label(frame\_addres, text=**'id:'**)  
 label\_addres\_country\_id.grid(row=1, column=0)  
 text\_addres\_country\_id = Entry(frame\_addres, width=40)  
 text\_addres\_country\_id.grid(row=1, column=1)  
  
 label\_addres\_country\_name = Label(frame\_addres, text=**'название:'**)  
 label\_addres\_country\_name.grid(row=2, column=0)  
 text\_addres\_country\_name = Entry(frame\_addres, width=40)  
 text\_addres\_country\_name.grid(row=2, column=1)  
  
 btn = Button(frame\_addres, text=**'Добавить'**, command=insert\_addres\_county)  
 btn.grid(row=3, column=0)  
  
 btnOutput = Button(frame\_addres, text=**'Вывести таблицу'**, command=output\_addres\_county)  
 btnOutput.grid(row=3, column=1)  
  
 lbl = Label(frame\_addres, text=**"город"**)  
 lbl.grid(row=0, column=3)  
  
 label\_addres\_city\_id = Label(frame\_addres, text=**'id:'**)  
 label\_addres\_city\_id.grid(row=1, column=2)  
 text\_addres\_city\_id = Entry(frame\_addres, width=40)  
 text\_addres\_city\_id.grid(row=1, column=3)  
  
 label\_addres\_city\_name = Label(frame\_addres, text=**'название:'**)  
 label\_addres\_city\_name.grid(row=2, column=2)  
 text\_addres\_city\_name = Entry(frame\_addres, width=40)  
 text\_addres\_city\_name.grid(row=2, column=3)  
  
 btn = Button(frame\_addres, text=**'Добавить'**, command=insert\_addres\_sity)  
 btn.grid(row=3, column=2)  
  
 btnOutput = Button(frame\_addres, text=**'Вывести таблицу'**, command=output\_addres\_sity)  
 btnOutput.grid(row=3, column=3)  
  
 lbl = Label(frame\_addres, text=**"улица"**)  
 lbl.grid(row=0, column=5)  
  
 label\_addres\_street\_id = Label(frame\_addres, text=**'id:'**)  
 label\_addres\_street\_id.grid(row=1, column=4)  
 text\_addres\_street\_id = Entry(frame\_addres, width=40)  
 text\_addres\_street\_id.grid(row=1, column=5)  
  
 label\_addres\_street\_name = Label(frame\_addres, text=**'название:'**)  
 label\_addres\_street\_name.grid(row=2, column=4)  
 text\_addres\_street\_name = Entry(frame\_addres, width=40)  
 text\_addres\_street\_name.grid(row=2, column=5)  
  
 btn = Button(frame\_addres, text=**'Добавить'**, command=insert\_addres\_street)  
 btn.grid(row=3, column=4)  
  
 btnOutput = Button(frame\_addres, text=**'Вывести таблицу'**, command=output\_addres\_street)  
 btnOutput.grid(row=3, column=5)  
  
 lbl = Label(frame\_addres, text=**"здание "**)  
 lbl.grid(row=0, column=7)  
  
 label\_addres\_bilding\_id = Label(frame\_addres, text=**'id:'**)  
 label\_addres\_bilding\_id.grid(row=1, column=6)  
 text\_addres\_bilding\_id = Entry(frame\_addres, width=40)  
 text\_addres\_bilding\_id.grid(row=1, column=7)  
  
 label\_addres\_bilding\_name = Label(frame\_addres, text=**'название:'**)  
 label\_addres\_bilding\_name.grid(row=2, column=6)  
 text\_addres\_bilding\_name = Entry(frame\_addres, width=40)  
 text\_addres\_bilding\_name.grid(row=2, column=7)  
  
 btn = Button(frame\_addres, text=**'Добавить'**, command=insert\_addres\_bilding)  
 btn.grid(row=3, column=6)  
  
 btnOutput = Button(frame\_addres, text=**'Вывести таблицу'**, command=output\_addres\_bilding)  
 btnOutput.grid(row=3, column=7)  
  
 lbl = Label(frame\_addres, text=**"адрес "**)  
 lbl.grid(row=4, column=1)  
  
 label\_addres\_id = Label(frame\_addres, text=**'id:'**)  
 label\_addres\_id.grid(row=5, column=0)  
 text\_addres\_id = Entry(frame\_addres, width=40)  
 text\_addres\_id.grid(row=5, column=1)  
  
 label\_addres\_county = Label(frame\_addres, text=**'id страны:'**)  
 label\_addres\_county.grid(row=6, column=0)  
 text\_addres\_county = Entry(frame\_addres, width=40)  
 text\_addres\_county.grid(row=6, column=1)  
  
 label\_addres\_city = Label(frame\_addres, text=**'id города:'**)  
 label\_addres\_city.grid(row=7, column=0)  
 text\_addres\_city = Entry(frame\_addres, width=40)  
 text\_addres\_city.grid(row=7, column=1)  
  
 label\_addres\_street = Label(frame\_addres, text=**'id улицы:'**)  
 label\_addres\_street.grid(row=8, column=0)  
 text\_addres\_street = Entry(frame\_addres, width=40)  
 text\_addres\_street.grid(row=8, column=1)  
  
 label\_addres\_bilding = Label(frame\_addres, text=**'id здания:'**)  
 label\_addres\_bilding.grid(row=9, column=0)  
 text\_addres\_bilding = Entry(frame\_addres, width=40)  
 text\_addres\_bilding.grid(row=9, column=1)  
  
 btn = Button(frame\_addres, text=**'Добавить'**, command=insert\_addres)  
 btn.grid(row=10, column=0)  
  
 btnOutput = Button(frame\_addres, text=**'Вывести таблицу'**, command=output\_addres)  
 btnOutput.grid(row=10, column=1)  
  
 textOutputAddres = Text(frame\_addres, width=200, height=20, font=**'Consolas 10'**)  
 textOutputAddres.place(x=10, y=300)  
  
 window.mainloop()



