**МИНОБРНАУКИ РОССИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ**

**УНИВЕРСИТЕТ им. В.Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Институт ЭИТУС

Кафедра информационных технологий

**Отчет о прохождении преддипломной практики**

Выполнил:

студент группы ИТ-42

Курбатова Софья Андреевна

Проверил:

и. о. зав. кафедрой ИТ, канд. техн. наук

Старченко Денис Николаевич

Белгород 2022

Содержание

[Введение 3](#_Toc76641464)

[Описание предприятия 4](#_Toc76641465)

[Аппаратные и программные средства предприятия 5](#_Toc76641466)

[Выполняемые задания 6](#_Toc76641467)

[Описание индивидуального задания 6](#_Toc76641468)

[Выполнение индивидуального задания 7](#_Toc76641469)

[1. Закладка «Персональные данные», блок «Место рождения» 7](#_Toc76641470)

[2. Закладка «Персональные данные», блок «Сведения об инвалидности» 7](#_Toc76641471)

[3. Закладка «Персональные данные», блок «Сведения об инвалидности» 8](#_Toc76641472)

[4. Закладка «Кадры», блок «Воинский учет» 9](#_Toc76641473)

[Заключение 10](#_Toc76641474)

**Введение**

Основной целью производственной практики является практическое приложение и закрепление знаний, приобретенных в процессе лекционных, лабораторных занятий, а также самостоятельной работы студента. Она способствует получению представления об этике профессиональной коммуникации, получению опыта принятия оперативных управленческих решений и аргументации.

Производственная практика представляет собой важнейшую составную часть учебного процесса по подготовке специалистов, так как способствует повышению общего уровня профессиональной подготовки. Она может проводиться в структурных подразделениях (учебных мастерских, лабораториях, компьютерных классах и т.д.) университета, а также на предприятиях, в учреждениях и организациях.

В процессе прохождения практики студенты должны проявлять самостоятельность, трудоспособность, нарабатывать навыки коммуникативного общения в коллективе.

**Описание предприятия**

Общество с ограниченной ответственностью «Сайнер» (ООО «Сайнер») — системный интегратор для предприятий энергетики и ЖКХ, лидер в реализации проектов создания ERP-систем, входит в «Консист Бизнес Групп». Важнейшее направление деятельности — автоматизация основных процессов энергетических компаний. Более 12 лет специалисты компании оказывают услуги управленческого консалтинга, проектирования и внедрения информационных систем, их последующего сопровождения. С начала своей деятельности «Сайнер» сотрудничает с «САП АГ» (SAP SE) — мировым лидеров в производстве программного обеспечения.

ТУРБО — одно из ключевых направлений бизнеса «Консист Бизнес Групп», входящего в ГК ЛАНИТ, лидера на рынке российского программного обеспечения для управления предприятиями и холдингами. История ТУРБО началась в 1988 году с разработки программы «Турбо Бухгалтер», которая быстро распространилась в бизнес-среде и получила признание многих компаний на молодом отечественном рынке благодаря качеству архитектуры, скорости работы, возможности масштабироваться под потребности растущего бизнеса. В 2007 году платформа Турбо9 завоевала популярность у пользователей. Архитектура ТУРБО обеспечивает открытость прикладных решений, высокую функциональность, гибкость и масштабируемость и может использоваться в компаниях разного масштаба. Программная платформа ТУРБО включена в Единый реестр российских программ для ЭВМ и БД.

Решения на базе ТУРБО широко распространены в России и странах СНГ и успешно применяются во многих странах мира, например, в Китае, Венгрии, Казахстане, Кипре и др

На базе платформы ТУРБО создано множество отраслевых и специализированных решений, успешно работающих в различных индустриях и сферах экономики: в промышленности, банковском секторе и страховании, государственном управлении и образовательной сфере, в транспортной отрасли, здравоохранении, энергетике, телекоммуникациях, нефтегазовой отрасли, легкой промышленности, торговле, сфере профессиональных услуг и других. Бизнес-приложения на платформе можно создавать на одном из трех уровней ‒ уровне разработки на объектно-ориентированном языке, более высоком уровне компоновки учетной системы из преднастроенных подходов, классов и объектов, ставших результатом многолетнего опыта создания учетных систем для российского рынка, а также уровне моделирования бизнес-процессов за счет использования больших блоков-процессов.

Разработка на ТУРБО обладает рядом следующих преимуществ:

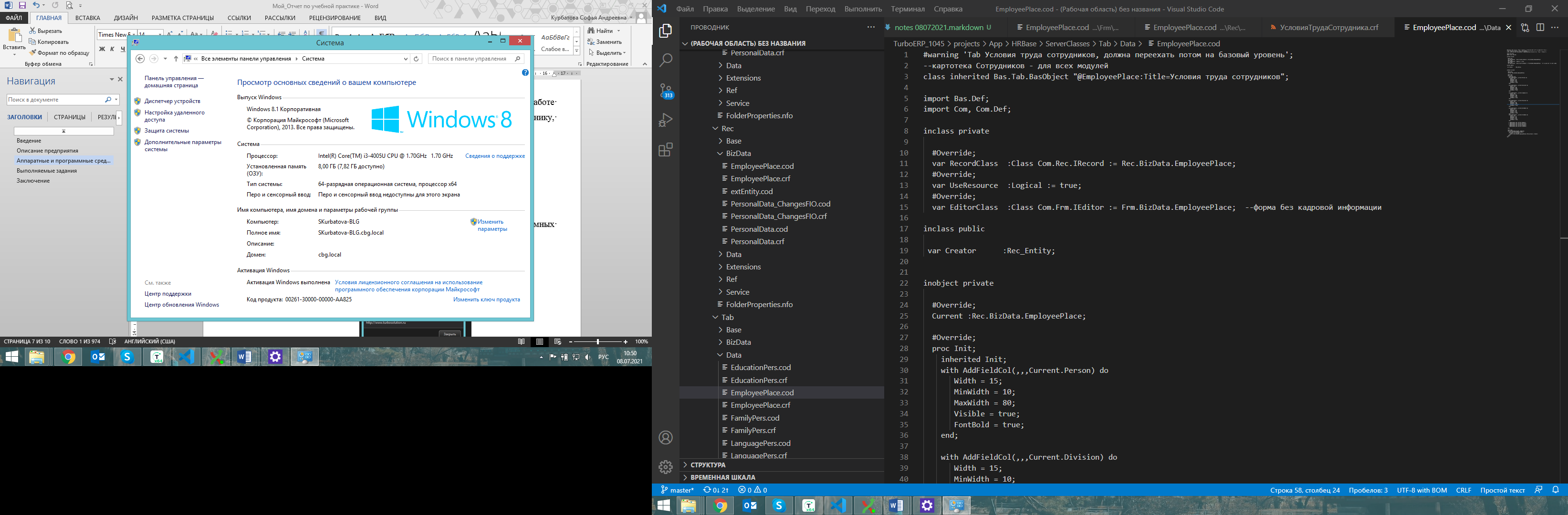
* Интегрированная среда разработки – удобный редактор исходного кода, встроенный отладчик с возможностью отладки серверных скриптов, несколько уровней логирования;
* Визуальный редактор форм, который подразумевает такие настройки форм как: размер, видимость, цвет, шрифт, выравнивание, группировка, подключение различных компонент и полей, наследование форм;
* Интегрированная шина и ETL – возможность встроить систему на платформе в любую ИТ-инфраструктуру и наладить двухсторонний обмен данными;
* Встроенный язык ТБ.Скрипт, в котором доступны замыкания, доступ к локальным переменным внешних функций из замыканий, «слабые» ссылки на объект, поддержка getters/setters.

Решения на платформе ТУРБО X регулярно дополняются и обновляются в соответствии с новыми требованиями законодательства и разработками вендора и партнеров.

**Аппаратные и программные средства предприятия**

В настоящее время большая часть организаций использует в работе современные автоматизированные системы и компьютерную технику, программное обеспечение и носители информации.

В работе используется Ноутбук HP ProBook сведения о котором представлены на рисунке рисунок 1).



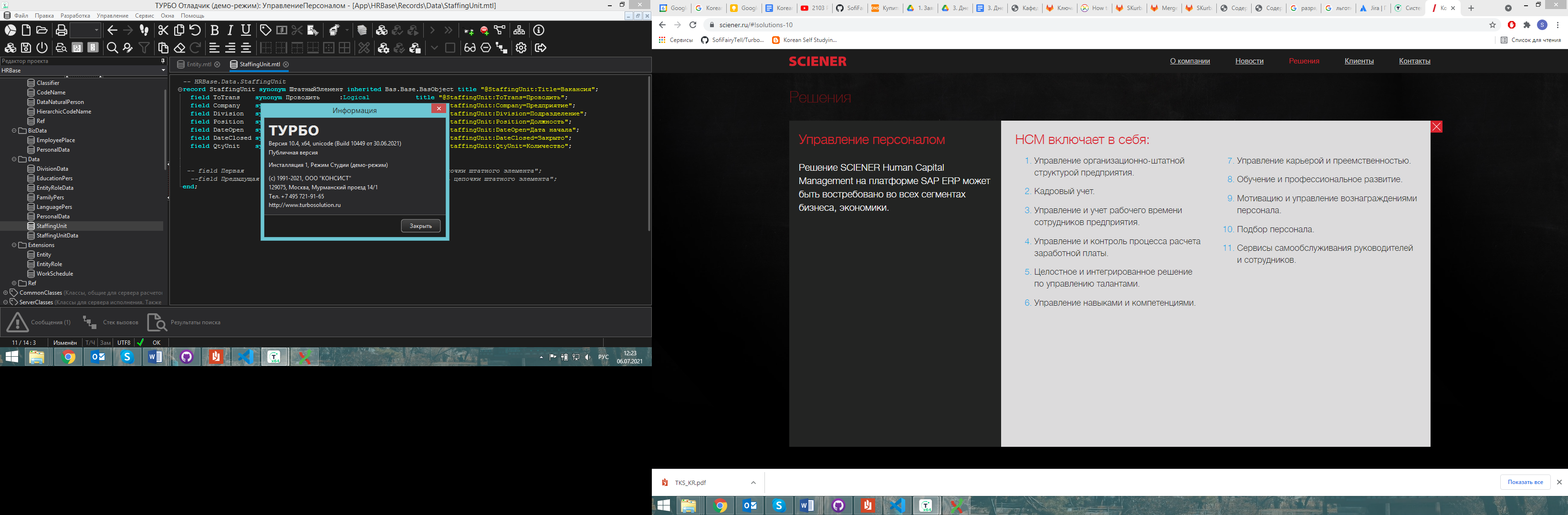
1. Основные сведения о компьютере

– Мышь Oklick 145M Black (1000 dpi) USB ;

В процессе работы использует следующий перечень программных средств:

– **ТУРБО 10.4 –** программа, реализованная в виде клиент-серверного приложения и поддерживающая два типа интерфейса пользователя (см. рисунок 2). Взаимодействие с серверной частью может осуществляться через десктопный клиент (исполняемый файл для ОС Windows) или через браузер семейства Chrome или FireFox.

Сервер баз данных работает под управлением СУБД MS SQL. Серверная часть Программы может быть установлена на ОС семейства Windows и состоит из следующих компонентов: сервера данных, расчетов, процедур (приложений), WEBсервера и сервера лицензирования.



1. Информация о версии информационной базы

**– Git Extensions -** визуальный клиент системы управления версиями Git, позволяющий использовать Git без использования консольного интерфейса.

Также на ПК установлены следующие программные продукты:

– 7-zip 19.00;

– Adobe Acrobat Reader DC Russian;

– Microsoft Office профессиональный плюс 2010;

– VLC media player;

– WinDjView 2.1;

# Выполняемые задания

В состав программного комплекса входит технологическое ядро, позволяющее разрабатывать различные прикладные проекты на основе Базовой модели приложений ТУРБО. Система поставляется как совокупность серверных, интеграционных и клиентских компонентов. При запуске локальной версии все компоненты системы располагаются в одном адресном пространстве приложения. Программное обеспечение функционально разделено на системную и прикладную часть, в связи с чем программа функционирует в двух основных режимах работы:

* Режим разработки – называемый также режимом Студии, этот режим используется для разработки, отладки и настройки прикладных проектов. Для обеспечения этих функций используется программное средство среды разработки ТУРБО X Студия.
* Режим исполнения – называемый также режимом сессии, используется пользователями при эксплуатации готовых прикладных проектов (разработанных и отлаженных), которые не требуют от пользователей какого-либо программирования и предоставляют готовые решения для конкретной отрасли.

Клиент ТУРБО подключается к серверу данных ТУРБО X, а тот, в свою очередь, подключается к СУБД. Система работает со следующими СУБД: MS SQL Sever, Interbase, Firebird, Sybase SQL, MySQL, Linter, PostgreSQL, Dbase. Типичное приложение, построенное на Машине Компонентов ТУРБО Х имеет определенную структуру и включает в себя несколько групп классов, каждая из которых объединяет классы определенного назначения:

* MTL-описания;
* классы-интерфейсы записей (обертки MTL-описаний) или Rec-классы;
* классы-формы для редактирования документов (записей) или Frm-классы;
* классы-таблицы документов (записей) или Tab-классы; структуры учета;
* учетные операции; классы–отчеты или Rep-классы.
* команды, панели управления и ролевых центры

Rec-, Frm-, Tab- и Rep-классы реализуются как серверные, для их хранения в каждом проекте рекомендуется создавать папку ServerClasses с четырьмя соответствующими подпапками — Rec, Frm, Tab и Rep.

## Описание индивидуального задания

Необходимо осуществить следующую доработку интерфейса карточки «Физ.лицо (Сотрудник)» и произвести корректировку полей и таблиц карточки в соответствии с параметрами и требованиями системы ТУРБО:

1. Закладка «Персональные данные», блок «Место рождения»: удалить слово «рождения» из наименования полей «Страна рождения», «Область рождения», «Район рождения». Наименования полей, которые должны содержаться в блоке «Место рождения»: «Тип места рожд.», «Страна», «Область», «Район», «Город»;

2. Закладка «Персональные данные», блок «Сведения об инвалидности»: предусмотреть возможность выбора из перечня значений. Значения: пустое, 1 группа, 2 группа, 3 группа. По умолчанию поле остается пустым, пользователь вручную осуществляет выбор одного из возможных значений;

3. Закладка «Персональные данные», блок «Сведения об инвалидности»: поля «Серия» и «Номер» таблицы «Сведения об инвалидности». В шапке таблицы необходимо сделать единое поле Документ над полями Серия и Номер

4. Закладка «Кадры», блок «Воинский учет»: в таблице оставить столбцы: Дата, Воинская обязанность (убрать слово отношение), Воинский учет (выводить краткое наименование в таблице (состоит, не состоит, снят по возрасту, снят по здоровью)), воинское звание, состав/профиль.

## Выполнение индивидуального задания

### 1. Закладка «Персональные данные», блок «Место рождения»

Для предоставления пользователю возможности редактировать или вносить изменения в БД используются классы-формы для редактирования документов. Изменение отображаемого ими интерфейса возможно двумя способами: дизайн-режим и изменение crf-файла с описанием компонентов формы. Поэтому в изменения были внесены в файл КарточкаФЛ.crf, фрагмент которых представлен на листинге 1. Это позволило изменить внешний вид формы не меняя при этом описание исходной таблицы.

*Листинг 1: изменение в файле КарточкаФЛ.crf*.

<Component Name="бксМестоРождения" Class="Com.Ctrl.IBox" After="bxIdentityDocument" Caption="@Место рождения" Collapsed="True" HasCaption="True">

<Component Name="грМестоРождения" Class="Com.Ctrl.IGroup" Adaptive="True" ByColumn="True" MaxWidth="230" MinWidth="150">

<Component Name="ТипМестаРождения" Class="Com.Ctrl.IEdit" Alignment="LeftAlign" Caption="Тип места рожд." CellFormat="EnumFormat" Field="EdtRec.ТипМестаРождения" HasCaption="True"/>

<Component Name="СтранаРождения" Class="Com.Ctrl.IEdit" Field="EdtRec.СтранаРождения" Caption="Страна" HasCaption="True"/>

<Component Name="РегионРождения" Class="Com.Ctrl.IEdit" Field="EdtRec.РегионРождения" Caption="Область" HasCaption="True"/>

<Component Name="РайонРождения" Class="Com.Ctrl.IEdit" Field="EdtRec.РайонРождения" Caption="Район" HasCaption="True"/>

<Component Name="МестоРождения" Class="Com.Ctrl.IEdit" Field="EdtRec.МестоРождения" Caption="Город" HasCaption="True"/>

</Component>

</Component>

Внешний вид до внесенных изменений и после представлен на рисунке ниже.



1. Внешний вид карточки до/после изменений

### 2. Закладка «Персональные данные», блок «Сведения об инвалидности»

В файл ТипыСписки.cod, содержимое которого представлено описанием типов используемых в приложении списков, был добавлен новый список, описание которого содержится в листинге 2. В файл КарточкаФЛ.cod с классом frm была добавлена процедура для отображения добавленного списка в процессе изменения данных в форме. Добавленная процедура описана в листинге 3.

*Листинг 2: Изменение в файле ТипыСписки.cod*

--ФизЛицо КарточкаФЛ Группа Инвалидности

type ТипГрИнвалидности = (тгрНеЗадан = 0,

тгр1 = 1,

тгр2 = 2,

тгр3 = 3);

func ТипГрИнвалидностиСписок :String[];

Result = [Voc.Txt("@<Не указан>") + "|"+Str(тгрНеЗадан)

,Voc.Txt("@грп 1") + "|"+Str(тгр1)

,Voc.Txt("@грп 2") + "|"+Str(тгр2)

,Voc.Txt("@грп 3") + "|"+Str(тгр3)

];

end;

### 3. Закладка «Персональные данные», блок «Сведения об инвалидности»

В шапке таблицы было добавлено единое поле Документ над полями Серия и Номер. Для этого была изменена описанная в листинге 3 процедура для отображения списка групп. Для создания группирующего поля используется функция AddColGroup .

*Листинг 3: Процедура для отображения списка групп Инвалидности*

proc ДобавитьБксИнвалидность;

тблИнвалидность = EdtRec.Инвалидность.DefStruct;

with фрмИнвалидность do

Subtable = EdtRec.Инвалидность;

Numbering = False;

with AddColGroup('Период действия') do

AddFieldCol("с", 30, тблИнвалидность.ДатаНачала);

AddFieldCol("по", 30, тблИнвалидность.ДатаОкончания);

end;

with AddColGroup('Документ') do

AddFieldCol("Серия", 30, тблИнвалидность.СерияСправки);

AddFieldCol("Номер", 30, тблИнвалидность.НомерСправки);

end;

with AddFieldCol("Группа", 20, тблИнвалидность.Группа)as IEdit do

CellFormat = Consts.EnumFormat;

Alignment = Consts.LeftAlign;

FillList(ТипГрИнвалидностиСписок);

end;

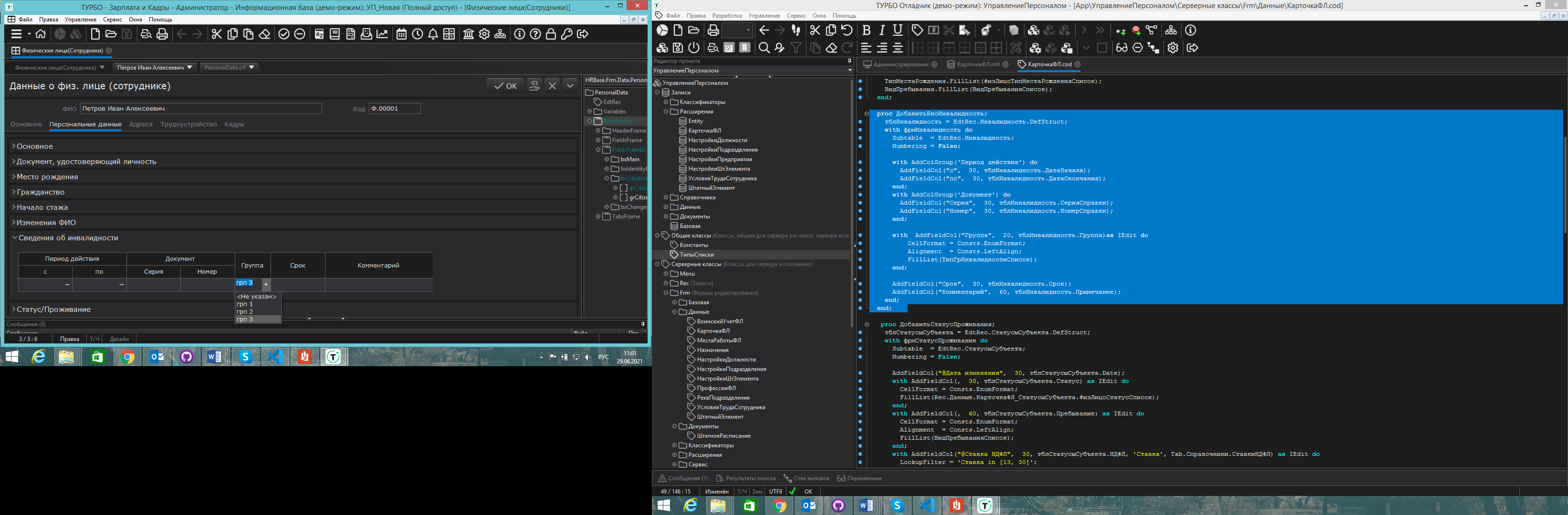
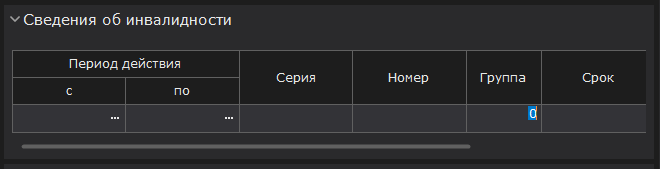
AddFieldCol("Срок", 30, тблИнвалидность.Срок);

AddFieldCol("Комментарий", 60, тблИнвалидность.Примечание);

end;

end;

Внешний вид до внесенных изменений и после представлен на рисунке 4.

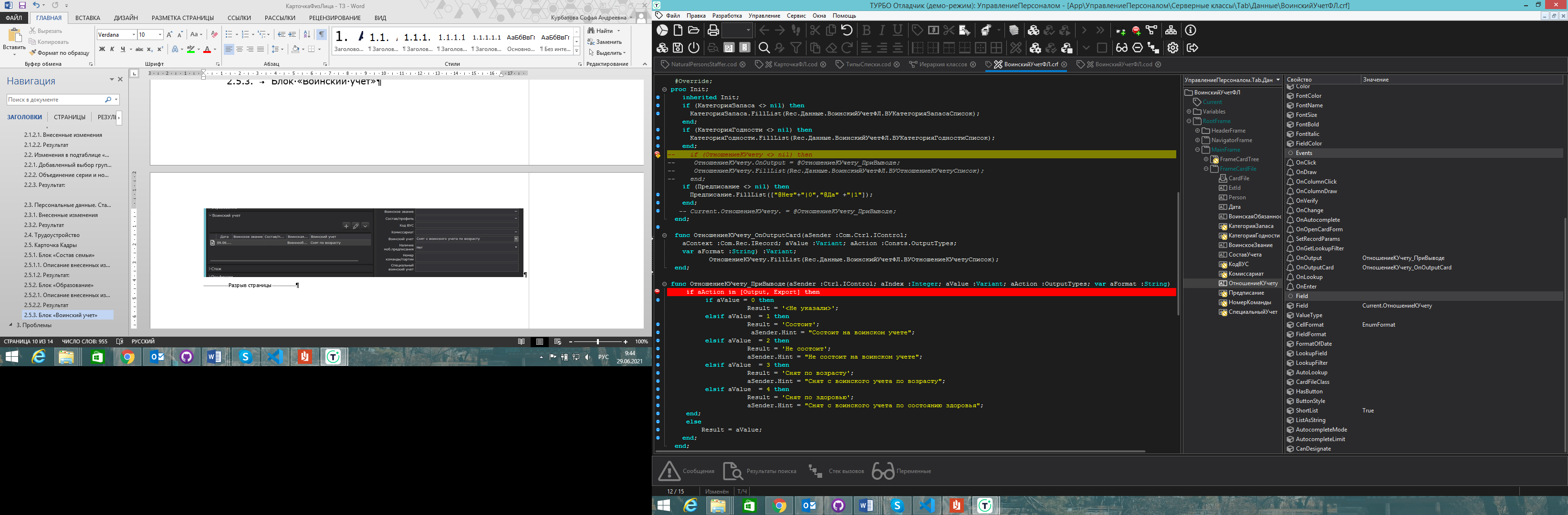


1. Внешний вид карточки до/после изменений

### 4. Закладка «Кадры», блок «Воинский учет»

В таблице оставлены столбцы: Дата, Воинская обязанность (убрано слово отношение). Для столбца Воинский учет осуществляется вывод краткого наименования в таблице (состоит, не состоит, снят по возрасту, снят по здоровью)). Оставлены столбцы воинское звание, состав/профиль. Результат представлен на рисунке 6.

В процессе изменения таблицы был использован дизайн-режим. Пример демонстрирующий его работу представлен на рисунке 5, где желтым символом обозначены поля, которые были исключены из отображения в таблице «Воинский учет».



1. Исключенные поля

Для того, чтобы в таблице происходило отображение краткого наименования, а в соответствующей карточке сохранялось полное наименование категории воинского учета, была написана функция, реагирующая на изменение поля Воинский учет на форме:

*Листинг 4: Функция для вывода краткого наименования*

func ОтношениеКУчету\_OnOutputCard(aSender :Com.Ctrl.IControl;

aContext :Com.Rec.IRecord; aValue :Variant; aAction :Consts.OutputTypes;

var aFormat :String) :Variant;

if aAction in [Output, Export, Copy] then

if aValue = 0 then

Result = '<Не указали>';

elsif aValue = 1 then

Result = 'Состоит';

elsif aValue = 2 then

Result = 'Не состоит';

elsif aValue = 3 then

Result = 'Снят по возрасту';

elsif aValue = 4 then

Result = 'Снят по здоровью';

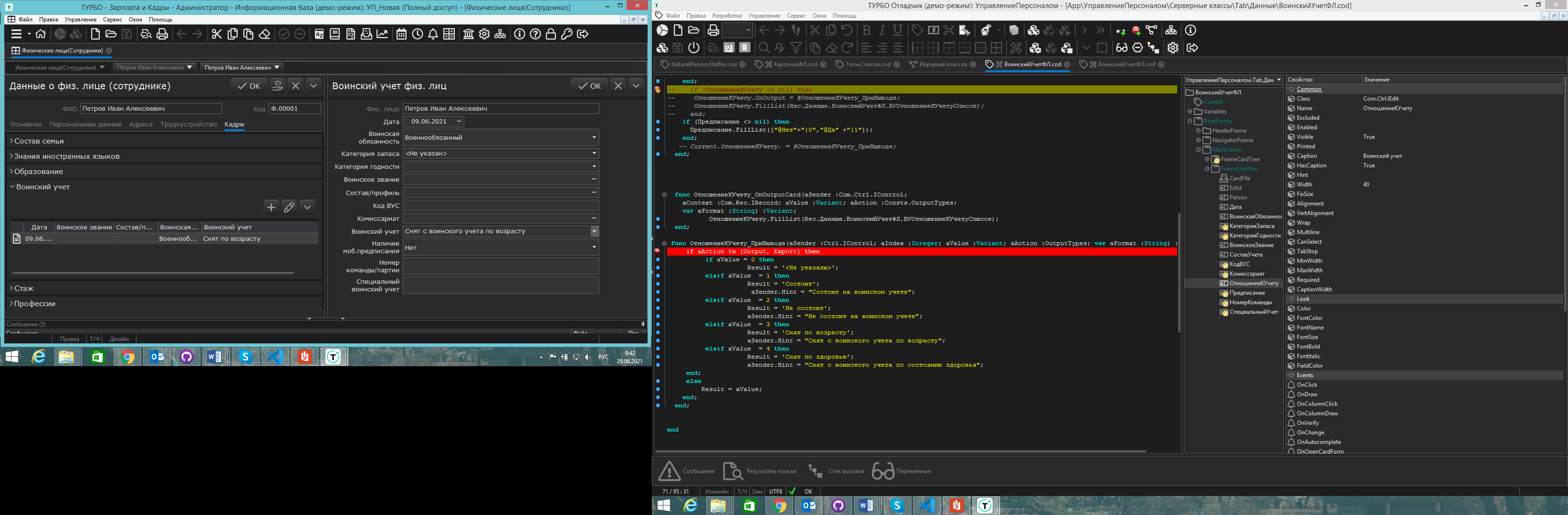
end;

else

Result = aValue;

end;

end;



1. Внешний вид карточки после изменений

# Заключение

В процессе прохождения практики в ООО «Сайнер» было осуществлено практическое приложение знаний, приобретенных в процессе лекционных и лабораторных занятий, а также была изучена организационная структура предприятия. Так как производственная практика способствует повышению общего уровня профессиональной подготовки, то был приобретены следующие профессиональные навыки:

– составления отчетной документации;

– консультации пользователей информационной системы и разработки фрагментов методики обучения пользователей информационной системы;

– обеспечение организации доступа пользователей информационной системы;

Практическая полезность производственной практики состоит в том, что были приобретены навыки разработчика программного обеспечения. Были приобретен навык коммуникации, принятия правильного и быстрого решения в нестандартных ситуациях. В процессе были развиты аналитические способности, и знания из области теории были перенесены на самостоятельное решение поставленных задач.