



УНИВЕРСИТЕТСКИЙ
КОЛЛЕДЖ БРИКС

Автономная некоммерческая организация
профессиональная образовательная организация
«Университетский колледж БРИКС»
(Университетский колледж БРИКС)

ОТЧЁТ О ПРАКТИКЕ
учебной практики/ производственной практики/
производственной практики (по профилю специальности)/
производственной практики (преддипломной)

по профессиональному модулю
ПМ.02 Осуществление интеграции программных модулей

Специальность /профессия 09.02.07 Информационные системы и программирование

Студента(ки) _____ курса ____ группы

Форма обучения _____
(очная, заочная, очно-заочная)

(ФИО)

Место проведения практики (организация) _____
наименование организации, юридический адрес

Срок прохождения практики с «06» июня 2025 г. по «20» июня 2025 г.

Отчет сдан « _____ » _____ 20__ г.

Руководитель практики от
образовательной организации _____ / _____ /
(подпись/Ф.И.О.)

Оценка о защите отчета _____ (_____)

Содержание:

Практическая часть:

Предметная область.....	3
Работа с vs-code.....	3
Файлы для создания сайта.....	3
ERD-диаграмма и код БД.....	53
Скриншоты страниц сайта и информация к ним.....	55

Предметная область:

Приложение для учета номенклатуры, а также поставщиков и покупателей в компании. Выбрана для того, чтобы сделать визуализацию в работе с базой данных. И для повышения эффективности работы.

Работа с vs code:

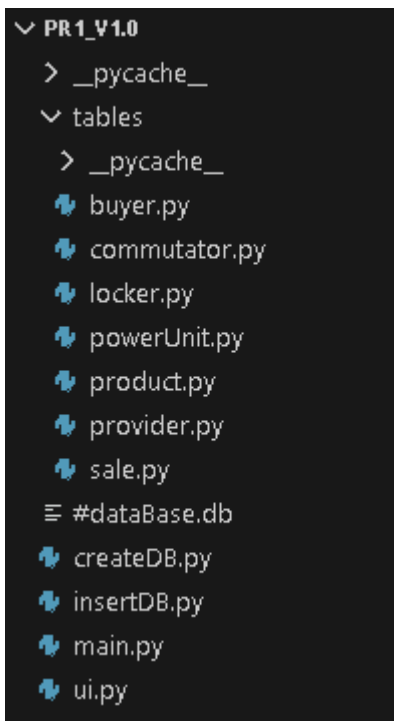
Для начала работы перейдём в терминал visual studio code и введём следующие команду:

```
py -m pip install PyQt6
```

это единственная команда которая нам понадобится.

Файлы для создания сайта:

Папка проекта (site), в которой представлены следующие файлы:



tables: Папка, содержащая логику работ таблиц.

buyer.py: Файл отвечающий за логику таблицы *buyer*.

commutator.py: Файл отвечающий за логику таблицы *commutator*.

locker.py: Файл отвечающий за логику таблицы *locker*.

powerUnit.py: Файл отвечающий за логику таблицы *powerUnit*.

product.py: Файл отвечающий за логику таблицы *product*.

provider.py: Файл отвечающий за логику таблицы *provider*.

sale.py: Файл отвечающий за логику таблицы *sale*.

#dataBase.db: Файл с базой данных SQL.

createDB.py: Создает таблицы в SQLite, если их ещё нет.

insertDB.py: Заполняет таблицы БД по 10 элементов.

ui.py: Создание интерфейса приложения с помощью библиотеки PyQt.

main.py: Отвечает за запуск приложения.

Ниже представлены коды всех файлов и их пути в расчёт берем что изначальный путь идёт из папки PR1_v1.0:

main.py

```
import sys
from ui import Ui_Form
from PyQt6 import QtWidgets
from tables.provider import provider
from tables.locker import locker
from tables.commutator import commutator
from tables.powerUnit import powerUnit
from tables.product import product
from tables.sale import sale
from tables.buyer import buyer

app = QtWidgets.QApplication(sys.argv)
Form = QtWidgets.QWidget()
ui = Ui_Form()
ui.setupUi(Form)

ui.menu.currentRowChanged.connect(ui.stackedWidget.setCurrentIndex)

provider(ui)
locker(ui)
commutator(ui)
powerUnit(ui)
product(ui)
sale(ui)
buyer(ui)

Form.show()
sys.exit(app.exec())
```

```
from PyQt6 import QtCore, QtGui, QtWidgets

class Ui_Form(object):
    def setupUi(self, Form):
        Form.setObjectName("Form")
        Form.resize(1236, 737)
        self.menu = QtWidgets.QListWidget(parent=Form)
        self.menu.setGeometry(QtCore.QRect(20, 20, 200, 691))
        self.menu.setMaximumSize(QtCore.QSize(200, 16777215))
        self.menu.setObjectName("menu")
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        item = QtWidgets.QListWidgetItem()
        self.menu.addItem(item)
        self.stackedWidget = QtWidgets.QStackedWidget(parent=Form)
        self.stackedWidget.setGeometry(QtCore.QRect(240, 20, 971, 691))
        self.stackedWidget.setObjectName("stackedWidget")
        self.stackedProvider = QtWidgets.QWidget()
        self.stackedProvider.setObjectName("stackedProvider")
        self.label_title_provider = QtWidgets.QLabel(parent=self.stackedProvider)
        self.label_title_provider.setGeometry(QtCore.QRect(0, 0, 558, 23))
        self.label_title_provider.setObjectName("label_title_provider")
        self.TVProvider = QtWidgets.QTableView(parent=self.stackedProvider)
        self.TVProvider.setGeometry(QtCore.QRect(0, 160, 971, 531))
        self.TVProvider.setObjectName("TVProvider")
        self.addProvider = QtWidgets.QPushButton(parent=self.stackedProvider)
        self.addProvider.setGeometry(QtCore.QRect(0, 60, 75, 23))
        self.addProvider.setObjectName("addProvider")
```

```

self.widget = QtWidgets.QWidget(parent=self.stackedProvider)
self.widget.setGeometry(QtCore.QRect(0, 90, 887, 55))
self.widget.setObjectName("widget")
self.horizontalLayout_2 = QtWidgets.QHBoxLayout(self.widget)
self.horizontalLayout_2.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_2.setObjectName("horizontalLayout_2")
self.horizontalLayout = QtWidgets.QHBoxLayout()
self.horizontalLayout.setObjectName("horizontalLayout")
self.verticalLayout = QtWidgets.QVBoxLayout()
self.verticalLayout.setObjectName("verticalLayout")
self.label_2 = QtWidgets.QLabel(parent=self.widget)
self.label_2.setMinimumSize(QtCore.QSize(150, 20))
self.label_2.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_2.setFont(font)
self.label_2.setObjectName("label_2")
self.verticalLayout.addWidget(self.label_2)
self.innProvider = QtWidgets.QLineEdit(parent=self.widget)
self.innProvider.setMinimumSize(QtCore.QSize(150, 20))
self.innProvider.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.innProvider.setFont(font)
self.innProvider.setObjectName("innProvider")
self.verticalLayout.addWidget(self.innProvider)
self.horizontalLayout.addLayout(self.verticalLayout)
self.verticalLayout_2 = QtWidgets.QVBoxLayout()
self.verticalLayout_2.setObjectName("verticalLayout_2")
self.label_3 = QtWidgets.QLabel(parent=self.widget)
self.label_3.setMinimumSize(QtCore.QSize(150, 20))
self.label_3.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_3.setFont(font)
self.label_3.setObjectName("label_3")
self.verticalLayout_2.addWidget(self.label_3)
self.fnameProvider = QtWidgets.QLineEdit(parent=self.widget)
self.fnameProvider.setMinimumSize(QtCore.QSize(150, 20))
self.fnameProvider.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()

```

```

font.setPointSize(10)
self.fnameProvider.setFont(font)
self.fnameProvider.setObjectName("fnameProvider")
self.verticalLayout_2.addWidget(self.fnameProvider)
self.horizontalLayout.addLayout(self.verticalLayout_2)
self.verticalLayout_3 = QtWidgets.QVBoxLayout()
self.verticalLayout_3.setObjectName("verticalLayout_3")
self.label_4 = QtWidgets.QLabel(parent=self.widget)
self.label_4.setMinimumSize(QtCore.QSize(150, 20))
self.label_4.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_4.setFont(font)
self.label_4.setObjectName("label_4")
self.verticalLayout_3.addWidget(self.label_4)
self.lnameProvider = QtWidgets.QLineEdit(parent=self.widget)
self.lnameProvider.setMinimumSize(QtCore.QSize(150, 20))
self.lnameProvider.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.lnameProvider.setFont(font)
self.lnameProvider.setObjectName("lnameProvider")
self.verticalLayout_3.addWidget(self.lnameProvider)
self.horizontalLayout.addLayout(self.verticalLayout_3)
self.verticalLayout_4 = QtWidgets.QVBoxLayout()
self.verticalLayout_4.setObjectName("verticalLayout_4")
self.label_5 = QtWidgets.QLabel(parent=self.widget)
self.label_5.setMinimumSize(QtCore.QSize(150, 20))
self.label_5.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_5.setFont(font)
self.label_5.setObjectName("label_5")
self.verticalLayout_4.addWidget(self.label_5)
self.patronymicProvider = QtWidgets.QLineEdit(parent=self.widget)
self.patronymicProvider.setMinimumSize(QtCore.QSize(150, 20))
self.patronymicProvider.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.patronymicProvider.setFont(font)
self.patronymicProvider.setObjectName("patronymicProvider")

```

```

self.verticalLayout_4.addWidget(self.patronymicProvider)
self.horizontalLayout.addLayout(self.verticalLayout_4)
self.verticalLayout_5 = QtWidgets.QVBoxLayout()
self.verticalLayout_5.setObjectName("verticalLayout_5")
self.label_6 = QtWidgets.QLabel(parent=self.widget)
self.label_6.setMinimumSize(QtCore.QSize(150, 20))
self.label_6.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_6.setFont(font)
self.label_6.setObjectName("label_6")
self.verticalLayout_5.addWidget(self.label_6)
self.telProvider = QtWidgets.QLineEdit(parent=self.widget)
self.telProvider.setMinimumSize(QtCore.QSize(150, 20))
self.telProvider.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.telProvider.setFont(font)
self.telProvider.setObjectName("telProvider")
self.verticalLayout_5.addWidget(self.telProvider)
self.horizontalLayout.addLayout(self.verticalLayout_5)
self.horizontalLayout_2.addLayout(self.horizontalLayout)
self.verticalLayout_6 = QtWidgets.QVBoxLayout()
self.verticalLayout_6.setObjectName("verticalLayout_6")
self.label_7 = QtWidgets.QLabel(parent=self.widget)
self.label_7.setMinimumSize(QtCore.QSize(91, 21))
self.label_7.setText("")
self.label_7.setObjectName("label_7")
self.verticalLayout_6.addWidget(self.label_7)
self.okProvider = QtWidgets.QPushButton(parent=self.widget)
self.okProvider.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()
font.setPointSize(10)
self.okProvider.setFont(font)
self.okProvider.setObjectName("okProvider")
self.verticalLayout_6.addWidget(self.okProvider)
self.horizontalLayout_2.addLayout(self.verticalLayout_6)
self.stackedWidget.addWidget(self.stackedProvider)
self.stackedLocker = QtWidgets.QWidget()
self.stackedLocker.setObjectName("stackedLocker")
self.label_title_provider_2 = QtWidgets.QLabel(parent=self.stackedLocker)

```



```

self.label_title_provider_2.setGeometry(QtCore.QRect(0, 0, 558, 23))
self.label_title_provider_2.setObjectName("label_title_provider_2")
self.TVLocker = QtWidgets.QTableView(parent=self.stackedLocker)
self.TVLocker.setGeometry(QtCore.QRect(0, 160, 971, 531))
self.TVLocker.setObjectName("TVLocker")
self.addLocker = QtWidgets.QPushButton(parent=self.stackedLocker)
self.addLocker.setGeometry(QtCore.QRect(0, 60, 75, 23))
self.addLocker.setObjectName("addLocker")
self.widget1 = QtWidgets.QWidget(parent=self.stackedLocker)
self.widget1.setGeometry(QtCore.QRect(2, 91, 569, 55))
self.widget1.setObjectName("widget1")
self.horizontalLayout_5 = QtWidgets.QHBoxLayout(self.widget1)
self.horizontalLayout_5.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_5.setObjectName("horizontalLayout_5")
self.verticalLayout_17 = QtWidgets.QVBoxLayout()
self.verticalLayout_17.setObjectName("verticalLayout_17")
self.label_18 = QtWidgets.QLabel(parent=self.widget1)
self.label_18.setMinimumSize(QtCore.QSize(150, 20))
self.label_18.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_18.setFont(font)
self.label_18.setObjectName("label_18")
self.verticalLayout_17.addWidget(self.label_18)
self.nameLocker = QtWidgets.QLineEdit(parent=self.widget1)
self.nameLocker.setMinimumSize(QtCore.QSize(150, 20))
self.nameLocker.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.nameLocker.setFont(font)
self.nameLocker.setObjectName("nameLocker")
self.verticalLayout_17.addWidget(self.nameLocker)
self.horizontalLayout_5.addLayout(self.verticalLayout_17)
self.verticalLayout_18 = QtWidgets.QVBoxLayout()
self.verticalLayout_18.setObjectName("verticalLayout_18")
self.label_19 = QtWidgets.QLabel(parent=self.widget1)
self.label_19.setMinimumSize(QtCore.QSize(150, 20))
self.label_19.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_19.setFont(font)

```

```

self.label_19.setObjectName("label_19")
self.verticalLayout_18.addWidget(self.label_19)
self.providerIDLocker = QtWidgets.QLineEdit(parent=self.widget1)
self.providerIDLocker.setMinimumSize(QtCore.QSize(150, 20))
self.providerIDLocker.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.providerIDLocker.setFont(font)
self.providerIDLocker.setObjectName("providerIDLocker")
self.verticalLayout_18.addWidget(self.providerIDLocker)
self.horizontalLayout_5.addLayout(self.verticalLayout_18)
self.verticalLayout_19 = QtWidgets.QVBoxLayout()
self.verticalLayout_19.setObjectName("verticalLayout_19")
self.label_20 = QtWidgets.QLabel(parent=self.widget1)
self.label_20.setMinimumSize(QtCore.QSize(150, 20))
self.label_20.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_20.setFont(font)
self.label_20.setObjectName("label_20")
self.verticalLayout_19.addWidget(self.label_20)
self.countLocker = QtWidgets.QLineEdit(parent=self.widget1)
self.countLocker.setMinimumSize(QtCore.QSize(150, 20))
self.countLocker.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.countLocker.setFont(font)
self.countLocker.setObjectName("countLocker")
self.verticalLayout_19.addWidget(self.countLocker)
self.horizontalLayout_5.addLayout(self.verticalLayout_19)
self.verticalLayout_22 = QtWidgets.QVBoxLayout()
self.verticalLayout_22.setObjectName("verticalLayout_22")
self.label_23 = QtWidgets.QLabel(parent=self.widget1)
self.label_23.setMinimumSize(QtCore.QSize(91, 21))
self.label_23.setText("")
self.label_23.setObjectName("label_23")
self.verticalLayout_22.addWidget(self.label_23)
self.okLocker = QtWidgets.QPushButton(parent=self.widget1)
self.okLocker.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()
font.setPointSize(10)

```

```

self.okLocker.setFont(font)
self.okLocker.setObjectName("okLocker")
self.verticalLayout_22.addWidget(self.okLocker)
self.horizontalLayout_5.addLayout(self.verticalLayout_22)
self.stackedWidget.addWidget(self.stackedLocker)
self.stackedCommutator = QtWidgets.QWidget()
self.stackedCommutator.setObjectName("stackedCommutator")
self.TVCommutator = QtWidgets.QTableView(parent=self.stackedCommutator)
self.TVCommutator.setGeometry(QtCore.QRect(0, 160, 971, 531))
self.TVCommutator.setObjectName("TVCommutator")
self.label_title_provider_3 =
QtWidgets.QLabel(parent=self.stackedCommutator)
self.label_title_provider_3.setGeometry(QtCore.QRect(0, 0, 558, 23))
self.label_title_provider_3.setObjectName("label_title_provider_3")
self.layoutWidget_2 = QtWidgets.QWidget(parent=self.stackedCommutator)
self.layoutWidget_2.setGeometry(QtCore.QRect(2, 91, 569, 55))
self.layoutWidget_2.setObjectName("layoutWidget_2")
self.horizontalLayout_6 = QtWidgets.QHBoxLayout(self.layoutWidget_2)
self.horizontalLayout_6.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_6.setObjectName("horizontalLayout_6")
self.verticalLayout_20 = QtWidgets.QVBoxLayout()
self.verticalLayout_20.setObjectName("verticalLayout_20")
self.label_21 = QtWidgets.QLabel(parent=self.layoutWidget_2)
self.label_21.setMinimumSize(QtCore.QSize(150, 20))
self.label_21.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_21.setFont(font)
self.label_21.setObjectName("label_21")
self.verticalLayout_20.addWidget(self.label_21)
self.nameCommutator = QtWidgets.QLineEdit(parent=self.layoutWidget_2)
self.nameCommutator.setMinimumSize(QtCore.QSize(150, 20))
self.nameCommutator.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.nameCommutator.setFont(font)
self.nameCommutator.setObjectName("nameCommutator")
self.verticalLayout_20.addWidget(self.nameCommutator)
self.horizontalLayout_6.addLayout(self.verticalLayout_20)
self.verticalLayout_21 = QtWidgets.QVBoxLayout()
self.verticalLayout_21.setObjectName("verticalLayout_21")

```

```

self.label_22 = QtWidgets.QLabel(parent=self.layoutWidget_2)
self.label_22.setMinimumSize(QtCore.QSize(150, 20))
self.label_22.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_22.setFont(font)
self.label_22.setObjectName("label_22")
self.verticalLayout_21.addWidget(self.label_22)
self.providerIDCommutator =
QtWidgets.QLineEdit(parent=self.layoutWidget_2)
self.providerIDCommutator.setMinimumSize(QtCore.QSize(150, 20))
self.providerIDCommutator.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.providerIDCommutator.setFont(font)
self.providerIDCommutator.setObjectName("providerIDCommutator")
self.verticalLayout_21.addWidget(self.providerIDCommutator)
self.horizontalLayout_6.addLayout(self.verticalLayout_21)
self.verticalLayout_23 = QtWidgets.QVBoxLayout()
self.verticalLayout_23.setObjectName("verticalLayout_23")
self.label_24 = QtWidgets.QLabel(parent=self.layoutWidget_2)
self.label_24.setMinimumSize(QtCore.QSize(150, 20))
self.label_24.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_24.setFont(font)
self.label_24.setObjectName("label_24")
self.verticalLayout_23.addWidget(self.label_24)
self.countCommutator = QtWidgets.QLineEdit(parent=self.layoutWidget_2)
self.countCommutator.setMinimumSize(QtCore.QSize(150, 20))
self.countCommutator.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.countCommutator.setFont(font)
self.countCommutator.setObjectName("countCommutator")
self.verticalLayout_23.addWidget(self.countCommutator)
self.horizontalLayout_6.addLayout(self.verticalLayout_23)
self.verticalLayout_24 = QtWidgets.QVBoxLayout()
self.verticalLayout_24.setObjectName("verticalLayout_24")
self.label_25 = QtWidgets.QLabel(parent=self.layoutWidget_2)
self.label_25.setMinimumSize(QtCore.QSize(91, 21))

```

```

self.label_25.setText("")
self.label_25.setObjectName("label_25")
self.verticalLayout_24.addWidget(self.label_25)
self.okCommutator = QtWidgets.QPushButton(parent=self.layoutWidget_2)
self.okCommutator.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()
font.setPointSize(10)
self.okCommutator.setFont(font)
self.okCommutator.setObjectName("okCommutator")
self.verticalLayout_24.addWidget(self.okCommutator)
self.horizontalLayout_6.addLayout(self.verticalLayout_24)
self.addCommutator = QtWidgets.QPushButton(parent=self.stackedCommutator)
self.addCommutator.setGeometry(QtCore.QRect(0, 60, 75, 23))
self.addCommutator.setObjectName("addCommutator")
self.stackedWidget.addWidget(self.stackedCommutator)
self.stackedPowerUnit = QtWidgets.QWidget()
self.stackedPowerUnit.setObjectName("stackedPowerUnit")
self.TVPowerUnit = QtWidgets.QTableView(parent=self.stackedPowerUnit)
self.TVPowerUnit.setGeometry(QtCore.QRect(0, 160, 971, 531))
self.TVPowerUnit.setObjectName("TVPowerUnit")
self.label_title_provider_4 =
QtWidgets.QLabel(parent=self.stackedPowerUnit)
self.label_title_provider_4.setGeometry(QtCore.QRect(0, 0, 558, 23))
self.label_title_provider_4.setObjectName("label_title_provider_4")
self.addPowerUnit = QtWidgets.QPushButton(parent=self.stackedPowerUnit)
self.addPowerUnit.setGeometry(QtCore.QRect(0, 60, 75, 23))
self.addPowerUnit.setObjectName("addPowerUnit")
self.layoutWidget_3 = QtWidgets.QWidget(parent=self.stackedPowerUnit)
self.layoutWidget_3.setGeometry(QtCore.QRect(2, 91, 569, 55))
self.layoutWidget_3.setObjectName("layoutWidget_3")
self.horizontalLayout_7 = QtWidgets.QHBoxLayout(self.layoutWidget_3)
self.horizontalLayout_7.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_7.setObjectName("horizontalLayout_7")
self.verticalLayout_25 = QtWidgets.QVBoxLayout()
self.verticalLayout_25.setObjectName("verticalLayout_25")
self.label_26 = QtWidgets.QLabel(parent=self.layoutWidget_3)
self.label_26.setMinimumSize(QtCore.QSize(150, 20))
self.label_26.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_26.setFont(font)

```

```

self.label_26.setObjectName("label_26")
self.verticalLayout_25.addWidget(self.label_26)
self.namePowerUnit = QtWidgets.QLineEdit(parent=self.layoutWidget_3)
self.namePowerUnit.setMinimumSize(QtCore.QSize(150, 20))
self.namePowerUnit.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.namePowerUnit.setFont(font)
self.namePowerUnit.setObjectName("namePowerUnit")
self.verticalLayout_25.addWidget(self.namePowerUnit)
self.horizontalLayout_7.addLayout(self.verticalLayout_25)
self.verticalLayout_26 = QtWidgets.QVBoxLayout()
self.verticalLayout_26.setObjectName("verticalLayout_26")
self.label_27 = QtWidgets.QLabel(parent=self.layoutWidget_3)
self.label_27.setMinimumSize(QtCore.QSize(150, 20))
self.label_27.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_27.setFont(font)
self.label_27.setObjectName("label_27")
self.verticalLayout_26.addWidget(self.label_27)
self.providerIDPowerUnit =
QtWidgets.QLineEdit(parent=self.layoutWidget_3)
self.providerIDPowerUnit.setMinimumSize(QtCore.QSize(150, 20))
self.providerIDPowerUnit.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.providerIDPowerUnit.setFont(font)
self.providerIDPowerUnit.setObjectName("providerIDPowerUnit")
self.verticalLayout_26.addWidget(self.providerIDPowerUnit)
self.horizontalLayout_7.addLayout(self.verticalLayout_26)
self.verticalLayout_27 = QtWidgets.QVBoxLayout()
self.verticalLayout_27.setObjectName("verticalLayout_27")
self.label_28 = QtWidgets.QLabel(parent=self.layoutWidget_3)
self.label_28.setMinimumSize(QtCore.QSize(150, 20))
self.label_28.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_28.setFont(font)
self.label_28.setObjectName("label_28")
self.verticalLayout_27.addWidget(self.label_28)

```



```

self.countPowerUnit = QtWidgets.QLineEdit(parent=self.layoutWidget_3)
self.countPowerUnit.setMinimumSize(QtCore.QSize(150, 20))
self.countPowerUnit.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.countPowerUnit.setFont(font)
self.countPowerUnit.setObjectName("countPowerUnit")
self.verticalLayout_27.addWidget(self.countPowerUnit)
self.horizontalLayout_7.addLayout(self.verticalLayout_27)
self.verticalLayout_28 = QtWidgets.QVBoxLayout()
self.verticalLayout_28.setObjectName("verticalLayout_28")
self.label_29 = QtWidgets.QLabel(parent=self.layoutWidget_3)
self.label_29.setMinimumSize(QtCore.QSize(91, 21))
self.label_29.setText("")
self.label_29.setObjectName("label_29")
self.verticalLayout_28.addWidget(self.label_29)
self.okPowerUnit = QtWidgets.QPushButton(parent=self.layoutWidget_3)
self.okPowerUnit.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()
font.setPointSize(10)
self.okPowerUnit.setFont(font)
self.okPowerUnit.setObjectName("okPowerUnit")
self.verticalLayout_28.addWidget(self.okPowerUnit)
self.horizontalLayout_7.addLayout(self.verticalLayout_28)
self.stackedWidget.addWidget(self.stackedPowerUnit)
self.stackedProduct = QtWidgets.QWidget()
self.stackedProduct.setObjectName("stackedProduct")
self.label_title_provider_6 =
QtWidgets.QLabel(parent=self.stackedProduct)
self.label_title_provider_6.setGeometry(QtCore.QRect(0, 0, 558, 23))
self.label_title_provider_6.setObjectName("label_title_provider_6")
self.TVProduct = QtWidgets.QTableView(parent=self.stackedProduct)
self.TVProduct.setGeometry(QtCore.QRect(0, 160, 971, 531))
self.TVProduct.setObjectName("TVProduct")
self.addProduct = QtWidgets.QPushButton(parent=self.stackedProduct)
self.addProduct.setGeometry(QtCore.QRect(0, 60, 75, 23))
self.addProduct.setObjectName("addProduct")
self.widget2 = QtWidgets.QWidget(parent=self.stackedProduct)
self.widget2.setGeometry(QtCore.QRect(2, 91, 727, 55))
self.widget2.setObjectName("widget2")
self.horizontalLayout_8 = QtWidgets.QHBoxLayout(self.widget2)

```

```

self.horizontalLayout_8.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_8.setObjectName("horizontalLayout_8")
self.verticalLayout_29 = QtWidgets.QVBoxLayout()
self.verticalLayout_29.setObjectName("verticalLayout_29")
self.label_30 = QtWidgets.QLabel(parent=self.widget2)
self.label_30.setMinimumSize(QtCore.QSize(150, 20))
self.label_30.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_30.setFont(font)
self.label_30.setObjectName("label_30")
self.verticalLayout_29.addWidget(self.label_30)
self.nameProduct = QtWidgets.QLineEdit(parent=self.widget2)
self.nameProduct.setMinimumSize(QtCore.QSize(150, 20))
self.nameProduct.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.nameProduct.setFont(font)
self.nameProduct.setObjectName("nameProduct")
self.verticalLayout_29.addWidget(self.nameProduct)
self.horizontalLayout_8.addLayout(self.verticalLayout_29)
self.verticalLayout_30 = QtWidgets.QVBoxLayout()
self.verticalLayout_30.setObjectName("verticalLayout_30")
self.label_31 = QtWidgets.QLabel(parent=self.widget2)
self.label_31.setMinimumSize(QtCore.QSize(150, 20))
self.label_31.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_31.setFont(font)
self.label_31.setObjectName("label_31")
self.verticalLayout_30.addWidget(self.label_31)
self.lockerIDProduct = QtWidgets.QLineEdit(parent=self.widget2)
self.lockerIDProduct.setMinimumSize(QtCore.QSize(150, 20))
self.lockerIDProduct.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.lockerIDProduct.setFont(font)
self.lockerIDProduct.setObjectName("lockerIDProduct")
self.verticalLayout_30.addWidget(self.lockerIDProduct)
self.horizontalLayout_8.addLayout(self.verticalLayout_30)
self.verticalLayout_31 = QtWidgets.QVBoxLayout()

```



```

self.verticalLayout_31.setObjectName("verticalLayout_31")
self.label_32 = QtWidgets.QLabel(parent=self.widget2)
self.label_32.setMinimumSize(QtCore.QSize(150, 20))
self.label_32.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_32.setFont(font)
self.label_32.setObjectName("label_32")
self.verticalLayout_31.addWidget(self.label_32)
self.commutatorIDProduct = QtWidgets.QLineEdit(parent=self.widget2)
self.commutatorIDProduct.setMinimumSize(QtCore.QSize(150, 20))
self.commutatorIDProduct.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.commutatorIDProduct.setFont(font)
self.commutatorIDProduct.setObjectName("commutatorIDProduct")
self.verticalLayout_31.addWidget(self.commutatorIDProduct)
self.horizontalLayout_8.addLayout(self.verticalLayout_31)
self.verticalLayout_32 = QtWidgets.QVBoxLayout()
self.verticalLayout_32.setObjectName("verticalLayout_32")
self.label_33 = QtWidgets.QLabel(parent=self.widget2)
self.label_33.setMinimumSize(QtCore.QSize(150, 20))
self.label_33.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_33.setFont(font)
self.label_33.setObjectName("label_33")
self.verticalLayout_32.addWidget(self.label_33)
self.powerUnitIDProduct = QtWidgets.QLineEdit(parent=self.widget2)
self.powerUnitIDProduct.setMinimumSize(QtCore.QSize(150, 20))
self.powerUnitIDProduct.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.powerUnitIDProduct.setFont(font)
self.powerUnitIDProduct.setObjectName("powerUnitIDProduct")
self.verticalLayout_32.addWidget(self.powerUnitIDProduct)
self.horizontalLayout_8.addLayout(self.verticalLayout_32)
self.verticalLayout_34 = QtWidgets.QVBoxLayout()
self.verticalLayout_34.setObjectName("verticalLayout_34")
self.label_35 = QtWidgets.QLabel(parent=self.widget2)
self.label_35.setMinimumSize(QtCore.QSize(91, 21))

```

```

self.label_35.setText("")
self.label_35.setObjectName("label_35")
self.verticalLayout_34.addWidget(self.label_35)
self.okProduct = QtWidgets.QPushButton(parent=self.widget2)
self.okProduct.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()
font.setPointSize(10)
self.okProduct.setFont(font)
self.okProduct.setObjectName("okProduct")
self.verticalLayout_34.addWidget(self.okProduct)
self.horizontalLayout_8.addLayout(self.verticalLayout_34)
self.stackedWidget.addWidget(self.stackedProduct)
self.stackedSale = QtWidgets.QWidget()
self.stackedSale.setObjectName("stackedSale")
self.label_title_provider_7 = QtWidgets.QLabel(parent=self.stackedSale)
self.label_title_provider_7.setGeometry(QtCore.QRect(0, 0, 558, 23))
self.label_title_provider_7.setObjectName("label_title_provider_7")
self.TVSale = QtWidgets.QTableView(parent=self.stackedSale)
self.TVSale.setGeometry(QtCore.QRect(0, 160, 971, 531))
self.TVSale.setObjectName("TVSale")
self.addSale = QtWidgets.QPushButton(parent=self.stackedSale)
self.addSale.setGeometry(QtCore.QRect(-2, 59, 75, 23))
self.addSale.setObjectName("addSale")
self.layoutWidget_4 = QtWidgets.QWidget(parent=self.stackedSale)
self.layoutWidget_4.setGeometry(QtCore.QRect(0, 90, 727, 55))
self.layoutWidget_4.setObjectName("layoutWidget_4")
self.horizontalLayout_9 = QtWidgets.QHBoxLayout(self.layoutWidget_4)
self.horizontalLayout_9.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_9.setObjectName("horizontalLayout_9")
self.verticalLayout_33 = QtWidgets.QVBoxLayout()
self.verticalLayout_33.setObjectName("verticalLayout_33")
self.label_34 = QtWidgets.QLabel(parent=self.layoutWidget_4)
self.label_34.setMinimumSize(QtCore.QSize(150, 20))
self.label_34.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_34.setFont(font)
self.label_34.setObjectName("label_34")
self.verticalLayout_33.addWidget(self.label_34)
self.productIDSale = QtWidgets.QLineEdit(parent=self.layoutWidget_4)
self.productIDSale.setMinimumSize(QtCore.QSize(150, 20))

```

```

self.productIDSale.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.productIDSale.setFont(font)
self.productIDSale.setObjectName("productIDSale")
self.verticalLayout_33.addWidget(self.productIDSale)
self.horizontalLayout_9.addLayout(self.verticalLayout_33)
self.verticalLayout_35 = QtWidgets.QVBoxLayout()
self.verticalLayout_35.setObjectName("verticalLayout_35")
self.label_36 = QtWidgets.QLabel(parent=self.layoutWidget_4)
self.label_36.setMinimumSize(QtCore.QSize(150, 20))
self.label_36.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_36.setFont(font)
self.label_36.setObjectName("label_36")
self.verticalLayout_35.addWidget(self.label_36)
self.buyerIDSale = QtWidgets.QLineEdit(parent=self.layoutWidget_4)
self.buyerIDSale.setMinimumSize(QtCore.QSize(150, 20))
self.buyerIDSale.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.buyerIDSale.setFont(font)
self.buyerIDSale.setObjectName("buyerIDSale")
self.verticalLayout_35.addWidget(self.buyerIDSale)
self.horizontalLayout_9.addLayout(self.verticalLayout_35)
self.verticalLayout_36 = QtWidgets.QVBoxLayout()
self.verticalLayout_36.setObjectName("verticalLayout_36")
self.label_37 = QtWidgets.QLabel(parent=self.layoutWidget_4)
self.label_37.setMinimumSize(QtCore.QSize(150, 20))
self.label_37.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_37.setFont(font)
self.label_37.setObjectName("label_37")
self.verticalLayout_36.addWidget(self.label_37)
self.countSale = QtWidgets.QLineEdit(parent=self.layoutWidget_4)
self.countSale.setMinimumSize(QtCore.QSize(150, 20))
self.countSale.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)

```

```

self.countSale.setFont(font)
self.countSale.setObjectName("countSale")
self.verticalLayout_36.addWidget(self.countSale)
self.horizontalLayout_9.addLayout(self.verticalLayout_36)
self.verticalLayout_37 = QtWidgets.QVBoxLayout()
self.verticalLayout_37.setObjectName("verticalLayout_37")
self.label_38 = QtWidgets.QLabel(parent=self.layoutWidget_4)
self.label_38.setMinimumSize(QtCore.QSize(150, 20))
self.label_38.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_38.setFont(font)
self.label_38.setObjectName("label_38")
self.verticalLayout_37.addWidget(self.label_38)
self.costSale = QtWidgets.QLineEdit(parent=self.layoutWidget_4)
self.costSale.setMinimumSize(QtCore.QSize(150, 20))
self.costSale.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.costSale.setFont(font)
self.costSale.setObjectName("costSale")
self.verticalLayout_37.addWidget(self.costSale)
self.horizontalLayout_9.addLayout(self.verticalLayout_37)
self.verticalLayout_38 = QtWidgets.QVBoxLayout()
self.verticalLayout_38.setObjectName("verticalLayout_38")
self.label_39 = QtWidgets.QLabel(parent=self.layoutWidget_4)
self.label_39.setMinimumSize(QtCore.QSize(91, 21))
self.label_39.setText("")
self.label_39.setObjectName("label_39")
self.verticalLayout_38.addWidget(self.label_39)
self.okSale = QtWidgets.QPushButton(parent=self.layoutWidget_4)
self.okSale.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()
font.setPointSize(10)
self.okSale.setFont(font)
self.okSale.setObjectName("okSale")
self.verticalLayout_38.addWidget(self.okSale)
self.horizontalLayout_9.addLayout(self.verticalLayout_38)
self.stackedWidget.addWidget(self.stackedSale)
self.stackedBuyer = QtWidgets.QWidget()
self.stackedBuyer.setObjectName("stackedBuyer")

```

```

self.TVBuyer = QtWidgets.QTableView(parent=self.stackedBuyer)
self.TVBuyer.setGeometry(QtCore.QRect(0, 160, 971, 531))
self.TVBuyer.setObjectName("TVBuyer")
self.label_title_provider_5 = QtWidgets.QLabel(parent=self.stackedBuyer)
self.label_title_provider_5.setGeometry(QtCore.QRect(0, 0, 558, 23))
self.label_title_provider_5.setObjectName("label_title_provider_5")
self.addBuyer = QtWidgets.QPushButton(parent=self.stackedBuyer)
self.addBuyer.setGeometry(QtCore.QRect(0, 60, 75, 23))
self.addBuyer.setObjectName("addBuyer")
self.layoutWidget_5 = QtWidgets.QWidget(parent=self.stackedBuyer)
self.layoutWidget_5.setGeometry(QtCore.QRect(0, 90, 887, 55))
self.layoutWidget_5.setObjectName("layoutWidget_5")
self.horizontalLayout_10 = QtWidgets.QHBoxLayout(self.layoutWidget_5)
self.horizontalLayout_10.setContentsMargins(0, 0, 0, 0)
self.horizontalLayout_10.setObjectName("horizontalLayout_10")
self.horizontalLayout_11 = QtWidgets.QHBoxLayout()
self.horizontalLayout_11.setObjectName("horizontalLayout_11")
self.verticalLayout_39 = QtWidgets.QVBoxLayout()
self.verticalLayout_39.setObjectName("verticalLayout_39")
self.label_40 = QtWidgets.QLabel(parent=self.layoutWidget_5)
self.label_40.setMinimumSize(QtCore.QSize(150, 20))
self.label_40.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_40.setFont(font)
self.label_40.setObjectName("label_40")
self.verticalLayout_39.addWidget(self.label_40)
self.innBuyer = QtWidgets.QLineEdit(parent=self.layoutWidget_5)
self.innBuyer.setMinimumSize(QtCore.QSize(150, 20))
self.innBuyer.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.innBuyer.setFont(font)
self.innBuyer.setObjectName("innBuyer")
self.verticalLayout_39.addWidget(self.innBuyer)
self.horizontalLayout_11.addLayout(self.verticalLayout_39)
self.verticalLayout_40 = QtWidgets.QVBoxLayout()
self.verticalLayout_40.setObjectName("verticalLayout_40")
self.label_41 = QtWidgets.QLabel(parent=self.layoutWidget_5)
self.label_41.setMinimumSize(QtCore.QSize(150, 20))
self.label_41.setMaximumSize(QtCore.QSize(150, 20))

```

```

font = QtGui.QFont()
font.setPointSize(10)
self.label_41.setFont(font)
self.label_41.setObjectName("label_41")
self.verticalLayout_40.addWidget(self.label_41)
self.fnameBuyer = QtWidgets.QLineEdit(parent=self.layoutWidget_5)
self.fnameBuyer.setMinimumSize(QtCore.QSize(150, 20))
self.fnameBuyer.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.fnameBuyer.setFont(font)
self.fnameBuyer.setObjectName("fnameBuyer")
self.verticalLayout_40.addWidget(self.fnameBuyer)
self.horizontalLayout_11.addLayout(self.verticalLayout_40)
self.verticalLayout_41 = QtWidgets.QVBoxLayout()
self.verticalLayout_41.setObjectName("verticalLayout_41")
self.label_42 = QtWidgets.QLabel(parent=self.layoutWidget_5)
self.label_42.setMinimumSize(QtCore.QSize(150, 20))
self.label_42.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_42.setFont(font)
self.label_42.setObjectName("label_42")
self.verticalLayout_41.addWidget(self.label_42)
self.lnameBuyer = QtWidgets.QLineEdit(parent=self.layoutWidget_5)
self.lnameBuyer.setMinimumSize(QtCore.QSize(150, 20))
self.lnameBuyer.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.lnameBuyer.setFont(font)
self.lnameBuyer.setObjectName("lnameBuyer")
self.verticalLayout_41.addWidget(self.lnameBuyer)
self.horizontalLayout_11.addLayout(self.verticalLayout_41)
self.verticalLayout_42 = QtWidgets.QVBoxLayout()
self.verticalLayout_42.setObjectName("verticalLayout_42")
self.label_43 = QtWidgets.QLabel(parent=self.layoutWidget_5)
self.label_43.setMinimumSize(QtCore.QSize(150, 20))
self.label_43.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_43.setFont(font)

```

```

self.label_43.setObjectName("label_43")
self.verticalLayout_42.addWidget(self.label_43)
self.patronymicBuyer = QtWidgets.QLineEdit(parent=self.layoutWidget_5)
self.patronymicBuyer.setMinimumSize(QtCore.QSize(150, 20))
self.patronymicBuyer.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.patronymicBuyer.setFont(font)
self.patronymicBuyer.setObjectName("patronymicBuyer")
self.verticalLayout_42.addWidget(self.patronymicBuyer)
self.horizontalLayout_11.addLayout(self.verticalLayout_42)
self.verticalLayout_43 = QtWidgets.QVBoxLayout()
self.verticalLayout_43.setObjectName("verticalLayout_43")
self.label_44 = QtWidgets.QLabel(parent=self.layoutWidget_5)
self.label_44.setMinimumSize(QtCore.QSize(150, 20))
self.label_44.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.label_44.setFont(font)
self.label_44.setObjectName("label_44")
self.verticalLayout_43.addWidget(self.label_44)
self.telBuyer = QtWidgets.QLineEdit(parent=self.layoutWidget_5)
self.telBuyer.setMinimumSize(QtCore.QSize(150, 20))
self.telBuyer.setMaximumSize(QtCore.QSize(150, 20))
font = QtGui.QFont()
font.setPointSize(10)
self.telBuyer.setFont(font)
self.telBuyer.setObjectName("telBuyer")
self.verticalLayout_43.addWidget(self.telBuyer)
self.horizontalLayout_11.addLayout(self.verticalLayout_43)
self.horizontalLayout_10.addLayout(self.horizontalLayout_11)
self.verticalLayout_44 = QtWidgets.QVBoxLayout()
self.verticalLayout_44.setObjectName("verticalLayout_44")
self.label_45 = QtWidgets.QLabel(parent=self.layoutWidget_5)
self.label_45.setMinimumSize(QtCore.QSize(91, 21))
self.label_45.setText("")
self.label_45.setObjectName("label_45")
self.verticalLayout_44.addWidget(self.label_45)
self.okBuyer = QtWidgets.QPushButton(parent=self.layoutWidget_5)
self.okBuyer.setMinimumSize(QtCore.QSize(91, 21))
font = QtGui.QFont()

```



```

font.setPointSize(10)
self.okBuyer.setFont(font)
self.okBuyer.setObjectName("okBuyer")
self.verticalLayout_44.addWidget(self.okBuyer)
self.horizontalLayout_10.addLayout(self.verticalLayout_44)
self.stackedWidget.addWidget(self.stackedBuyer)
self.label = QtWidgets.QLabel(parent=Form)
self.label.setGeometry(QtCore.QRect(0, 0, 1251, 741))
self.label.setStyleSheet("background-color: rgb(107, 171, 255);")
self.label.setText("")
self.label.setObjectName("label")
self.label.raise_()
self.menu.raise_()
self.stackedWidget.raise_()

self.retranslateUi(Form)
self.stackedWidget.setCurrentIndex(5)
QtCore.QMetaObject.connectSlotsByName(Form)

def retranslateUi(self, Form):
    _translate = QtCore.QCoreApplication.translate
    Form.setWindowTitle(_translate("Form", "Form"))
    __sortingEnabled = self.menu.isSortingEnabled()
    self.menu.setSortingEnabled(False)
    item = self.menu.item(0)
    item.setText(_translate("Form", "Provider (Поставщики)"))
    item = self.menu.item(1)
    item.setText(_translate("Form", "Locker (Шкафы)"))
    item = self.menu.item(2)
    item.setText(_translate("Form", "Commutator (Коммутаторы)"))
    item = self.menu.item(3)
    item.setText(_translate("Form", "PowerUnit (БП/ИБП)"))
    item = self.menu.item(4)
    item.setText(_translate("Form", "Product (Товары)"))
    item = self.menu.item(5)
    item.setText(_translate("Form", "Sale (Продажи)"))
    item = self.menu.item(6)
    item.setText(_translate("Form", "Buyer (Покупатели)"))
    self.menu.setSortingEnabled(__sortingEnabled)

```



```

        self.label_title_provider.setText(_translate("Form",
"<html><head></head><body><p><span style=\" font-size:14pt;
font-weight:600;\>Таблица: Поставщики</span></p></body></html>"))
        self.addProvider.setText(_translate("Form", "Добавить"))
        self.label_2.setText(_translate("Form", "ИНН"))
        self.label_3.setText(_translate("Form", "Имя"))
        self.label_4.setText(_translate("Form", "Фамилия"))
        self.label_5.setText(_translate("Form", "Отчество"))
        self.label_6.setText(_translate("Form", "Телефон"))
        self.okProvider.setText(_translate("Form", "ОК"))
        self.label_title_provider_2.setText(_translate("Form",
"<html><head></head><body><p><span style=\" font-size:14pt;
font-weight:600;\>Таблица: Шкафы</span></p></body></html>"))
        self.addLocker.setText(_translate("Form", "Добавить"))
        self.label_18.setText(_translate("Form", "Номенклатура"))
        self.label_19.setText(_translate("Form", "Поставщик"))
        self.label_20.setText(_translate("Form", "Кол-во"))
        self.okLocker.setText(_translate("Form", "ОК"))
        self.label_title_provider_3.setText(_translate("Form",
"<html><head></head><body><p><span style=\" font-size:14pt;
font-weight:600;\>Таблица: Коммутаторы</span></p></body></html>"))
        self.label_21.setText(_translate("Form", "Номенклатура"))
        self.label_22.setText(_translate("Form", "Поставщик"))
        self.label_24.setText(_translate("Form", "Кол-во"))
        self.okCommutator.setText(_translate("Form", "ОК"))
        self.addCommutator.setText(_translate("Form", "Добавить"))
        self.label_title_provider_4.setText(_translate("Form",
"<html><head></head><body><p><span style=\" font-size:14pt;
font-weight:600;\>Таблица: Блоки питания</span></p></body></html>"))
        self.addPowerUnit.setText(_translate("Form", "Добавить"))
        self.label_26.setText(_translate("Form", "Номенклатура"))
        self.label_27.setText(_translate("Form", "Поставщик"))
        self.label_28.setText(_translate("Form", "Кол-во"))
        self.okPowerUnit.setText(_translate("Form", "ОК"))
        self.label_title_provider_6.setText(_translate("Form",
"<html><head></head><body><p><span style=\" font-size:14pt;
font-weight:600;\>Таблица: Товары</span></p></body></html>"))
        self.addProduct.setText(_translate("Form", "Добавить"))
        self.label_30.setText(_translate("Form", "Номенклатура"))
        self.label_31.setText(_translate("Form", "Шкаф"))
        self.label_32.setText(_translate("Form", "Коммутатор"))

```

```

        self.label_33.setText(_translate("Form", "Блок питания"))
        self.okProduct.setText(_translate("Form", "ОК"))
        self.label_title_provider_7.setText(_translate("Form",
"<html><head/><body><p><span style=\" font-size:14pt;
font-weight:600;\">Таблица: Продажи</span></p></body></html>"))
        self.addSale.setText(_translate("Form", "Добавить"))
        self.label_34.setText(_translate("Form", "Номенклатура"))
        self.label_36.setText(_translate("Form", "Покупатель"))
        self.label_37.setText(_translate("Form", "Кол-во"))
        self.label_38.setText(_translate("Form", "Цена"))
        self.okSale.setText(_translate("Form", "ОК"))
        self.label_title_provider_5.setText(_translate("Form",
"<html><head/><body><p><span style=\" font-size:14pt;
font-weight:600;\">Таблица: Покупатели</span></p></body></html>"))
        self.addBuyer.setText(_translate("Form", "Добавить"))
        self.label_40.setText(_translate("Form", "ИНН"))
        self.label_41.setText(_translate("Form", "Имя"))
        self.label_42.setText(_translate("Form", "Фамилия"))
        self.label_43.setText(_translate("Form", "Отчество"))
        self.label_44.setText(_translate("Form", "Телефон"))
        self.okBuyer.setText(_translate("Form", "ОК"))

if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
    Form = QtWidgets.QWidget()
    ui = Ui_Form()
    ui.setupUi(Form)
    Form.show()
    sys.exit(app.exec())

```

createDB.py

```

import sqlite3

db = sqlite3.connect('#dataBase.db')
cursor = db.cursor()

```

```

cursor.execute('''
create table if not exists provider(
    id integer primary key,
    inn integer not null,
    fname varchar(20) not null,
    lname varchar(20) not null,
    patronymic varchar(20) not null,
    tel integer not null
);
''')
cursor.execute('''
create table if not exists locker(
    id integer primary key,
    name varchar(20) not null,
    providerID integer references provider(id) not null,
    count integer not null
);
''')
cursor.execute('''
create table if not exists commutator(
    id integer primary key,
    name varchar(20) not null,
    providerID integer references provider(id) not null,
    count integer not null
);
''')
cursor.execute('''
create table if not exists powerUnit(
    id integer primary key,
    name varchar(20) not null,
    providerID integer references provider(id) not null,
    count integer not null
);
''')
cursor.execute('''
create table if not exists buyer(
    id integer primary key,
    inn varchar(11) not null,
    fname varchar(20) not null,
    lname varchar(20) not null,
    patronymic varchar(20) not null,

```

```

        tel varchar(11) not null
    );
'''
cursor.execute('''
create table if not exists product(
    id integer primary key,
    name varchar(20) not null,
    lockerID integer references locker(id) not null,
    commutatorID integer references commutator(id) not null,
    powerUnitID integer references powerUnit(id) not null
);
'''
cursor.execute('''
create table if not exists sale(
    id integer primary key,
    productID integer references product(id) not null,
    buyerID integer references buyer(id) not null,
    count integer not null,
    cost float not null
);
'''

db.commit()
db.close()

print('БД создана')

```

insertDB.py

```

import sqlite3

db = sqlite3.connect('#dataBase.db')
cursor = db.cursor()

```

```

insertProvider = [
    (1, '12345678901', 'Иван', 'Иванов', 'Иванович', '89012345678'),
    (2, '23456789012', 'Петр', 'Петров', 'Петрович', '89023456789'),
    (3, '34567890123', 'Анна', 'Сидорова', 'Александровна', '89034567890'),
    (4, '45678901234', 'Мария', 'Кузнецова', 'Игоревна', '89045678901'),
    (5, '56789012345', 'Алексей', 'Смирнов', 'Сергеевич', '89056789012'),
    (6, '67890123456', 'Ольга', 'Морозова', 'Петровна', '89067890123'),
    (7, '78901234567', 'Дмитрий', 'Волков', 'Андреевич', '89078901234'),
    (8, '89012345678', 'Елена', 'Лебедева', 'Викторовна', '89089012345'),
    (9, '90123456789', 'Сергей', 'Козлов', 'Дмитриевич', '89090123456'),
    (10, '10234567890', 'Наталья', 'Новикова', 'Владимировна', '89101234567'),
]

insertQuery = """
INSERT INTO provider (id, inn, fname, lname, patronymic, tel)
VALUES (?, ?, ?, ?, ?, ?)
"""

for insert in insertProvider:
    cursor.execute(insertQuery, insert)

insertLocker = [
    (1, 'Шкаф-10U', 5, 50),
    (2, 'Шкаф-20U', 1, 45),
    (3, 'Шкаф-42U', 8, 30),
    (4, 'Шкаф-настенный', 3, 100),
    (5, 'Стойка-открытая', 10, 75),
    (6, 'Шкаф-IP55', 2, 20),
    (7, 'Шкаф-серверный-угл', 4, 15),
    (8, 'Шкаф-телеком-малый', 9, 60),
    (9, 'Шкаф-антивандальный', 6, 10),
    (10, 'Шкаф-уличный-больш', 7, 5)
]

insertQuery = '''
INSERT INTO locker (id, name, providerID, count)
VALUES (?, ?, ?, ?)
'''

for insert in insertLocker:

```

```

        cursor.execute(insertQuery, insert)

insertCommutator = [
    (1, 'Cisco-2960-8TC', 7, 200),
    (2, 'Mikrotik-CRS328', 3, 150),
    (3, 'Huawei-S5700', 9, 80),
    (4, 'DLink-DGS-1210', 1, 300),
    (5, 'TP-Link-SG108', 5, 500),
    (6, 'Cisco-Nexus-9K', 2, 20),
    (7, 'Juniper-EX4300', 8, 45),
    (8, 'Arista-7050S', 4, 30),
    (9, 'Extreme-X440-G2', 10, 60),
    (10, 'Zyxel-GS1920', 6, 120)
]

insertQuery = '''
INSERT INTO commutator (id, name, providerID, count)
VALUES (?, ?, ?, ?)
'''

for insert in insertCommutator:
    cursor.execute(insertQuery, insert)

insertPowerUnit = [
    (1, 'UPS-APC-Smart-750', 2, 90),
    (2, 'UPS-Eaton-5PX', 4, 70),
    (3, 'PDU-8-розеток-C13', 6, 400),
    (4, 'БП-серверный-750W', 1, 150),
    (5, 'БП-серверный-1200W', 9, 110),
    (6, 'UPS-CyberPower-1500', 3, 60),
    (7, 'PDU-управляемый-16A', 8, 50),
    (8, 'БП-Cisco-PWR-C2-1025', 5, 80),
    (9, 'Стабилизатор-220V', 10, 200),
    (10, 'Дизель-генератор-5kW', 7, 5)
]

insertQuery = '''
INSERT INTO powerUnit (id, name, providerID, count)
VALUES (?, ?, ?, ?)
'''

```

```

'''
for insert in insertPowerUnit:
    cursor.execute(insertQuery, insert)

insertProduct = [
    (1, 'Сборка-Офис-Мин', 1, 1, 1),
    (2, 'Сборка-Офис-Станд', 2, 2, 2),
    (3, 'Сборка-ЦОД-База', 3, 3, 3),
    (4, 'Сборка-ЦОД-Продв', 3, 6, 6),
    (5, 'Сборка-Малый-Бизн', 4, 4, 4),
    (6, 'Сборка-Склад-Угл', 5, 5, 5),
    (7, 'Сборка-Улица-IP55', 6, 9, 9),
    (8, 'Сборка-Телеком-ЭКО', 8, 5, 3),
    (9, 'Сборка-Серверная-S', 1, 1, 4),
    (10, 'Сборка-Серверная-L', 3, 3, 5)
]

insertQuery = '''
INSERT INTO product (id, name, lockerID, commutatorID, powerUnitID)
VALUES (?, ?, ?, ?, ?)
'''

for insert in insertProduct:
    cursor.execute(insertQuery, insert)

insertBuyer = [
    (1, 99001234567, 'Александр', 'Смирнов', 'Владимирович', 89201112233),
    (2, 99002345678, 'Екатерина', 'Петрова', 'Сергеевна', 89202223344),
    (3, 99003456789, 'Михаил', 'Соколов', 'Иванович', 89203334455),
    (4, 99004567890, 'Оксана', 'Федорова', 'Андреевна', 89204445566),
    (5, 99005678901, 'Дмитрий', 'Морозов', 'Павлович', 89205556677),
    (6, 99006789012, 'Светлана', 'Волкова', 'Алексеевна', 89206667788),
    (7, 99007890123, 'Игорь', 'Лебедев', 'Дмитриевич', 89207778899),
    (8, 99008901234, 'Татьяна', 'Козлова', 'Геннадьевна', 89208889900),
    (9, 99009012345, 'Павел', 'Семенов', 'Олегович', 89209990011),
    (10, 99010123456, 'Юлия', 'Голубева', 'Николаевна', 89210001122)
]

```

```

insertQuery = '''
INSERT INTO buyer (id, inn, fname, lname, patronymic, tel)
VALUES (?, ?, ?, ?, ?, ?)
'''

for insert in insertBuyer:
    cursor.execute(insertQuery, insert)

insetrSale = [
    (1, 1, 1, 2, 150000.00),
    (2, 3, 2, 1, 550000.00),
    (3, 5, 3, 5, 85000.50),
    (4, 2, 4, 1, 220000.00),
    (5, 10, 5, 1, 990000.00),
    (6, 8, 6, 3, 110000.25),
    (7, 4, 7, 2, 750000.00),
    (8, 9, 8, 4, 180000.00),
    (9, 6, 9, 1, 300000.00),
    (10, 7, 10, 1, 450000.00)
]

insertQuery = '''
INSERT INTO sale (id, productID, buyerID, count, cost)
VALUES (?, ?, ?, ?, ?)
'''

for insert in insetrSale:
    cursor.execute(insertQuery, insert)

db.commit()
db.close()
print('данные введены')

```

tables\provider.py


```

import sqlite3
from PyQt6 import QtCore, QtWidgets
from PyQt6.QtGui import QStandardItemModel, QStandardItem

def provider(ui):
    ui.okProvider.setVisible(False)
    global modelTableProvider
    modelTableProvider = QStandardItemModel()

def createTable():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
        select
            id,
            inn ИНН,
            fname Имя,
            lname Фамилия,
            patronymic Отчество,
            tel Телефон
        from
            provider;''')
    tableProvider = cursor.fetchall()
    db.commit()
    db.close()

    header = [f[0] for f in cursor.description] + ['']
    modelTableProvider.clear()
    modelTableProvider.setHorizontalHeaderLabels(header)
    for tab in tableProvider:
        items = []
        for item in tab:
            items.append(QStandardItem(str(item)))
        modelTableProvider.appendRow(items)

    ui.TVProvider.setModel(modelTableProvider)

```

```

        for num, tab in enumerate(tableProvider):
            but = QtWidgets.QPushButton('ИЗМЕНИТЬ')
            provider_id = tab[0]
            but.clicked.connect(lambda checked, n=provider_id :
change_provider(n))
            index = modelTableProvider.index(num, len(header)-1)
            ui.TVProvider.setIndexWidget(index, but)

def add_provider():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO provider (inn, fname, lname, patronymic, tel)
VALUES (?, ?, ?, ?, ?);
''', (
        ui.innProvider.text(),
        ui.fnameProvider.text(),
        ui.lnameProvider.text(),
        ui.patronymicProvider.text(),
        ui.telProvider.text()
    ))
    db.commit()
    db.close()
    createTable()

def change_provider(provider_id):
    ui.okProvider.setVisible(True)
    try:
        ui.okProvider.clicked.disconnect()
    except TypeError:
        pass
    ui.okProvider.clicked.connect(lambda : update_provider(provider_id))
    provider_id-=1
    ui.innProvider.setText(dataCell_provider(provider_id, 1)),
    ui.fnameProvider.setText(dataCell_provider(provider_id, 2)),
    ui.lnameProvider.setText(dataCell_provider(provider_id, 3)),
    ui.patronymicProvider.setText(dataCell_provider(provider_id, 4)),
    ui.telProvider.setText(dataCell_provider(provider_id, 5))

```

```

def update_provider(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
UPDATE
    provider
SET
    inn = ?,
    fname = ?,
    lname = ?,
    patronymic = ?,
    tel = ?
WHERE
    id = ?
''', (
        ui.innProvider.text(),
        ui.fnameProvider.text(),
        ui.lnameProvider.text(),
        ui.patronymicProvider.text(),
        ui.telProvider.text(),
        row
    ))
    db.commit()
    db.close()
    ui.innProvider.setText('')
    ui.fnameProvider.setText('')
    ui.lnameProvider.setText('')
    ui.patronymicProvider.setText('')
    ui.telProvider.setText('')
    createTable()
    ui.okProvider.setVisible(False)

def dataCell_provider(row, column):
    index = modelTableProvider.index(row, column)
    value = modelTableProvider.data(index,
QtCore.Qt.ItemDataRole.DisplayRole)
    return value

ui.addProvider.clicked.connect(add_provider)
createTable()

```

tables\buyer.py

```
import sqlite3
from PyQt6 import QtCore, QtWidgets
from PyQt6.QtGui import QStandardItemModel, QStandardItem

def buyer(ui):
    ui.okBuyer.setVisible(False)
    global modelTableBuyer
    modelTableBuyer = QStandardItemModel()

def createTable():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
        select
            id,
            inn ИНН,
            fname Имя,
            lname Фамилия,
            patronymic Отчество,
            tel Телефон
        from
            buyer;''')
    tableBuyer = cursor.fetchall()
    db.commit()
    db.close()

    header = [f[0] for f in cursor.description] + ['']
    modelTableBuyer.clear()
    modelTableBuyer.setHorizontalHeaderLabels(header)
    for tab in tableBuyer:
        items = []
        for item in tab:
            items.append(QStandardItem(str(item)))
```

```

        modelTableBuyer.appendRow(items)

    ui.TVBuyer.setModel(modelTableBuyer)

    for num, tab in enumerate(tableBuyer):
        but = QtWidgets.QPushButton('ИЗМЕНИТЬ')
        buyer_id = tab[0]
        but.clicked.connect(lambda checked, n=buyer_id : change_buyer(n))
        index = modelTableBuyer.index(num, len(header)-1)
        ui.TVBuyer.setIndexWidget(index, but)

def add_buyer():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO buyer (inn, fname, lname, patronymic, tel)
VALUES (?, ?, ?, ?, ?);
''', (
        ui.innBuyer.text(),
        ui.fnameBuyer.text(),
        ui.lnameBuyer.text(),
        ui.patronymicBuyer.text(),
        ui.telBuyer.text()
    ))
    db.commit()
    db.close()
    createTable()

def change_buyer(buyer_id):
    ui.okBuyer.setVisible(True)
    try:
        ui.okBuyer.clicked.disconnect()
    except TypeError:
        pass
    ui.okBuyer.clicked.connect(lambda : update_buyer(buyer_id))
    buyer_id-=1
    ui.innBuyer.setText(dataCell_buyer(buyer_id, 1)),
    ui.fnameBuyer.setText(dataCell_buyer(buyer_id, 2)),
    ui.lnameBuyer.setText(dataCell_buyer(buyer_id, 3)),
    ui.patronymicBuyer.setText(dataCell_buyer(buyer_id, 4)),

```

```

        ui.telBuyer.setText(dataCell_buyer(buyer_id, 5))

def update_buyer(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
UPDATE
    buyer
SET
    inn = ?,
    fname = ?,
    lname = ?,
    patronymic = ?,
    tel = ?
WHERE
    id = ?
''', (
        ui.innBuyer.text(),
        ui.fnameBuyer.text(),
        ui.lnameBuyer.text(),
        ui.patronymicBuyer.text(),
        ui.telBuyer.text(),
        row
    ))
    db.commit()
    db.close()
    ui.innBuyer.setText('')
    ui.fnameBuyer.setText('')
    ui.lnameBuyer.setText('')
    ui.patronymicBuyer.setText('')
    ui.telBuyer.setText('')
    createTable()
    ui.okBuyer.setVisible(False)

def dataCell_buyer(row, column):
    index = modelTableBuyer.index(row, column)
    value = modelTableBuyer.data(index, QtCore.Qt.ItemDataRole.DisplayRole)
    return value

ui.addBuyer.clicked.connect(add_buyer)

```

```
createTable()
```

tables\commutator.py

```
import sqlite3
from PyQt6 import QtCore, QtWidgets
from PyQt6.QtGui import QStandardItemModel, QStandardItem

def commutator(ui):

    ui.okCommutator.setVisible(False)
    global modelTableCommutator
    modelTableCommutator = QStandardItemModel()

    def createTable():
        db = sqlite3.connect('#dataBase.db')
        cursor = db.cursor()
        cursor.execute('''
            select
                id,
                name Коммутатор,
                providerID id_Поставщика,
                count Количество
            from
                commutator;''')
        tableCommutator = cursor.fetchall()
        db.commit()
        db.close()

        header = [f[0] for f in cursor.description] + ['']
        modelTableCommutator.clear()
        modelTableCommutator.setHorizontalHeaderLabels(header)
        for tab in tableCommutator:
            items = []
            for item in tab:
                items.append(QStandardItem(str(item)))
            modelTableCommutator.appendRow(items)

    ui.TVCommutator.setModel(modelTableCommutator)
```

```

        for num, tab in enumerate(tableCommutator):
            but = QtWidgets.QPushButton('ИЗМЕНИТЬ')
            commutator_id = tab[0]
            but.clicked.connect(lambda checked, n=commutator_id :
change_commutator(n))
            index = modelTableCommutator.index(num, len(header)-1)
            ui.TVCommutator.setIndexWidget(index, but)

def add_commutator():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO commutator (name, providerID, count)
VALUES (?, ?, ?);
''', (
        ui.nameCommutator.text(),
        ui.providerIDCommutator.text(),
        ui.countCommutator.text()
    ))
    db.commit()
    db.close()
    createTable()

def change_commutator(commutator_id):
    ui.okCommutator.setVisible(True)
    try:
        ui.okCommutator.clicked.disconnect()
    except TypeError:
        pass
    ui.okCommutator.clicked.connect(lambda :
update_commutator(commutator_id))
    commutator_id-=1
    ui.nameCommutator.setText(dataCell_commutator(commutator_id, 1)),
    ui.providerIDCommutator.setText(dataCell_commutator(commutator_id, 2)),
    ui.countCommutator.setText(dataCell_commutator(commutator_id, 3))

def update_commutator(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')

```



```

        cursor = db.cursor()
        cursor.execute('''
UPDATE
    commutator
SET
    name = ?,
    providerID = ?,
    count = ?
WHERE
    id = ?
''', (
        ui.nameCommutator.text(),
        ui.providerIDCommutator.text(),
        ui.countCommutator.text(),
        row
    ))
    db.commit()
    db.close()
    ui.nameCommutator.setText('')
    ui.providerIDCommutator.setText('')
    ui.countCommutator.setText('')
    createTable()
    ui.okCommutator.setVisible(False)

def dataCell_commutator(row, column):
    index = modelTableCommutator.index(row, column)
    value = modelTableCommutator.data(index,
QtCore.Qt.ItemDataRole.DisplayRole)
    return value

ui.addCommutator.clicked.connect(add_commutator)
createTable()

```

tables\locker.py

```

import sqlite3
from PyQt6 import QtCore, QtWidgets

```

```

from PyQt6.QtGui import QStandardItemModel, QStandardItem

def locker(ui):
    ui.okLocker.setVisible(False)
    global modelTableLocker
    modelTableLocker = QStandardItemModel()

    def createTable():
        db = sqlite3.connect('#dataBase.db')
        cursor = db.cursor()
        cursor.execute('''
            select
                id,
                name Шкаф,
                providerID id_Поставщика,
                count Количество
            from
                locker;''')
        tableLocker = cursor.fetchall()
        db.commit()
        db.close()

        header = [f[0] for f in cursor.description] + ['']
        modelTableLocker.clear()
        modelTableLocker.setHorizontalHeaderLabels(header)
        for tab in tableLocker:
            items = []
            for item in tab:
                items.append(QStandardItem(str(item)))
            modelTableLocker.appendRow(items)

    ui.TVLocker.setModel(modelTableLocker)

    for num, tab in enumerate(tableLocker):
        but = QtWidgets.QPushButton('Изменить')
        locker_id = tab[0]
        but.clicked.connect(lambda checked, n=locker_id : change_locker(n))
        index = modelTableLocker.index(num, len(header)-1)
        ui.TVLocker.setIndexWidget(index, but)

```

```

def add_locker():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO locker (name, providerID, count)
VALUES (?, ?, ?);
''', (
        ui.nameLocker.text(),
        ui.providerIDLocker.text(),
        ui.countLocker.text()
    ))
    db.commit()
    db.close()
    createTable()

def change_locker(locker_id):
    ui.okLocker.setVisible(True)
    try:
        ui.okLocker.clicked.disconnect()
    except TypeError:
        pass
    ui.okLocker.clicked.connect(lambda : update_locker(locker_id))
    locker_id-=1
    ui.nameLocker.setText(dataCell_locker(locker_id, 1)),
    ui.providerIDLocker.setText(dataCell_locker(locker_id, 2)),
    ui.countLocker.setText(dataCell_locker(locker_id, 3))

def update_locker(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
UPDATE
    locker
SET
    name = ?,
    providerID = ?,
    count = ?
WHERE

```

```

        id = ?
    ''' , (
        ui.nameLocker.text(),
        ui.providerIDLocker.text(),
        ui.countLocker.text(),
        row
    ))
    db.commit()
    db.close()
    ui.nameLocker.setText('')
    ui.providerIDLocker.setText('')
    ui.countLocker.setText('')
    createTable()
    ui.okLocker.setVisible(False)

def dataCell_locker(row, column):
    index = modelTableLocker.index(row, column)
    value = modelTableLocker.data(index, QtCore.Qt.ItemDataRole.DisplayRole)
    return value

ui.addLocker.clicked.connect(add_locker)
createTable()

```

tables\powerUnit.py

```

import sqlite3
from PyQt6 import QtCore, QtWidgets
from PyQt6.QtGui import QStandardItemModel, QStandardItem

def powerUnit(ui):
    ui.okPowerUnit.setVisible(False)
    global modelTablePowerUnit
    modelTablePowerUnit = QStandardItemModel()

    def createTable():
        db = sqlite3.connect('#dataBase.db')
        cursor = db.cursor()
        cursor.execute('

```

```

        select
            id,
            name БП,
            providerID id_Поставщика,
            count Количество
        from
            powerUnit;''')
tablePowerUnit = cursor.fetchall()
db.commit()
db.close()

header = [f[0] for f in cursor.description] + ['']
modelTablePowerUnit.clear()
modelTablePowerUnit.setHorizontalHeaderLabels(header)
for tab in tablePowerUnit:
    items = []
    for item in tab:
        items.append(QStandardItem(str(item)))
    modelTablePowerUnit.appendRow(items)

ui.TVPowerUnit.setModel(modelTablePowerUnit)

for num, tab in enumerate(tablePowerUnit):
    but = QtWidgets.QPushButton('ИЗМЕНИТЬ')
    powerUnit_id = tab[0]
    but.clicked.connect(lambda checked, n=powerUnit_id :
change_powerUnit(n))
    index = modelTablePowerUnit.index(num, len(header)-1)
    ui.TVPowerUnit.setIndexWidget(index, but)

def add_powerUnit():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO powerUnit (name, providerID, count)
VALUES (?, ?, ?);
''', (
    ui.namePowerUnit.text(),
    ui.providerIDPowerUnit.text(),
    ui.countPowerUnit.text()
))

```

```

        db.commit()
        db.close()
        createTable()

def change_powerUnit(powerUnit_id):
    ui.okPowerUnit.setVisible(True)
    try:
        ui.okPowerUnit.clicked.disconnect()
    except TypeError:
        pass
    ui.okPowerUnit.clicked.connect(lambda : update_powerUnit(powerUnit_id))
    powerUnit_id-=1
    ui.namePowerUnit.setText(dataCell_powerUnit(powerUnit_id, 1)),
    ui.providerIDPowerUnit.setText(dataCell_powerUnit(powerUnit_id, 2)),
    ui.countPowerUnit.setText(dataCell_powerUnit(powerUnit_id, 3))

def update_powerUnit(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
UPDATE
    powerUnit
SET
    name = ?,
    providerID = ?,
    count = ?
WHERE
    id = ?
''', (
        ui.namePowerUnit.text(),
        ui.providerIDPowerUnit.text(),
        ui.countPowerUnit.text(),
        row
    ))
    db.commit()
    db.close()
    ui.namePowerUnit.setText('')
    ui.providerIDPowerUnit.setText('')
    ui.countPowerUnit.setText('')

```

```

        createTable()
        ui.okPowerUnit.setVisible(False)

    def dataCell_powerUnit(row, column):
        index = modelTablePowerUnit.index(row, column)
        value = modelTablePowerUnit.data(index,
QtCore.Qt.ItemDataRole.DisplayRole)
        return value

    ui.addPowerUnit.clicked.connect(add_powerUnit)
    createTable()

```

tables\product.py

```

import sqlite3
from PyQt6 import QtCore, QtWidgets
from PyQt6.QtGui import QStandardItemModel, QStandardItem

def product(ui):

    ui.okProduct.setVisible(False)
    global modelTableProduct
    modelTableProduct = QStandardItemModel()

    def createTable():
        db = sqlite3.connect('#dataBase.db')
        cursor = db.cursor()
        cursor.execute('''
            select
                id,
                name Товар,
                lockerID id_Шкафа,
                commutatorID id_Коммутатора,
                powerUnitID id_БП
            from
                product;''')
        tableProduct = cursor.fetchall()
        db.commit()
        db.close()

```

```

header = [f[0] for f in cursor.description] + ['']
modelTableProduct.clear()
modelTableProduct.setHorizontalHeaderLabels(header)
for tab in tableProduct:
    items = []
    for item in tab:
        items.append(QStandardItem(str(item)))
    modelTableProduct.appendRow(items)

ui.TVProduct.setModel(modelTableProduct)

for num, tab in enumerate(tableProduct):
    but = QtWidgets.QPushButton('Изменить')
    product_id = tab[0]
    but.clicked.connect(lambda checked, n=product_id : change_product(n))
    index = modelTableProduct.index(num, len(header)-1)
    ui.TVProduct.setIndexWidget(index, but)

def add_product():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO product (name, lockerID, commutatorID, powerUnitID)
VALUES (?, ?, ?, ?);
''', (
    ui.nameProduct.text(),
    ui.lockerIDProduct.text(),
    ui.commutatorIDProduct.text(),
    ui.powerUnitIDProduct.text()
))
    db.commit()
    db.close()
    createTable()

def change_product(product_id):
    ui.okProduct.setVisible(True)
    try:
        ui.okProduct.clicked.disconnect()
    except TypeError:
        pass

```



```

        ui.okProduct.clicked.connect(lambda : update_product(product_id))
        product_id-=1
        ui.nameProduct.setText(dataCell_product(product_id, 1)),
        ui.lockerIDProduct.setText(dataCell_product(product_id, 2)),
        ui.commutatorIDProduct.setText(dataCell_product(product_id, 3)),
        ui.powerUnitIDProduct.setText(dataCell_product(product_id, 4))

def update_product(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute(''

UPDATE
    product
SET
    name = ?,
    lockerID = ?,
    commutatorID = ?,
    powerUnitID = ?
WHERE
    id = ?
'', (
        ui.nameProduct.text(),
        ui.lockerIDProduct.text(),
        ui.commutatorIDProduct.text(),
        ui.powerUnitIDProduct.text(),
        row
    ))
    db.commit()
    db.close()
    ui.nameProduct.setText('')
    ui.lockerIDProduct.setText('')
    ui.commutatorIDProduct.setText('')
    ui.powerUnitIDProduct.setText('')
    createTable()
    ui.okProduct.setVisible(False)

def dataCell_product(row, column):
    index = modelTableProduct.index(row, column)
    value = modelTableProduct.data(index, QtCore.Qt.ItemDataRole.DisplayRole)
    return value

```

```

ui.addProduct.clicked.connect(add_product)
createTable()

```

tables\sale.py

```

import sqlite3
from PyQt6 import QtCore, QtWidgets
from PyQt6.QtGui import QStandardItemModel, QStandardItem

def sale(ui):

    ui.okSale.setVisible(False)
    global modelTableSale
    modelTableSale = QStandardItemModel()

    def createTable():
        db = sqlite3.connect('#dataBase.db')
        cursor = db.cursor()
        cursor.execute('''
            select
                id,
                productID id_Товара,
                buyerID id_Покупателя,
                count Количество,
                cost Цена
            from
                sale;''')
        tableSale = cursor.fetchall()
        db.commit()
        db.close()

        header = [f[0] for f in cursor.description] + ['']
        modelTableSale.clear()
        modelTableSale.setHorizontalHeaderLabels(header)
        for tab in tableSale:
            items = []
            for item in tab:
                items.append(QStandardItem(str(item)))

```

```

        modelTableSale.appendRow(items)

    ui.TVSale.setModel(modelTableSale)

    for num, tab in enumerate(tableSale):
        but = QtWidgets.QPushButton('ИЗМЕНИТЬ')
        sale_id = tab[0]
        but.clicked.connect(lambda checked, n=sale_id : change_sale(n))
        index = modelTableSale.index(num, len(header)-1)
        ui.TVSale.setIndexWidget(index, but)

def add_sale():
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
INSERT INTO sale (productID, buyerID, count, cost)
VALUES (?, ?, ?, ?);
''', (
        ui.productIDSale.text(),
        ui.buyerIDSale.text(),
        ui.countSale.text(),
        ui.costSale.text()
    ))
    db.commit()
    db.close()
    createTable()

def change_sale(sale_id):
    ui.okSale.setVisible(True)
    try:
        ui.okSale.clicked.disconnect()
    except TypeError:
        pass
    ui.okSale.clicked.connect(lambda : update_sale(sale_id))
    sale_id-=1
    ui.productIDSale.setText(dataCell_sale(sale_id, 1)),
    ui.buyerIDSale.setText(dataCell_sale(sale_id, 2)),
    ui.countSale.setText(dataCell_sale(sale_id, 3)),
    ui.costSale.setText(dataCell_sale(sale_id, 4))

```

```

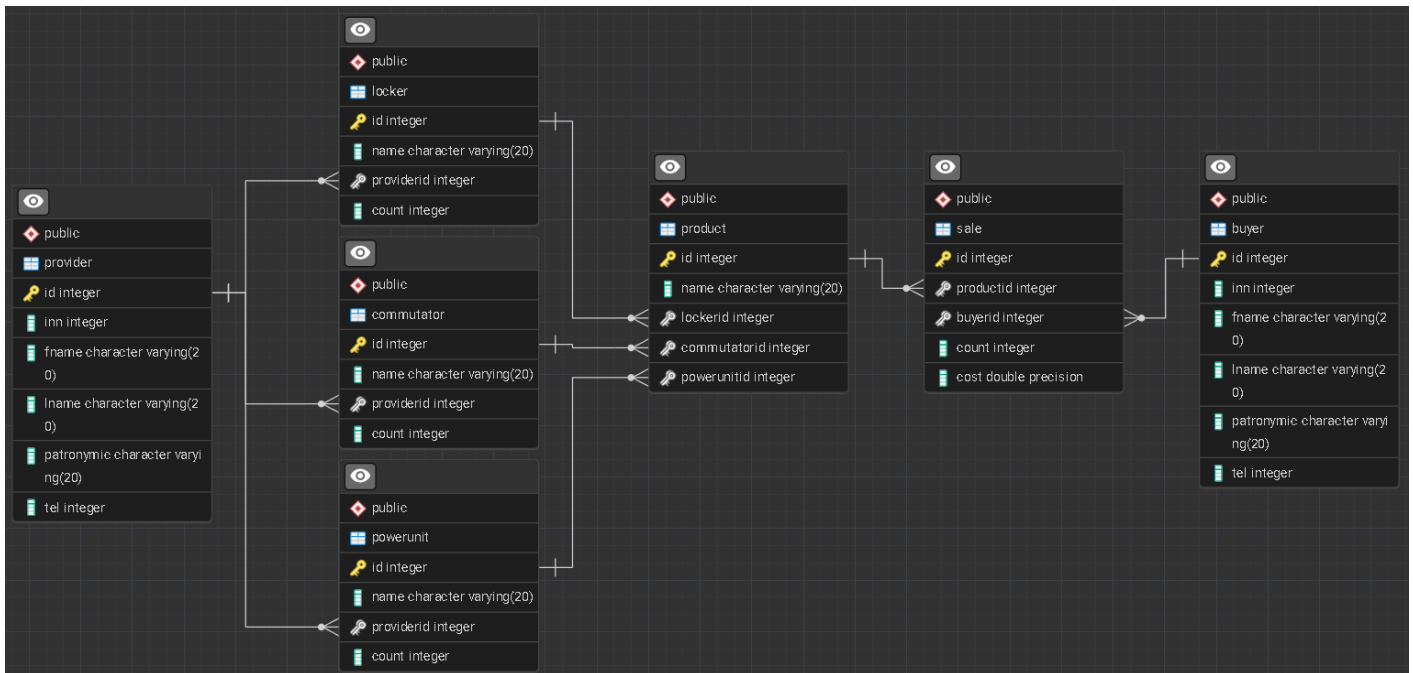
def update_sale(row):
    row+=1
    db = sqlite3.connect('#dataBase.db')
    cursor = db.cursor()
    cursor.execute('''
UPDATE
    sale
SET
    productID = ?,
    buyerID = ?,
    count = ?,
    cost = ?
WHERE
    id = ?
''', (
        ui.productIDSale.text(),
        ui.buyerIDSale.text(),
        ui.countSale.text(),
        ui.costSale.text(),
        row
    ))
    db.commit()
    db.close()
    ui.productIDSale.setText('')
    ui.buyerIDSale.setText('')
    ui.countSale.setText('')
    ui.costSale.setText('')
    createTable()
    ui.okSale.setVisible(False)

def dataCell_sale(row, column):
    index = modelTableSale.index(row, column)
    value = modelTableSale.data(index, QtCore.Qt.ItemDataRole.DisplayRole)
    return value

ui.addSale.clicked.connect(add_sale)
createTable()

```

ERD-диаграмма и код БД:



```

create table if not exists provider(
    id integer primary key,
    inn integer not null,
    fname varchar(20) not null,
    lname varchar(20) not null,
    patronymic varchar(20) not null,
    tel integer not null
);
    
```

```

create table if not exists locker(
    id integer primary key,
    name varchar(20) not null,
    providerID integer references provider(id) not null,
    count integer not null
);
    
```

```

create table if not exists commutator(
    id integer primary key,
    name varchar(20) not null,
    providerID integer references provider(id) not null,
    count integer not null
);
    
```

```

create table if not exists powerUnit(
    id integer primary key,
    
```

```

    name varchar(20) not null,
    providerID integer references provider(id) not null,
    count integer not null
);

create table if not exists buyer(
    id integer primary key,
    inn integer not null,
    fname varchar(20) not null,
    lname varchar(20) not null,
    patronymic varchar(20) not null,
    tel integer not null
);

create table if not exists product(
    id integer primary key,
    name varchar(20) not null,
    lockerID integer references locker(id) not null,
    commutatorID integer references commutator(id) not null,
    powerUnitID integer references powerUnit(id) not null
);

create table if not exists sale(
    id integer primary key,
    productID integer references product(id) not null,
    buyerID integer references buyer(id) not null,
    count integer not null,
    cost float not null
);

```

Скриншоты приложения и информация к ним:

Поставщики:

Form

Provider (Поставщики)
Locker (Шафы)
Commutator (Коммутаторы)
PowerUnit (БП/ИБП)
Product (Товары)
Sale (Продажи)
Buyer (Покупатели)

Таблица: Поставщики

Добавить

ИНН

Имя

Фамилия

Отчество

Телефон

	id	ИНН	Имя	Фамилия	Отчество	Телефон	
1	1	12345678901	Иван	Иванов	Иванович	89012345678	Изменить
2	2	23456789012	Петр	Петров	Петрович	89023456789	Изменить
3	3	34567890123	Анна	Сидорова	Александровна	89034567890	Изменить
4	4	45678901234	Мария	Кузнецова	Игоревна	89045678901	Изменить
5	5	56789012345	Алексей	Смирнов	Сергеевич	89056789012	Изменить
6	6	67890123456	Ольга	Морозова	Петровна	89067890123	Изменить
7	7	78901234567	Дмитрий	Волков	Андреевич	89078901234	Изменить
8	8	89012345678	Елена	Лебедева	Викторовна	89089012345	Изменить
9	9	90123456789	Сергей	Козлов	Дмитриевич	89090123456	Изменить
10	10	10234567890	Наталья	Новикова	Владимировна	89101234567	Изменить

Шафы:

Form

Provider (Поставщики)
Locker (Шафы)
Commutator (Коммутаторы)
PowerUnit (БП/ИБП)
Product (Товары)
Sale (Продажи)
Buyer (Покупатели)

Таблица: Шафы

Добавить

Номенклатура

Поставщик

Кол-во

	id	Шаф	id_Поставщика	Количество	
1	1	Шаф-10U	5	50	Изменить
2	2	Шаф-20U	1	45	Изменить
3	3	Шаф-42U	8	30	Изменить
4	4	Шаф-...	3	100	Изменить
5	5	Стойка-...	10	75	Изменить
6	6	Шаф-IP55	2	20	Изменить
7	7	Шаф-...	4	15	Изменить
8	8	Шаф-телеком...	9	60	Изменить
9	9	Шаф-...	6	10	Изменить
10	10	Шаф-уличны...	7	5	Изменить

Коммутаторы:

Form

Provider (Поставщики)

Locker (Шафы)

Commutator (Коммутаторы)

PowerUnit (БП/ИБП)

Product (Товары)

Sale (Продажи)

Buyer (Покупатели)

Таблица: Коммутаторы

Добавить

Номенклатура

Поставщик

Кол-во

	id	Коммутатор	id_Поставщика	Количество	
1	1	Cisco-2960-8TC	7	200	Изменить
2	2	Mikrotik-CRS328	3	150	Изменить
3	3	Huawei-S5700	9	80	Изменить
4	4	DLink-DGS-1210	1	300	Изменить
5	5	TP-Link-SG108	5	500	Изменить
6	6	Cisco-Nexus-9K	2	20	Изменить
7	7	Juniper-EX4300	8	45	Изменить
8	8	Arista-7050S	4	30	Изменить
9	9	Extreme-X440-G2	10	60	Изменить
10	10	ZyxeI-GS1920	6	120	Изменить

Блоки питания:

Form

Provider (Поставщики)

Locker (Шафы)

Commutator (Коммутаторы)

PowerUnit (БП/ИБП)

Product (Товары)

Sale (Продажи)

Buyer (Покупатели)

Таблица: Блоки питания

Добавить

Номенклатура

Поставщик

Кол-во

	id	БП	id_Поставщика	Количество	
1	1	UPS-APC-...	2	90	Изменить
2	2	UPS-Eaton-SPX	4	70	Изменить
3	3	PDU-8-розеток...	6	400	Изменить
4	4	БП-...	1	150	Изменить
5	5	БП-...	9	110	Изменить
6	6	UPS-...	3	60	Изменить
7	7	PDU-...	8	50	Изменить
8	8	БП-Cisco-PWR...	5	80	Изменить
9	9	Стабилизатор-...	10	200	Изменить
10	10	Дизель-...	7	5	Изменить

Товары:

Form

Provider (Поставщики)

Locker (Шафы)

Commutator (Коммутаторы)

PowerUnit (БП/ИБП)

Product (Товары)

Sale (Продажи)

Buyer (Покупатели)

Добавить

НоменклатураШафКоммутаторБлок питания

	id	Товар	id_Шафа	id_Коммутатора	id_БП	
1	1	Сборка-Офис-...	1	1	1	Изменить
2	2	Сборка-Офис-...	2	2	2	Изменить
3	3	Сборка-ЦОД-...	3	3	3	Изменить
4	4	Сборка-ЦОД-...	3	6	6	Изменить
5	5	Сборка-Малы...	4	4	4	Изменить
6	6	Сборка-Склад-...	5	5	5	Изменить
7	7	Сборка-Улица...	6	9	9	Изменить
8	8	Сборка-...	8	5	3	Изменить
9	9	Сборка-...	1	1	4	Изменить
10	10	Сборка-...	3	3	5	Изменить

Продажи:

Form

Provider (Поставщики)

Locker (Шафы)

Commutator (Коммутаторы)

PowerUnit (БП/ИБП)

Product (Товары)

Sale (Продажи)

Buyer (Покупатели)

Добавить

НоменклатураПокупательКол-воЦена

	id	id_Товара	id_Покупателя	Количество	Цена	
1	1	1	1	2	150000.0	Изменить
2	2	3	2	1	550000.0	Изменить
3	3	5	3	5	85000.5	Изменить
4	4	2	4	1	220000.0	Изменить
5	5	10	5	1	990000.0	Изменить
6	6	8	6	3	110000.25	Изменить
7	7	4	7	2	750000.0	Изменить
8	8	9	8	4	180000.0	Изменить
9	9	6	9	1	300000.0	Изменить
10	10	7	10	1	450000.0	Изменить

Покупатели:

Form

Provider (Поставщики)

Locker (Шафы)

Commutator (Коммутаторы)

PowerUnit (БП/ИБП)

Product (Товары)

Sale (Продажи)

Buyer (Покупатели)

Добавить

ИНН

Имя

Фамилия

Отчество

Телефон

	id	ИНН	Имя	Фамилия	Отчество	Телефон	
1	1	99001234567	Александр	Смирнов	Владимирович	89201112233	Изменить
2	2	99002345678	Екатерина	Петрова	Сергеевна	89202223344	Изменить
3	3	99003456789	Михаил	Соколов	Иванович	89203334455	Изменить
4	4	99004567890	Оксана	Федорова	Андреевна	89204445566	Изменить
5	5	99005678901	Дмитрий	Морозов	Павлович	89205556677	Изменить
6	6	99006789012	Светлана	Волкова	Алексеевна	89206667788	Изменить
7	7	99007890123	Игорь	Лебедев	Дмитриевич	89207778899	Изменить
8	8	99008901234	Татьяна	Козлова	Геннадьевна	89208889900	Изменить
9	9	99009012345	Павел	Семенов	Олегович	89209990011	Изменить
10	10	99010123456	Юлия	Голубева	Николаевна	89210001122	Изменить