

**Колледж космического машиностроения и технологий**

**ОТЧЕТ**

**по производственной практике ПП.01.01**

**по профессиональному модулю ПМ.01**

**«Разработка программных модулей программного обеспечения для компьютерных систем»**

Специальность 09.02.03 «Программирование в компьютерных

системах»

**Выполнил:**

**Студент 3 курса**

**Группы П1-17**

**Бобнев Алексей Олегович**

**Королев**

**2019**

Содержание

[Введение 3](#_Toc27860647)

[1 Характеристика объекта практики 4](#_Toc27860648)

[1.1 Структура организации, характеристика основных видов деятельности 4](#_Toc27860649)

[1.2 Должностные обязанности оператора ЭВМ, техника – программиста, инженера программиста 5](#_Toc27860650)

[1.2.1 Должностные обязанности оператора ЭВМ 5](#_Toc27860651)

[1.2.2 Должностные обязанности техника-программиста 6](#_Toc27860652)

[1.2.3 Должностные обязанности инженера-программиста 7](#_Toc27860653)

[1.3 Документооборот предприятия, структурного подразделения 7](#_Toc27860654)

[1.4 Состав технических средств обработки данных 9](#_Toc27860655)

[2 Разработка отдельных программных моделей 10](#_Toc27860656)

[2.1 Разработка технического задания 10](#_Toc27860657)

[2.2 Спецификация программного продукта 11](#_Toc27860658)

[2.3 Разработка клиентского приложения 12](#_Toc27860659)

[2.4 Разработка руководства программиста 17](#_Toc27860660)

[2.4.1 Назначение и условия применения программы 17](#_Toc27860661)

[2.4.2 Обращение к программе 17](#_Toc27860662)

[2.4.3 Входные и выходные данные 18](#_Toc27860663)

[2.4.4 Структура системы 18](#_Toc27860664)

[2.4.5 Описание основных функций 18](#_Toc27860665)

[2.4.6 Сообщения 18](#_Toc27860666)

[Дневник практики 19](#_Toc27860667)

[ЗАКЛЮЧЕНИЕ 20](#_Toc27860668)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 21](#_Toc27860669)

[Приложение 22](#_Toc27860670)

[Приложение А. 22](#_Toc27860671)

# Введение

С 27.10.2019 по 28.12.2019 я проходил производственную практику в Государственном бюджетном образовательном учреждении высшего образования Московской области «Технологический университет» в отделе формирования и реализации дистанционных технологий. Во время прохождения практики я поставил для себя следующие цели:

* приобрести опыт работы по специальности;
* закрепить теоретические знания, полученные во время учебы;
* выполнение требований и действий, предусмотренных программой

производственной практики и заданий руководителя;

* проанализировать работу отдела;
* оптимизировать работу организации;
* закрепить навыки в разработке проектной и технической документации;
* закрепить навыки отладки и тестирования программных модулей.

Для выполнения поставленных мной целей я выдвинул следующие задачи:

* изучить специфику деятельности организации;
* установить необходимые инструменты для работы с базами данных;
* найти подходящую литературу.

После прохождения этой практики я хочу повысить свои навыки в разработке программных модулей, а также получить новые.

# 1 Характеристика объекта практики

## Структура организации, характеристика основных видов деятельности

Данное предприятие работает в сфере образования. Университет образован 16 июля 1998 года в форме некоммерческой организации с названием: Негосударственное образовательное учреждение «Королевская академия управления, экономики и социологии».

Технологический университет (ранее Финансово-технологическая академия; Королевский институт управления, экономики и социологии) создан для подготовки кадров новой информации, воспроизводства интеллектуальных ресурсов, формирования инновационных проектов и технологий. Академия находится в наукограде Королеве Московской области – уникальном центре интеллектуальных ресурсов, которые используются для интеграции важнейших знаний и создания систем глобального масштаба.

20 января 2015 года постановлением Правительства Московской области Академии присвоен статус «университета» и вуз переименован в Государственное бюджетное образовательное учреждение высшего образования Московской области «Технологический университет». Структура университета представлена на рисунке 1.

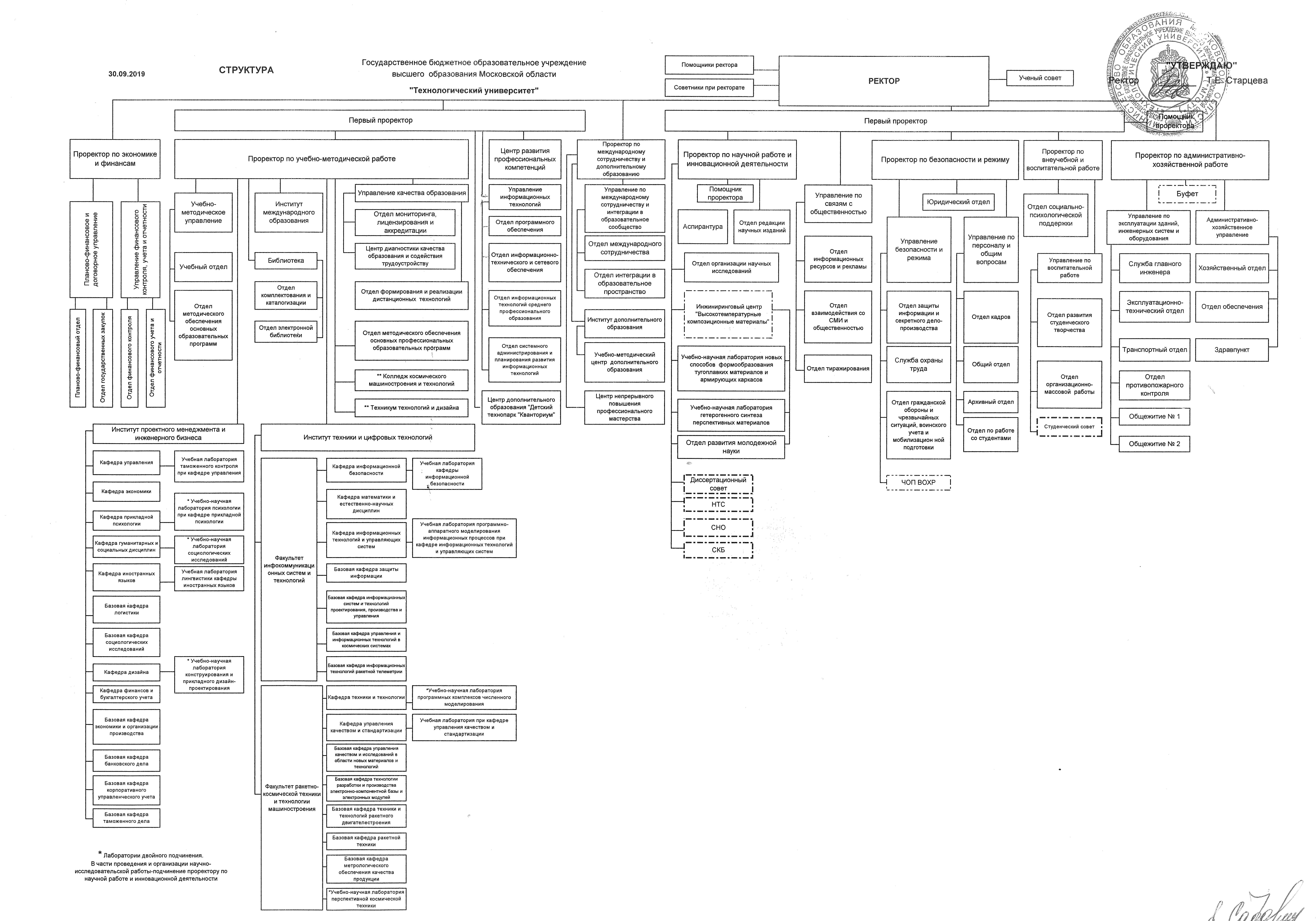


Рисунок 1 – структура «Технологического университета»

## Должностные обязанности оператора ЭВМ, техника – программиста, инженера программиста

### 1.2.1 Должностные обязанности оператора ЭВМ

* осуществляет техническую подготовку документации, необходимой в процессе работы компании. Выполняет копирование документов на ксероксе;
* выполняет набор различных текстов с соблюдением правил орфографии и пунктуации, а также стандартов оформления организационно-распорядительной документации;
* осуществляет работу с электронной почтой, принимает входящие электронные письма и следит за своевременной отправкой исходящих;
* распечатывает и систематизирует нужные документы;
* заносит в компьютерные базы данных различную информацию, важную и необходимую для работы компании;
* следит за состоянием компьютера и копировальной техники;
* своевременно информирует руководство о необходимости приобретения материалов, непосредственно относящихся к производственному процессу.

### 1.2.2 Должностные обязанности техника-программиста

* выполняет работу по обеспечению механизированной и автоматизированной обработки, поступающей в ВЦ (ИВЦ) информации, разработки технологии решения экономических и других задач производственного и научно-исследовательского характера;
* принимает участие в проектировании систем обработки данных и систем математического обеспечения машины;
* выполняет подготовительные операции, связанные с осуществлением вычислительного процесса, ведет наблюдение за работой машин;
* составляет простые схемы технологического процесса обработки информации, алгоритмы решения задач, схемы коммутации, макеты, рабочие инструкции и необходимые пояснения к ним;
* разрабатывает программы решения простых задач, проводит их отладку и экспериментальную проверку отдельных этапов работ;
* выполняет работу по подготовке технических носителей информации, обеспечивающих автоматический ввод данных в вычислительную машину, по накоплению и систематизации показателей нормативного и справочного фонда, разработке форм исходящих документов, внесению необходимых изменений и своевременному корректированию рабочих программ;
* участвует в выполнении различных операций технологического процесса обработки информации (прием и контроль входной информации, подготовка исходных данных, обработка информации, выпуск исходящей документации и передача ее заказчику);
* ведет учет использования машинного времени, объемов выполненных работ;
* выполняет отдельные служебные поручения своего непосредственного руководителя.

### 1.2.3 Должностные обязанности инженера-программиста

* на основе анализа математических моделей и алгоритмов решения экономических и других задач разрабатывает программы, обеспечивающие возможность выполнения алгоритма и соответственно поставленной задачи средствами вычислительной техники, проводит их тестирование и отладку;
* разрабатывает технологию решения задач по всем этапам обработки информации;
* осуществляет выбор языка программирования для описания алгоритмов и структур данных;
* определяет информацию, подлежащую обработке средствами вычислительной техники, ее объемы, структуру, макеты и схемы ввода, обработки, хранения и вывода, методы ее контроля;
* выполняет работу по подготовке программ к отладке и приводит отладку;
* определяет объем и содержание данных контрольных примеров, обеспечивающих наиболее полную проверку соответствия программ их функциональному назначению;
* осуществляет запуск отлаженных программ и ввод исходных данных, определяемых условиями поставленных задач;
* проводит корректировку разработанной программы на основе анализа выходных данных;
* разрабатывает инструкции по работе с программами, оформляет необходимую техническую документацию;
* определяет возможность использования готовых программных продуктов;
* осуществляет сопровождение внедрения программ и программных средств;
* разрабатывает и внедряет системы автоматической проверки правильности программ, типовые и стандартные программные средства, составляет технологию обработки информации;
* выполняет работу по унификации и типизации вычислительных процессов;
* принимает участие в создании каталогов и картотек стандартных программ, в разработке форм документов, подлежащих машинной обработке, в проектировании программ, позволяющих расширить область применения вычислительной техники.

## Документооборот предприятия, структурного подразделения

Документооборот Отдела в сфере поставленной мне на практике задачи состоит из нескольких этапов:

* получение приказа и распределение работы между сотрудниками;
* перечень существующих дел в Отделе;
* годовой план работ;
* годовой отчет по проделанной работе.

Вид построенной IDEF модели по плану документооборота представлен на рисунках 2 – 4:

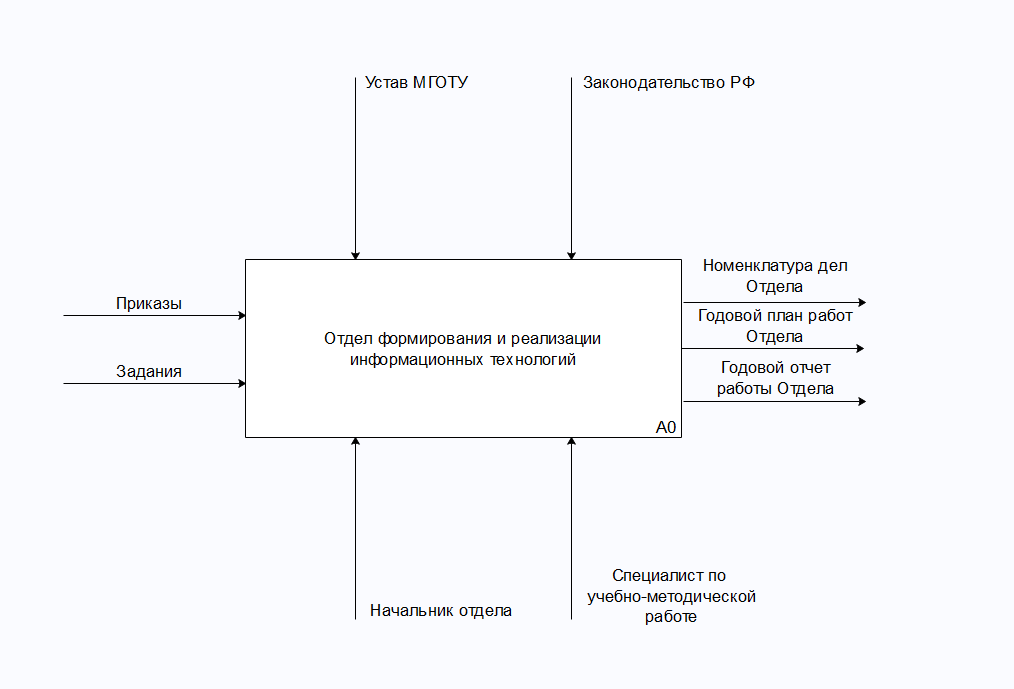


Рисунок 2 – IDEF – модель 1 уровень

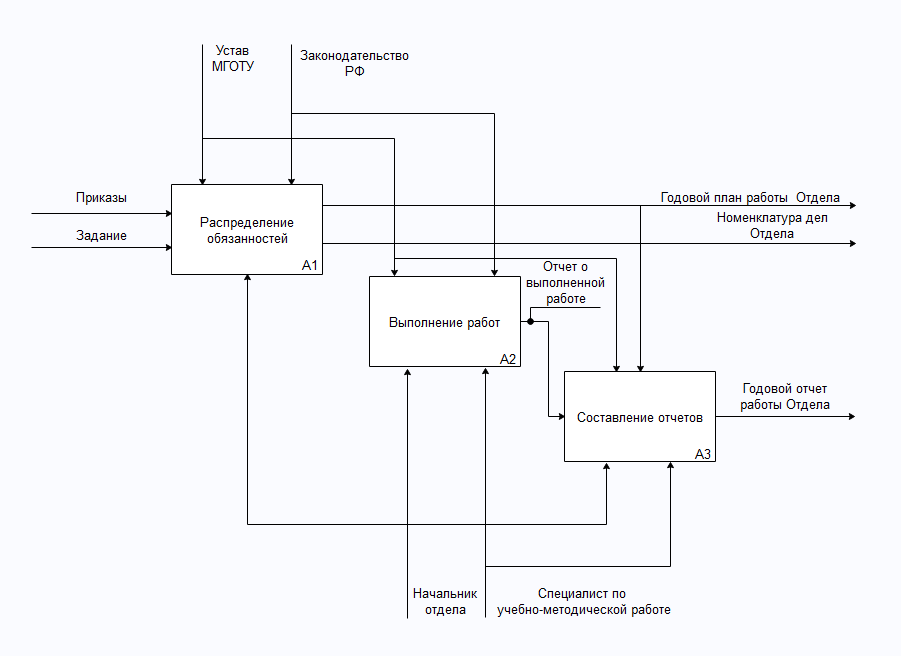


Рисунок 3 – IDEF – модель 2 уровень

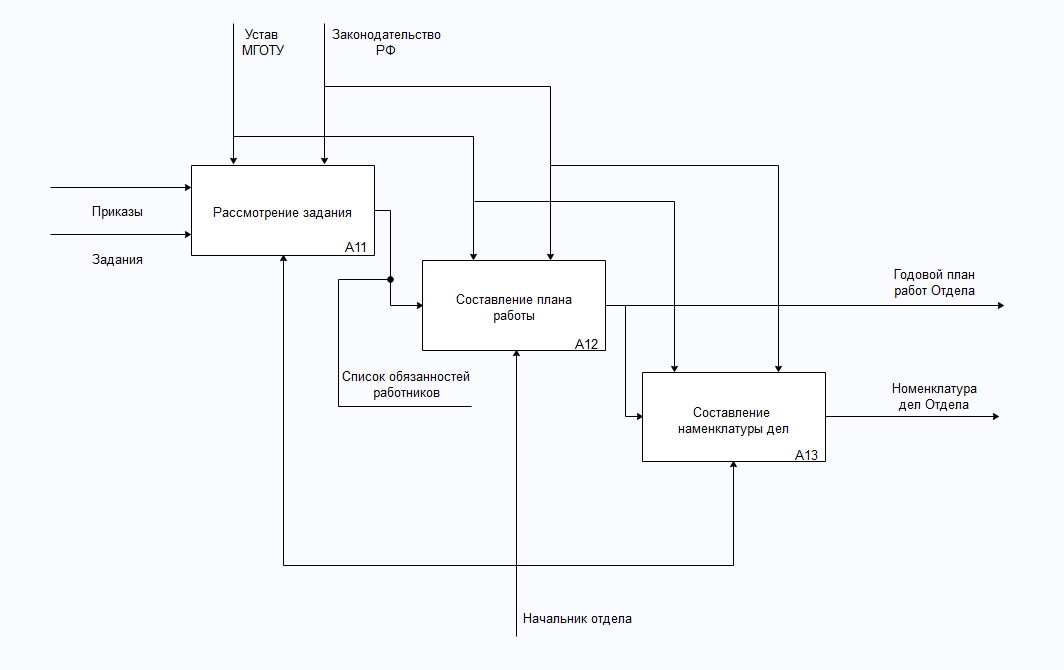


Рисунок 4 – IDEF – модель подуровня блока «Распределение задания»

## Состав технических средств обработки данных

При проектировании технологических процессов ориентируются на режимы их реализации. Режим реализации технологии зависит от объемно-временных особенностей решаемых задач: периодичности и срочности, требований к быстроте обработки сообщений, а также от режимных возможностей технических средств, и в первую очередь ЭВМ. Существуют следующие режимы:

* пакетный режим;
* режим реального масштаба времени;
* режим разделения времени;
* регламентный режим;
* запросной режим;
* диалоговый режим;
* режим телеобработки;
* интерактивный режим;
* однопрограммный и многопрограммный режимы.

В «Технологическом университете» прослеживаются три режима обработки данных: пакетный режим, диалоговый и регламентный. При использовании пакетного режима пользователь не имеет непосредственного общения с компьютером. Сбор и регистрация информации, ввод и обработка не совпадают по времени. Вначале работники собирают информацию, формируют ее в пакеты в соответствии с видом задач, затем вводят ее в компьютер и там она обрабатывается.

В состав технических средств обработки данных входит:

* ноутбук:

1. процессор Intel(R) Core(TM) i7 – 2670QM CPU @ 2.20 Ghz 2.20 Ghz
2. материнская плата Dell PWA PLN 2xCPU Motherboard (PV755N)
3. видеокарта GeForce GT 710;
4. оперативная память 8 ГБ

* компьютерная мышь Genius NX-7000;

# 2 Разработка отдельных программных моделей

## 2.1 Разработка технического задания

1. Краткая характеристика области применения программы

Данный проект создан для работы в автосервисе.

1. Назначение разработки
   1. Функциональное назначение программы

Функциональным назначением программы является обработка информации, связанная с автомобилями.

* 1. Эксплуатационное назначение программы

Программа может эксплуатироваться в любых условиях. Не требует специального оборудования и дополнительных установок. Пользователями программы могут быть обычные люди, не имеющие специального образования.

1. Требования к программному изделию
   1. Требования к функциональным характеристикам
      1. Требования к составу выполняемых функций

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

* функция введения данных с клавиатуры;
* функция диагностика двигателя;
* функция сравнения автомобилей;
* учет сотрудников;
* учет товара на складе;
* учет клиентской базы;
* учет финансов;
* функция вывода на экран информацию о разработчике;
* вывод информации о стоимости услуг в автосервисе;
* функция внесения данных в базу.
  + 1. Организация входных и выходных данных

Входные данные программы представлены в виде вводимых с клавиатуры данных.

Выходные файлы – готовые документы в формате «txt».

3.2 Требования к составу и параметрам технических средств

* видеокарта 4 Gb;
* процессор с тактовой частотой 800 MHz или более мощный;
* оперативная память 1 ГБ и выше;
* архитектура с разрядностью 32 бит или 64 бит;
* операционная система Windows 7, Windows 8 или Windows 10;
* свободное место на жестком диске от 64 Мб.

4. Требования к программным средствам, используемым программой

Системные программные средства, используемые программой, должны быть представлены локализованной версией операционной системы Windows 7 и выше.

## 2.2 Спецификация программного продукта

Таблица 1 – Спецификация

|  |  |  |
| --- | --- | --- |
| Обозначения | Наименование | Примечание |
|  | Документация |  |
| НОМЕР ДОКУМЕНТА 19.504-79 | Руководство программиста |  |
|  | Комплексы |  |
|  | Автосканер |  |
|  | Компоненты |  |
|  | Датчики электронного блока управления |  |

## 2.3 Разработка клиентского приложения

Согласно разработанному техническому заданию создаем приложение для автосервиса.

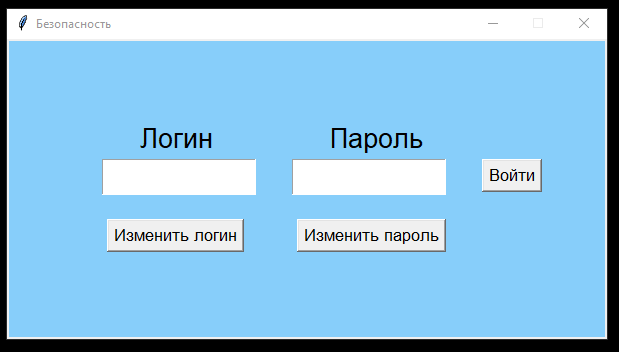


Рисунок 5 – Окно «Безопасность»

На рисунке 5 изображено окно, которое требует логин и пароль для входа в программу.

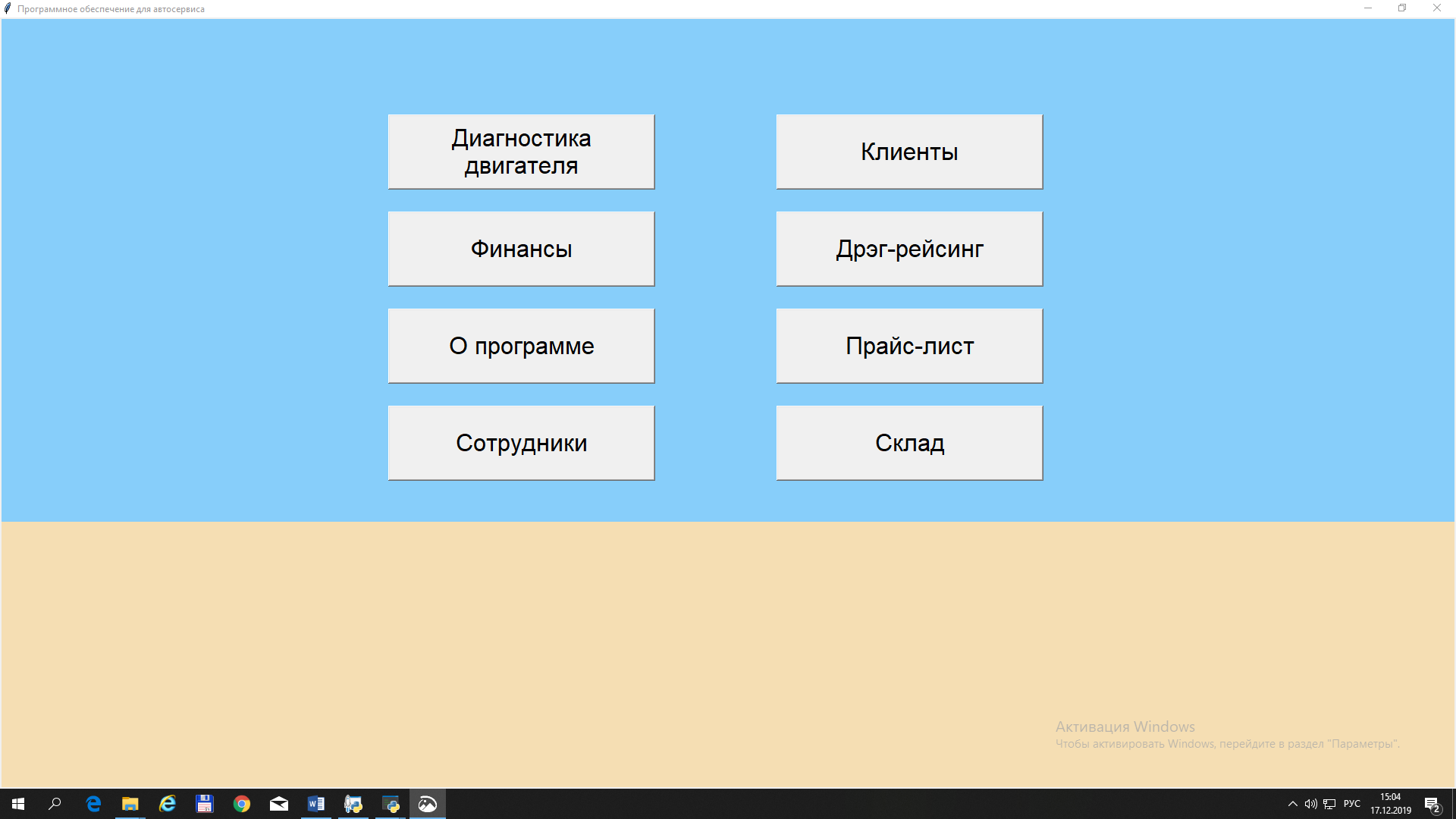


Рисунок 6 – Окно «Программное обеспечение для автосервиса»

На рисунке 6 изображено окно с главным интерфейсом.

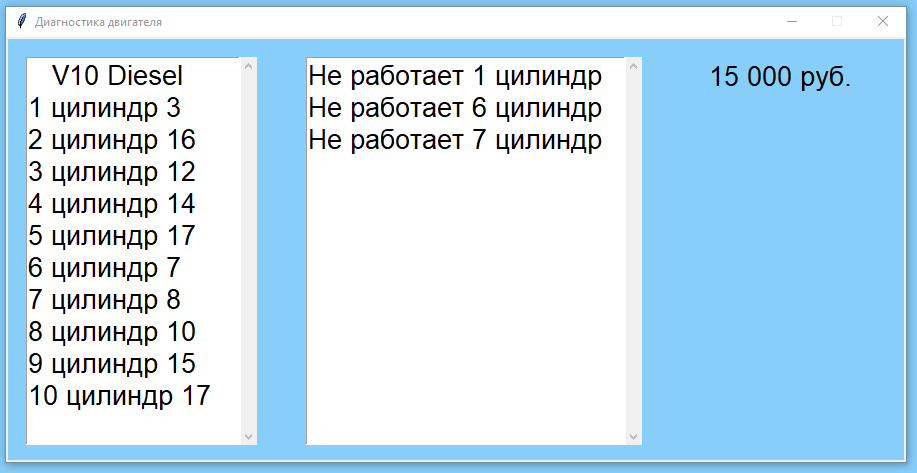


Рисунок 7 – Окно «Диагностика двигателя»

На рисунке 7 изображено окно, с помощью которого происходит диагностика двигателя. В результате мы получаем список цилиндров, которые находятся в двигателе. В середине окна расположен список неисправных цилиндров. Справа выводится стоимость ремонта.

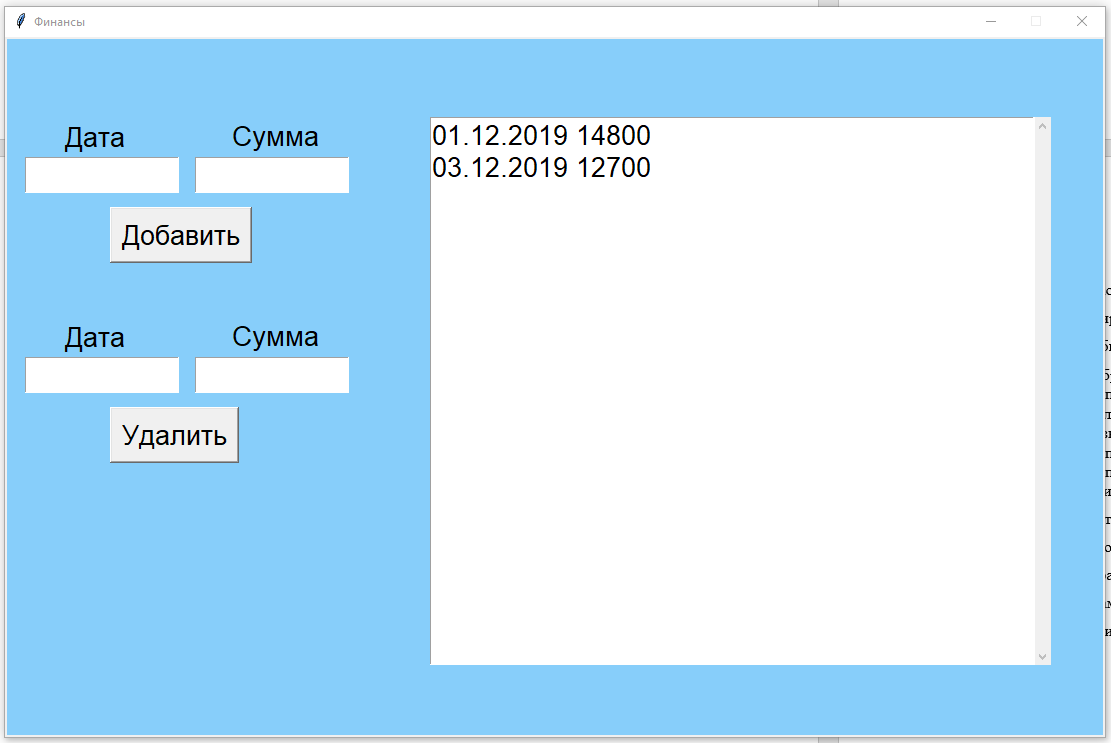


Рисунок 8 – Окно «Финансы»

На рисунке 8 изображено окно, с помощью которого можно отслеживать финансовую сторону предприятия.

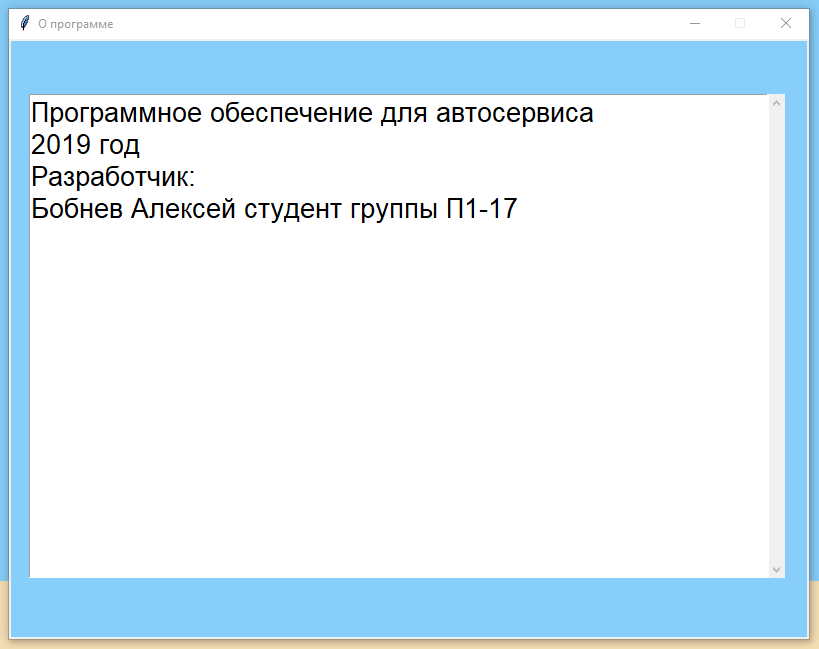


Рисунок 9 – Окно «О программе»

На рисунке 9 изображено окно, которое содержит информацию о разработчике.

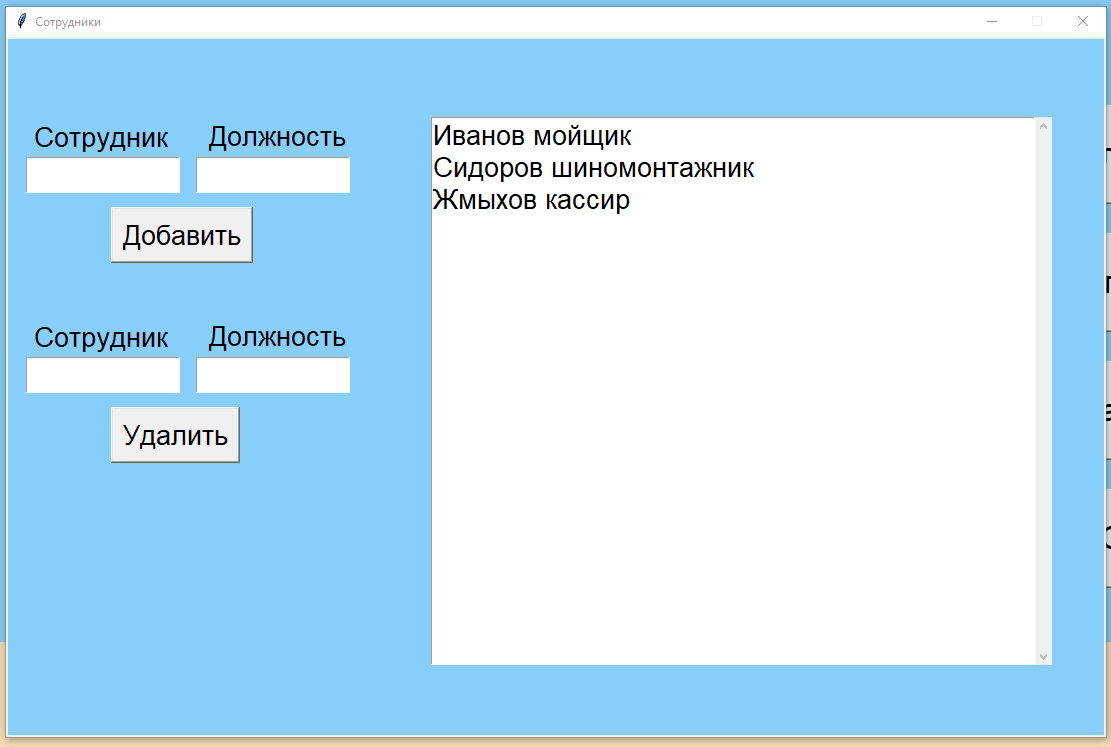


Рисунок 10 – Окно «Сотрудники»

На рисунке 10 изображено окно, с помощью которого можно производить учет сотрудников.

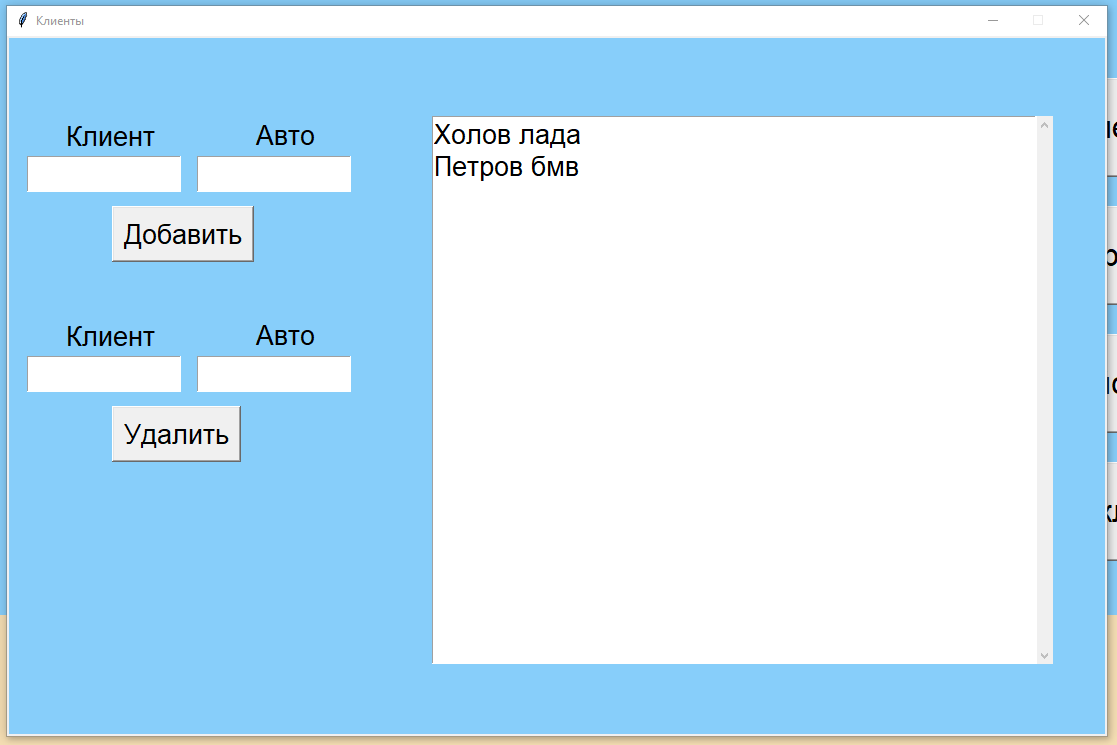


Рисунок 11 – Окно «Клиенты»

На рисунке 11 изображено окно, с помощью которого можно производить учет клиентов.

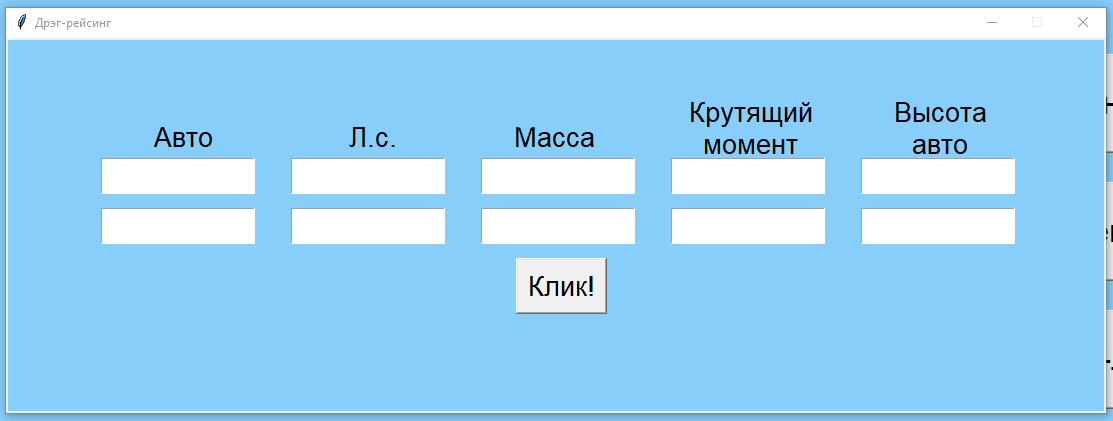


Рисунок 12 – Окно «Дрэг-рейсинг»

На рисунке 12 изображено окно, с помощью которого можно сравнить технические характеристики автомобилей.

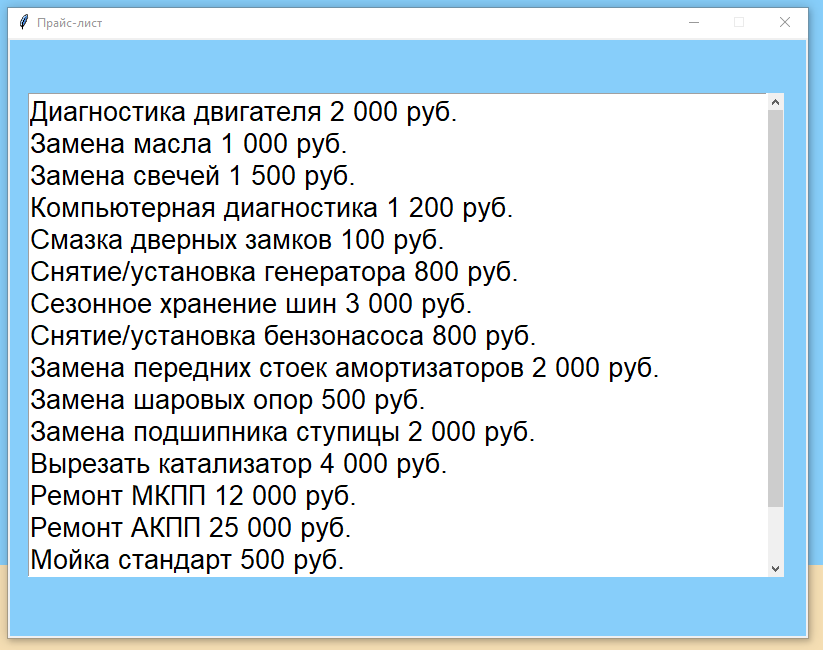


Рисунок 13 – Окно «Прайс-лист»

На рисунке 13 изображено окно, с помощью которого можно узнать стоимость услуг в автосервисе.



Рисунок 14 – Окно «Склад»

На рисунке 14 изображено окно, с помощью которого можно производить учет товаров.

## 2.4 Разработка руководства программиста

### 2.4.1 Назначение и условия применения программы

Программный продукт может быть использован в организациях, которые связаны с автомобильным делом.

Для работы программного продукта необходима следующая программно-аппаратная конфигурация:

* Windows 7, Windows Server 2003 Service Pack 2, Windows Server 2008, Windows Vista, Windows Vista Service Pack 1, Windows XP Service Pack 2, Windows XP Service Pack 3;
* 64-разрядные системы: процессор с тактовой частотой 1,4 ГГц или выше;
* минимальный объем ОЗУ 512 МБ;
* 1 ГБ свободного места на диске;
* автосканеры;
* датчики электронного блока управления;

### 2.4.2 Обращение к программе

Для запуска программы необходимо выполнить следующие действия:

* запустить программу на ПК двойным щелчком мыши;
* авторизоваться или зарегистрироваться в системе;
* откроется главное окно программы;
* начать работу.

Не авторизованный (не зарегистрированный) пользователь не может просматривать какую-либо вкладку.

### 2.4.3 Входные и выходные данные

В качестве входной информации представлены текстовые и числовые данные.

В качестве выходной информации представлены текстовые и числовые данные.

### 2.4.4 Структура системы

Разработанный программный продукт представлен следующими файлами:

* «py» - формат для разработки программы;
* «txt» - формат для записи и чтения данных.

### 2.4.5 Описание основных функций

При разработке программного обеспечения были выделены следующие основные функции:

* сравнение технических характеристик автомобилей;
* учет данных;
* диагностика автомобиля.

### 2.4.6 Сообщения

Таблица 2 – Сообщения

|  |  |  |
| --- | --- | --- |
| **Сообщения** | **Причина** | **Решение** |
| Неверно | Ввод неверных данных | Ввести верные данные |

# Дневник практики

Дневник прохождения производственной практики

Записи о выполненных работах

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Краткое содержание работ | Отметка о выполнении работы | Подпись руководителя практики |
| 27.10.2019 – 02.11.2019 | Составление технического задания, определение языка программирования, согласование задания с руководством | + |  |
| 03.11.2019 – 09.11.2019 | Изучение предметной области, написание спецификации | + |  |
| 10.11.2019 – 16.11.2019 | Изучение и закрепление теории для создания приложения | + |  |
| 17.11.2019 – 23.11.2019 | Создание клиентского приложения, подбор необходимых библиотек | + |  |
| 24.11.2019 – 30.11.2019 | Создание графического интерфейса пользователя (GUI), согласование GUI с руководством | + |  |
| 01.12.2019 – 07.12.2019 | Отладка и тестирование GUI, предварительный показ целевой аудитории | + |  |
| 08.12.2019 – 14.12.2019 | Отладка программы, рефакторинг, оптимизация | + |  |
| 15.12.2019 – 21.12.2019 | Тестирование программы, выявление и исправление ошибок | + |  |
| 22.12.2019 – 28.12.2019 | Разработка руководства пользователя, сдача отчета | + |  |

# ЗАКЛЮЧЕНИЕ

Перед прохождением производственной практики в Государственном бюджетном образовательном учреждении высшего образования Московской области «Технологический университет» мной были поставлены следующие основные цели:

* приобрести опыт работы по специальности;
* закрепить теоретические знания, полученные во время учебы;
* выполнение требований и действий, предусмотренных программой производственной практики и заданий руководителя;
* закрепить навыки в разработке проектной и технической документации;
* закрепить навыки отладки и тестирования программных модулей.

По окончании практики я достиг своих целей. Для этого я использовал учебную литературу, интернет – источники, документацию средств разработки.

Во время прохождения практики я приобрел опыт работы по специальности. Также был закреплен навык разработки проектной и технической документации и навык отладки и тестирования программных модулей.

По окончании практики был составлен отчет.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. ГОСТ 19.202-78 Единая система программной документации (ЕСПД). Спецификация. Требования к содержанию и оформлению
2. ГОСТ 19.201-78 Единая система программной документации (ЕСПД). Техническое задание. Требования к содержанию и оформлению
3. ГОСТ 19.504-79 ЕСПД. Руководство программиста. Требования к содержанию и оформлению
4. ГОСТ 7.32-2017 СИБИД. Отчет о научно-исследовательской работе. Структура и правила оформления
5. https://pythonru.com/uroki/obuchenie-python-gui-uroki-po-tkinter

# Приложение

## Приложение А.

Модуль «parol»

import ctypes

from tkinter import \*

import Cena # Импортируем модули

import Fina

import Gonka

import Klient

import Raz

import dia

import sklad

import sotr

import V023

window = Tk()

window.resizable(False, False)

window.title("Безопасность")

window.geometry('600x300')# Задаем размер окна

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

# user32.GetSystemMetrics(0) - ширина

# user32.GetSystemMetrics(1) - длина

canvas = Canvas(window, width = user32.GetSystemMetrics(0), height = user32.GetSystemMetrics(1))

canvas.configure(bg = 'lightskyblue')# Цвет окна

canvas.pack()

fontSize = user32.GetSystemMetrics(0) // 80

fontParameters = "Batang " + str(fontSize)

txt1 = Entry(width = 10, font = 'Batang 20')

txt1.place(x = 95, y = 120)

txt2 = Entry(width = 10, font = 'Batang 20')

txt2.config(show="\*")

txt2.place(x = 285, y = 120)

def read():# Функция для чтения из файла

t = open('login.txt', 'r')

z = t.read()

t1 = open('password.txt', 'r')

z1 = t1.read()

if z == txt1.get() and txt2.get() == z1:

V023.osnova()

else:

df = Tk()

df.resizable(False, False)

df.title("Ошибка")

df.geometry('250x150')# Задаем размер окна

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

canvas = Canvas(df, width = user32.GetSystemMetrics(0), height = user32.GetSystemMetrics(1))

canvas.configure(bg = 'lightskyblue')

canvas.pack()

l3 = Label(df)

l3.configure(text = 'Неверно', font = 'Batang 20', bg = 'lightskyblue')

l3.place(x = 60, y = 60)

def login():# Изменяем логин

t = open('login.txt', 'w')

t.write(txt1.get())

def password():# Изменяем пароль

t = open('password.txt', 'w')

t.write(txt2.get())

l1 = Label()# Метки

l2 = Label()

l1.configure(text = 'Логин', font = 'Batang 20', bg = 'lightskyblue')

l1.place(x = 130, y = 80)

l2.configure(text = 'Пароль', font = 'Batang 20', bg = 'lightskyblue')

l2.place(x = 320, y = 80)

btn = Button(text = "Войти", command = read, font = 'Batang 13')

btn.place(x = 475, y = 120)

btn1 = Button(text = "Изменить логин", command = login, font = 'Batang 13')

btn1.place(x = 100, y = 180)

btn1 = Button(text = "Изменить пароль", command = password, font = 'Batang 13')

btn1.place(x = 290, y = 180)

window.mainloop()

Модуль «Главное окно»

import ctypes

from tkinter import \*

import Cena # Импортируем модули

import Fina

import Gonka

import Klient

import Raz

import dia

import sklad

import sotr

def osnova(): # Главное меню

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

window = Tk()

window.state("zoomed") # Полноэкранный режим окна

window.title("Программное обеспечение для автосервиса")

canvas = Canvas(window, width=user32.GetSystemMetrics(0), height=user32.GetSystemMetrics(1))

canvas.create\_line(-710, 875, user32.GetSystemMetrics(0), 875, width=420, fill="wheat") # Нижняя полоса

canvas.configure(bg='lightskyblue') # Фон окна

canvas.pack()

fontSize = user32.GetSystemMetrics(0) // 80

fontParameters = "Batang " + str(

fontSize) # Устанавливаем шрифт, в качестве его размера указываем предыдущую переменную

btnDiagnosis = Button(window, text="Диагностика\nдвигателя", width=widthButton, height=2, font=fontParameters,

command=dia.diagnost).place(x=(widthButtonPlace \* 4), y=widthButtonPlace) # Кнопки

btnFinance = Button(window, text="Финансы", width=widthButton, height=2, font=fontParameters,

command=Fina.finance).place(x=(widthButtonPlace \* 4), y=widthButtonPlace \* 2)

btnClient = Button(window, text="Клиенты", width=widthButton, height=2, font=fontParameters,

command=Klient.client).place(x=(widthButtonPlace \* 8), y=widthButtonPlace)

btnDrag = Button(window, text="Дрэг-рейсинг", width=widthButton, height=2, font=fontParameters,

command=Gonka.drag).place(x=(widthButtonPlace \* 8), y=widthButtonPlace \* 2)

btnAbout = Button(window, text="О программе", width=widthButton, height=2, font=fontParameters,

command=Raz.razrab).place(x=(widthButtonPlace \* 4), y=widthButtonPlace \* 3)

btnPrice = Button(window, text="Прайс-лист", width=widthButton, height=2, font=fontParameters,

command=Cena.price).place(x=(widthButtonPlace \* 8), y=widthButtonPlace \* 3)

btnStaff = Button(window, text="Сотрудники", width=widthButton, height=2, font=fontParameters,

command=sotr.sotrudniki).place(x=(widthButtonPlace \* 4), y=widthButtonPlace \* 4)

btnStorage = Button(window, text="Склад", width=widthButton, height=2, font=fontParameters,

command=sklad.sklad).place(x=(widthButtonPlace \* 8), y=widthButtonPlace \* 4)

window.mainloop()

Модуль «dia»

''' В этом модуле происходит диагностика двигателя '''

from tkinter import \*

import tkinter as tk

import random

import ctypes

def diagnost():

window = Tk()

window.resizable(False, False)

window.title("Диагностика двигателя")# Название окна

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

canvas = Canvas(window, width = user32.GetSystemMetrics(0), height = user32.GetSystemMetrics(1))

canvas.configure(bg = 'lightskyblue')

canvas.pack()

window.geometry('900x425')

stoim = 0 # Для расчета стоимости ремонта

cilindrov = int(random.uniform(1,13))# Рандомное количество цилиндров

if cilindrov == 1:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V4\n')

for i in range (1,5):

a = str(int(random.uniform(1,13)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'# Определение не работающих цилиндров

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

txt.insert(INSERT, '\n')

if cilindrov == 2:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')# Вывод текста

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V4 Diesel\n')

for i in range (1,5):

a = str(int(random.uniform(1,21)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 3:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V6\n')

for i in range (1,7):

a = str(int(random.uniform(1,13)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 4:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V6 Diesel\n')

for i in range (1,7):

a = str(int(random.uniform(1,21)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 5:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V8\n')

for i in range (1,9):

a = str(int(random.uniform(1,13)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 6:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V8 Diesel\n')

for i in range (1,9):

a = str(int(random.uniform(1,21)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 7:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V10\n')

for i in range (1,11):

a = str(int(random.uniform(1,13)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 8:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V10 Diesel\n')

for i in range (1,11):

a = str(int(random.uniform(1,21)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 9:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V12\n')

for i in range (1,13):

a = str(int(random.uniform(1,13)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 10:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V12 Diesel\n')

for i in range (1,13):

a = str(int(random.uniform(1,21)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 11:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V16\n')

for i in range (1,17):

a = str(int(random.uniform(1,13)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if cilindrov == 12:

txt = scrolledtext.ScrolledText(window, width = '14', height = '12', font = 'Batang 20')

dxd = scrolledtext.ScrolledText(window, width = '21', height = '12', font = 'Batang 20')

txt.insert(INSERT, ' V16 Diesel\n')

for i in range (1,17):

a = str(int(random.uniform(1,21)))

i1 = str(i) + ' ' + 'цилиндр' + ' ' + str(a) + '\n'

txt.insert(INSERT, i1)

txt.place(x = 20, y = 20)

if int(a) < 9:

ia = 'Не работает ' + str(i) + ' ' + 'цилиндр' + '\n'

dxd.insert(INSERT, ia)

dxd.place(x = 300, y = 20)

stoim += 1

if stoim > 0:

g1 = Label(window)

g1.configure(text = str(stoim \* 5) + ' 000 руб.', font = 'Batang 20', bg = 'lightskyblue')

g1.place(x = 700, y = 20)# Подсчет стоимости ремонта и вывод на экран

else:

g1 = Label(window)

g1.configure(text = 'Все работает', font = 'Batang 20', bg = 'lightskyblue')# Если все работает то выводим на экран "Все работает"

g1.place(x = 500, y = 170)

window.mainloop()

Модуль «Fina»

''' В этом модуле добавляется в файл дата и выручка автосервиса '''

from tkinter import \*

from tkinter import scrolledtext

import ctypes

def finance():

window = Tk()# Окно "window"

window.resizable(False, False)

window.title("Финансы")

window.geometry('1100x700')

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

canvas = Canvas(window, width = user32.GetSystemMetrics(0), height = user32.GetSystemMetrics(1))

canvas.configure(bg = 'lightskyblue')

canvas.pack()

f = open('bug.txt', 'r', encoding = 'utf-8')# Чтение русских букв

text = Text()

text = scrolledtext.ScrolledText(window, height = '17', font = 'Batang 20', width = '40')# 'font' - шрифт

text.insert(INSERT, f.read())

text.place(x = 425, y = 80)

f.close()

def clicked():# Добавление

z = open('bug.txt', 'r', encoding = 'utf-8')

y = z.read()

x = y.split()

t = []

for i in range (len(x)):

if i%2 == 0:

u = x[i] + ' ' + x[i+1]

t.append(u)

b = txt1.get()

b1 = txt2.get()

f = open('bug.txt', 'a', encoding = 'utf-8')

u11 = txt1.get() + ' ' + txt2.get() + '\n'

f.write(u11)

f.close()

f = open('bug.txt', 'r', encoding = 'utf-8')

text = Text()

text = scrolledtext.ScrolledText(window, height = '17', font = 'Batang 20', width = '40')

text.insert(INSERT, f.read())

text.place(x = 425, y = 80)

f.close()

def clicked1():# Удаление

z = open('bug.txt', 'r', encoding = 'utf-8')

y = z.read()

x = y.split()

t = []

for i in range (len(x)):

if i%2 == 0:

u = x[i] + ' ' + x[i+1]

t.append(u)

b = txt3.get()

b1 = txt4.get()

v = open('bug.txt','w', encoding = 'utf-8')

for i in range(len(x)):

j = t[i].split()

if i%2 == 0 and j[0] != b:

v.write(j[0])

v.write(' ')

v.write(j[1])

v.write('\n')

v.close()

f = open('bug.txt', 'r', encoding = 'utf-8')

text = Text()

text = scrolledtext.ScrolledText(window, height = '17', font = 'Batang 20', width = '40')

text.insert(INSERT, f.read())

text.place(x = 425, y = 80)

f.close()

lbl1 = Label(window)# Вывод в окно "window"

lbl2 = Label(window)

lbl1.configure(font = 'Batang 20', text = ' Дата', bg = 'lightskyblue')

lbl1.place(x = 25, y = 81)

lbl2.configure(font = 'Batang 20', text = ' Сумма', bg = 'lightskyblue')

lbl2.place(x = 200, y = 80)

txt1 = Entry(window, width = 10, font = 'Batang 20')

txt1.place(x = 20, y = 120)

txt2 = Entry(window, width = 10, font = 'Batang 20')

txt2.place(x = 190, y = 120)

btn = Button(window, text = "Добавить", font = 'Batang 20', command=clicked)

btn.place(x = 105, y = 170)

lbl3 = Label(window)

lbl4 = Label(window)

lbl3.configure(font = 'Batang 20', text = ' Дата', bg = 'lightskyblue')

lbl3.place(x = 25, y = 281)

lbl4.configure(font = 'Batang 20', text = ' Сумма', bg = 'lightskyblue')

lbl4.place(x = 200, y = 280)

txt3 = Entry(window, width = 10, font = 'Batang 20')

txt3.place(x = 20, y = 320)

txt4 = Entry(window, width = 10, font = 'Batang 20')

txt4.place(x = 190, y = 320)

btn1 = Button(window, text = "Удалить", font = 'Batang 20', command=clicked1)

btn1.place(x = 105, y = 370)

Модуль «Gonka»

''' В этом модуле определяется какой автомобиль быстрее '''

from tkinter import \*

import tkinter as tk

import ctypes

def drag():

window = Tk()

window.resizable(False, False)

window.title("Дрэг-рейсинг")

window.geometry('1100x375')# Размер окна

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

canvas = Canvas(window, width = user32.GetSystemMetrics(0), height = user32.GetSystemMetrics(1))

canvas.configure(bg = 'lightskyblue')# Фон

canvas.pack()

def clicked():

lb = Label(window)

b = 0

ba = 0

if int(txt3.get())/int(txt2.get()) < int(txt8.get())/int(txt7.get()):# Присвоение баллов

b += 1

if int(txt3.get())/int(txt2.get()) > int(txt8.get())/int(txt7.get()):

ba += 1

if int(txt4.get()) > int(txt9.get()):

b += 1

if int(txt4.get()) < int(txt9.get()):

ba += 1

if int(txt5.get()) > int(txt10.get()):

ba += 1

if int(txt5.get()) < int(txt10.get()):

b += 1

if b > ba:

text = Text(window)

vod = 'Быстрее ' + txt1.get()

text.insert(INSERT, vod)

text.configure(height = '1', font = 'Batang 20', width = '20')

text.place(x = 395, y = 300)

if b < ba:

text = Text(window)

vod = 'Быстрее ' + txt6.get()

text.insert(INSERT, vod)

text.configure(height = '1', font = 'Batang 20', width = '20')

text.place(x = 395, y = 300)

if b == ba:

text = Text(window)

vod = 'Ровно'

text.insert(INSERT, vod)

text.configure(height = '1', font = 'Batang 20', width = '20')

text.place(x = 395, y = 300)

b1 = Label(window)# Вывод в окно "window"

b2 = Label(window)

b3 = Label(window)

b4 = Label(window)

b5 = Label(window)

b1.configure(text = 'Авто', font = 'Batang 20', bg = 'lightskyblue')# Метки

b1.place(x = 145, y = 80)

b2.configure(text = 'Л.с.', font = 'Batang 20', bg = 'lightskyblue')

b2.place(x = 340, y = 80)

b3.configure(text = 'Масса', font = 'Batang 20', bg = 'lightskyblue')

b3.place(x = 505, y = 80)

b4.configure(text = 'Крутящий\nмомент', font = 'Batang 20', bg = 'lightskyblue')

b4.place(x = 680, y = 55)

b5.configure(text = 'Высота\nавто', font = 'Batang 20', bg = 'lightskyblue')

b5.place(x = 885, y = 55)

txt1 = Entry(window, width = 10, font = 'Batang 20')

txt1.place(x = 95, y = 120)

txt2 = Entry(window, width = 10, font = 'Batang 20')

txt2.place(x = 285, y = 120)

txt3 = Entry(window, width = 10, font = 'Batang 20')

txt3.place(x = 475, y = 120)

txt4 = Entry(window, width = 10, font = 'Batang 20')

txt4.place(x = 665, y = 120)

txt5 = Entry(window, width = 10, font = 'Batang 20')

txt5.place(x = 855, y = 120)

txt6 = Entry(window, width = 10, font = 'Batang 20')

txt6.place(x = 95, y = 170)

txt7 = Entry(window, width = 10, font = 'Batang 20')

txt7.place(x = 285, y = 170)

txt8 = Entry(window, width = 10, font = 'Batang 20')

txt8.place(x = 475, y = 170)

txt9 = Entry(window, width=10, font = 'Batang 20')

txt9.place(x = 665, y = 170)

txt10 = Entry(window, width = 10, font = 'Batang 20')

txt10.place(x = 855, y = 170)

btn = Button(window, text = "Клик!", command = clicked, font = 'Batang 20')

btn.place(x = 510, y = 220)

Модуль «sklad»

''' В этом модуле информация о том что есть в наличии на складе '''

from tkinter import \*

import tkinter as tk

import random

from tkinter import scrolledtext

import ctypes

def sklad():

window = Tk()

window.resizable(False, False)

window.title("Склад")# Название окна

window.geometry('1100x700')# Размер окна

user32 = ctypes.windll.user32

widthButtonPlace = (user32.GetSystemMetrics(0) // 15)

widthButton = widthButtonPlace // 7

canvas = Canvas(window, width = user32.GetSystemMetrics(0), height = user32.GetSystemMetrics(1))# Ширина и длина

canvas.configure(bg = 'lightskyblue')

canvas.pack()

f = open('skl.txt', 'r', encoding = 'utf-8')# Для чтение русских букв

text = Text()

text = scrolledtext.ScrolledText(window, height = '17', font = 'Batang 20', width = '40')

text.insert(INSERT, f.read())

text.place(x = 425, y = 80)

f.close()# Закрываем файл

def clicked\_dobavlenie():# Добавляем файл

z = open('skl.txt', 'r', encoding = 'utf-8')

y = z.read()

x = y.split()

t = []

for i in range (len(x)):

if i%2 == 0:

u = x[i] + ' ' + x[i+1]#Делаем строку

t.append(u)

b = txt1.get()

b1 = int(txt2.get())

v = open('skl.txt','w', encoding = 'utf-8')# Чтение русских букв

for i in range(len(x)):

j = t[i].split()

if i%2 == 0 and j[0] != b:

v.write(j[0])

v.write(' ')

v.write(j[1])

v.write('\n')

if j[0] == b and i%2 == 0:

v.write(j[0])

l = int(j[1]) + b1

v.write(' ')

v.write(str(l))

v.write('\n')

v.close()

f = open('skl.txt', 'r', encoding = 'utf-8')

text = Text()

text = scrolledtext.ScrolledText(window, height = '17', font = 'Batang 20', width = '40')# font - шрифт

text.insert(INSERT, f.read())

text.place(x = 425, y = 80)

f.close()

def clicked\_delete():# Удаляем файл

z = open('skl.txt', 'r', encoding = 'utf-8')

y = z.read()

x = y.split()

t = []

for i in range (len(x)):

if i%2 == 0:

u = x[i] + ' ' + x[i+1]

t.append(u)

b = txt3.get()

b1 = int(txt4.get())

v = open('skl.txt','w', encoding = 'utf-8')

for i in range(len(x)):

j = t[i].split()

if i%2 == 0 and j[0] != b:

v.write(j[0])

v.write(' ')

v.write(j[1])

v.write('\n')

if j[0] == b and i%2 == 0:

v.write(j[0])

l = int(j[1]) - b1

v.write(' ')

v.write(str(l))

v.write('\n')

v.close()

f = open('skl.txt', 'r', encoding = 'utf-8')# Читаем файл

text = Text()

text = scrolledtext.ScrolledText(window, height = '17', font = 'Batang 20', width = '40')

text.insert(INSERT, f.read())

text.place(x = 425, y = 80)

f.close()# Закрываем файл

lbl1 = Label(window)

lbl2 = Label(window)

lbl1.configure(font = 'Batang 20', text = 'Название\nтовара', bg = 'lightskyblue')

lbl1.place(x = 45, y = 51)

lbl2.configure(font = 'Batang 20', text = 'Кол-во', bg = 'lightskyblue')

lbl2.place(x = 225, y = 80)

txt1 = Entry(window, width = 10, font = 'Batang 20')

txt1.place(x = 20, y = 120)

txt2 = Entry(window, width = 10, font = 'Batang 20')

txt2.place(x = 190, y = 120)

btn = Button(window, text = "Добавить", font = 'Batang 20', command = clicked\_dobavlenie)

btn.place(x = 105, y = 170)

lbl3 = Label(window)

lbl4 = Label(window)

lbl3.configure(font = 'Batang 20', text = 'Название\nтовара', bg = 'lightskyblue')

lbl3.place(x = 45, y = 251)

lbl4.configure(font = 'Batang 20', text = 'Кол-во', bg = 'lightskyblue')

lbl4.place(x = 225, y = 280)

txt3 = Entry(window, width = 10, font = 'Batang 20')

txt3.place(x = 20, y = 320)

txt4 = Entry(window, width = 10, font = 'Batang 20')

txt4.place(x = 190, y = 320)

btn1 = Button(window, text = "Удалить", font = 'Batang 20', command = clicked\_delete)

btn1.place(x = 105, y = 370)