



# **ИНСТРУКЦИЯ**

Учет больничных листов



## Настройка рабочего места

Для того, чтобы в клиентской части интерфейса (слева) отображались панели «Номерки», «График», «ЛУД» (Лист уточненных диагнозов) необходимо в пункте меню «Настройки» установить соответствующие атрибуты (рис. 1).

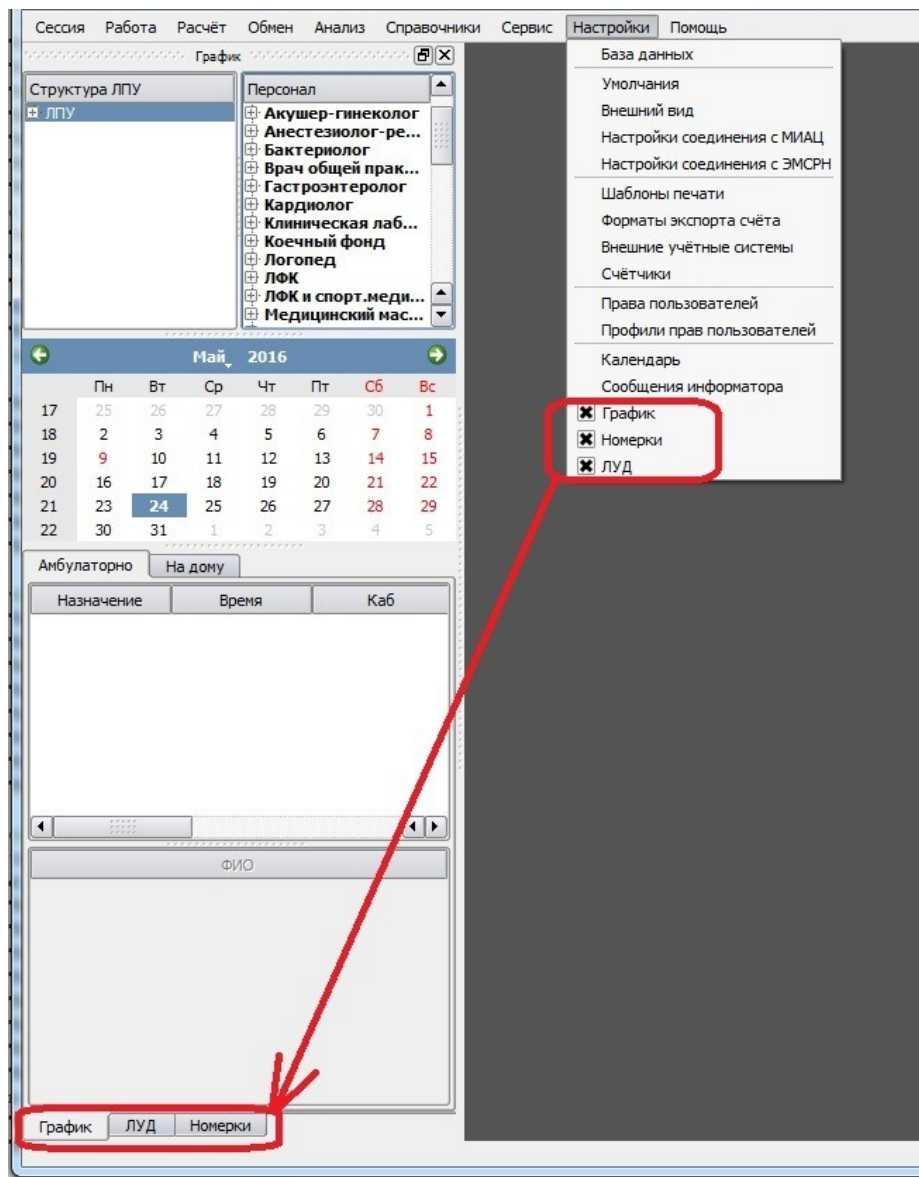


Рисунок 1 – Настройка отображения панелей «График», «ЛУД», «Номерки»

Для того чтобы начать работу нужно зайти в «Работа» - «Обслуживание пациентов» (рис. 2).

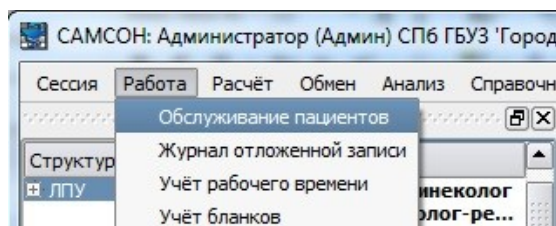


Рисунок 2 – Обслуживание пациентов

Откроется картотека пациентов. Чтобы быстро найти нужного пациента можно воспользоваться фильтром, который находится справа. Рекомендуем искать пациента по дате рождения (см. рис. 3).

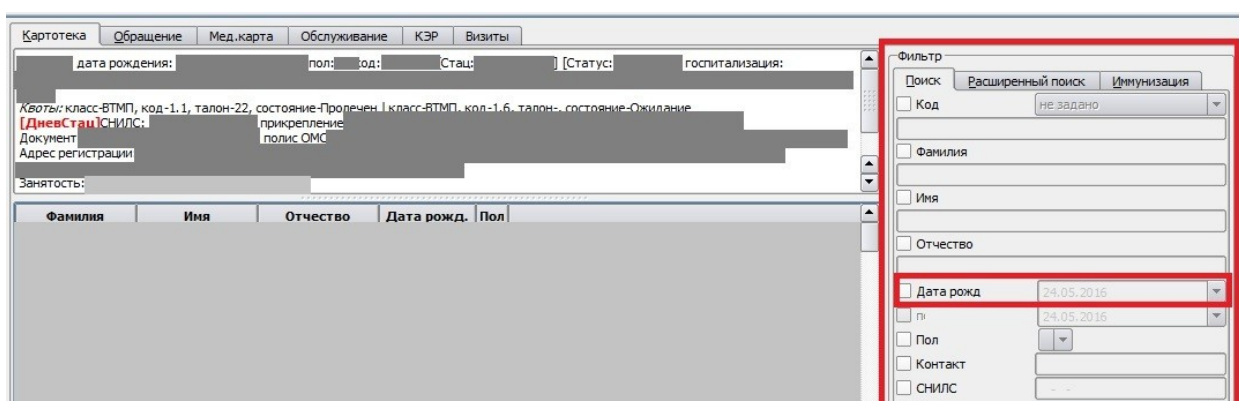


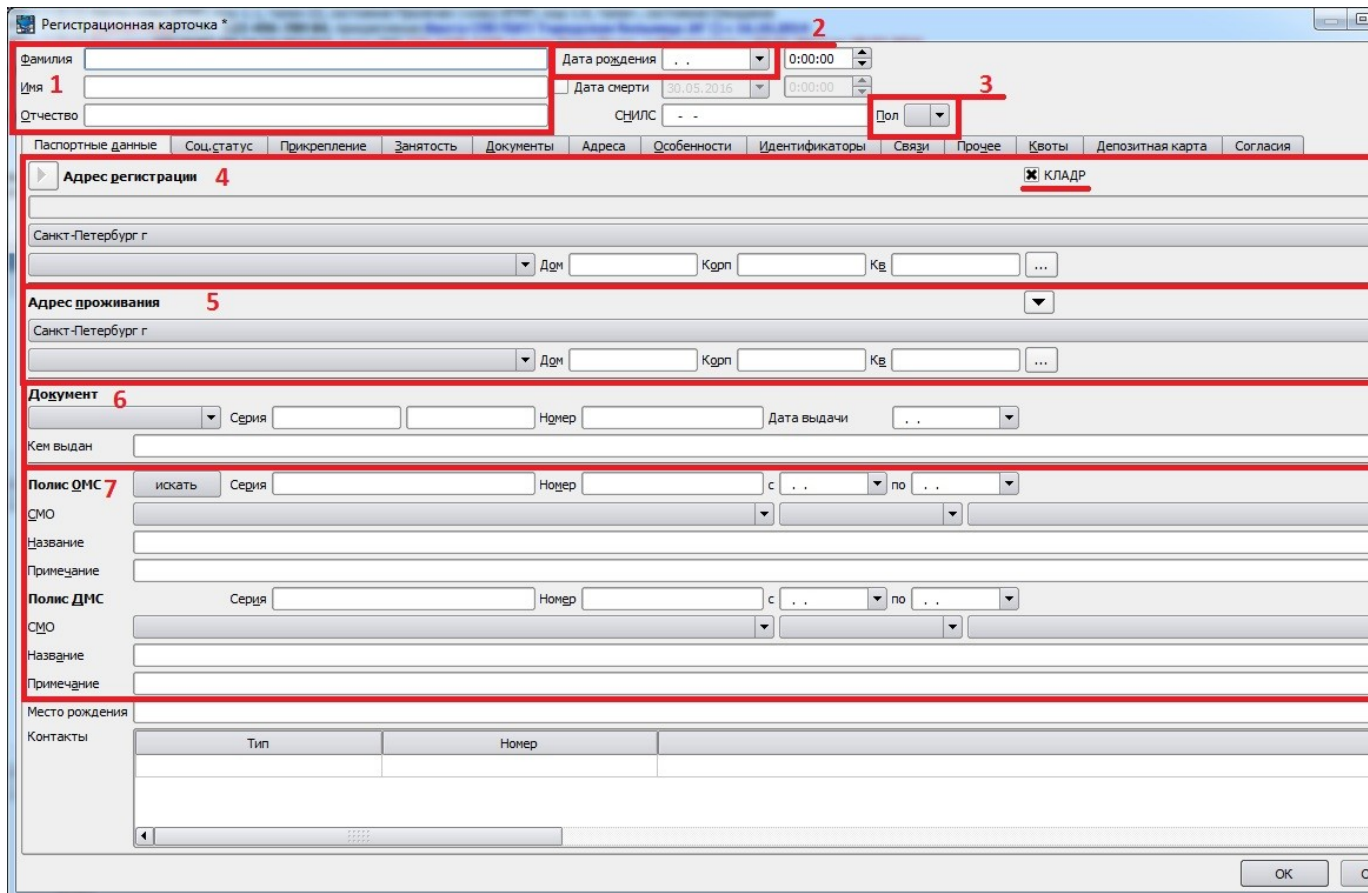
Рисунок 3 - Картотека

Вводим дату рождения и нажимаем «Enter». В картотеке отобразится список пациентов с указанной датой рождения, выбираем нужного нам пациента.

В случае, когда Пациент обратился за медицинской помощью в ЛПУ *впервые*, и его данные в картотеке ЛПУ отсутствуют, необходимо зарегистрировать Пациента (рис. 4).

*Фамилия, Имя, Отчество* Пациента (блок 1), *пол* (блок 3) и *дата рождения* (блок 2) – вот минимальный набор данных, необходимых для регистрации.

Если Пациент может указать больше сведений о себе, например: *адрес регистрации* (блок 4), *адрес проживания* (город и улица выбираются из справочника КЛАДР) (блок 5), *серия и номер документа*, удостоверяющего личность (блок 6), *данные по полису ОМС и/или полису ДМС* (блок 7) и т. д., то КПС "САМСОН" предоставляет возможность ввода этих данных.



Регистрационная карточка

Фамилия \_\_\_\_\_ Дата рождения .. 0:00:00

Имя **1** \_\_\_\_\_ Дата смерти 30.05.2016 0:00:00

Отчество \_\_\_\_\_ СНИЛС - - Пол **3**

Паспортные данные Соц.статус Прикрепление **Занятость** Документы Адреса Особенности Идентификаторы Связи Прочее Квоты Депозитная карта Соглашения

**4** Адрес регистрации  КЛАДР

Санкт-Петербург г

Дом \_\_\_\_\_ Корп \_\_\_\_\_ Кв \_\_\_\_\_

**5** Адрес проживания

Санкт-Петербург г

Дом \_\_\_\_\_ Корп \_\_\_\_\_ Кв \_\_\_\_\_

**6** Документ

Серия \_\_\_\_\_ Номер \_\_\_\_\_ Дата выдачи ..

Кем выдан \_\_\_\_\_

**7** Полис ОМС  Серия \_\_\_\_\_ Номер \_\_\_\_\_ с .. по ..

СМО \_\_\_\_\_

Название \_\_\_\_\_

Примечание \_\_\_\_\_

Полис ДМС Серия \_\_\_\_\_ Номер \_\_\_\_\_ с .. по ..

СМО \_\_\_\_\_

Название \_\_\_\_\_

Примечание \_\_\_\_\_

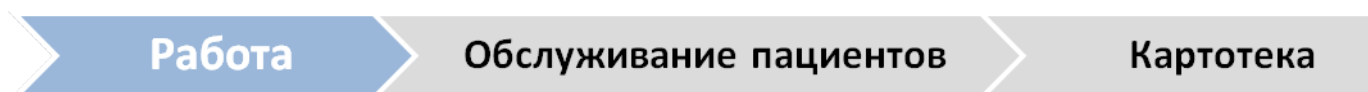
Место рождения \_\_\_\_\_

Контакты

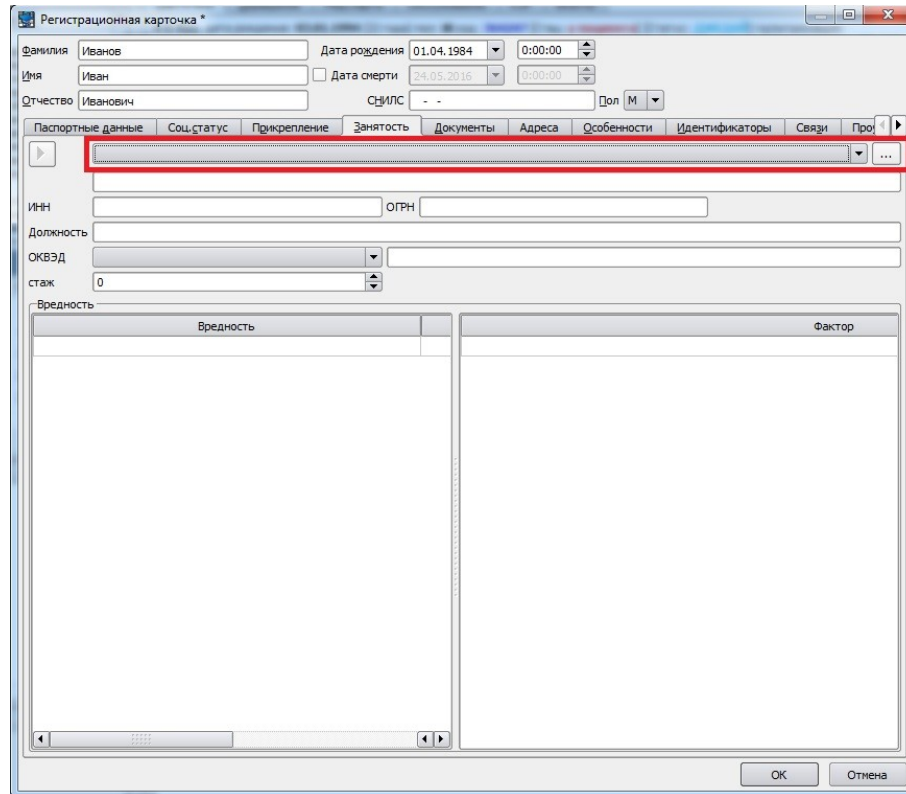
Тип	Номер

OK

Рисунок 4 – Создание новой регистрационной карточки



На вкладке «**Занятость**» (рис. 5) указываем организацию, где работает пациент либо выбираем ее из справочника.



Регистрационная карточка \*

Фамилия Иванов Дата рождения 01.04.1984 0:00:00  
 Имя Иван Дата смерти 24.05.2016 8:00:00  
 Отчество Иванович СНИЛС - - Пол М

Паспортные данные Соц.статус Прикрепление **Занятость** Документы Адреса Особенности Идентификаторы Связи Про

ИИН \_\_\_\_\_ ОГРН \_\_\_\_\_  
 Должность \_\_\_\_\_  
 ОКВЭД \_\_\_\_\_  
 стаж 0

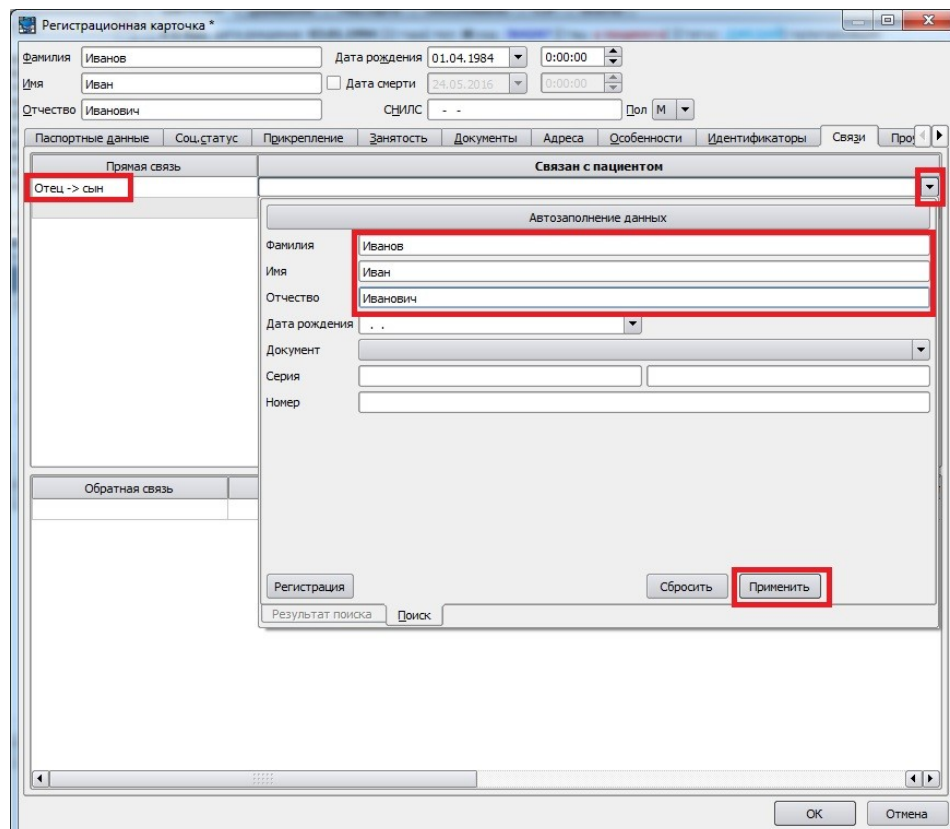
Вредность

Вредность	Фактор

OK Отмена

Рисунок 5 – Занятость пациента

На вкладке «Связи» (рис. 6) указываем связи данного пациента с другими пациентами.



Регистрационная карточка \*

Фамилия Иванов Дата рождения 01.04.1984 0:00:00  
 Имя Иван Дата смерти 24.05.2016 8:00:00  
 Отчество Иванович СНИЛС - - Пол М

Паспортные данные Соц.статус Прикрепление Занятость Документы Адреса Особенности Идентификаторы **Связи** Про

Прямая связь **Связан с пациентом**  
 Отец -> сын

Автозаполнение данных

Фамилия Иванов  
 Имя Иван  
 Отчество Иванович  
 Дата рождения . . .  
 Документ \_\_\_\_\_  
 Серия \_\_\_\_\_  
 Номер \_\_\_\_\_

Обратная связь

Регистрация Сбросить **Применить**

Результат поиска Поиск

OK Отмена

Рисунок 6 – Связи

## Регистрация больничного листа

## По основному месту работы

Для регистрации больничного листа переходим на вкладку **КЭР**, нажимаем внизу окна кнопку «Регистрация» или клавишу F9 на клавиатуре (рис. 7).

## Работа

## Обслуживание пациентов

## КЭР

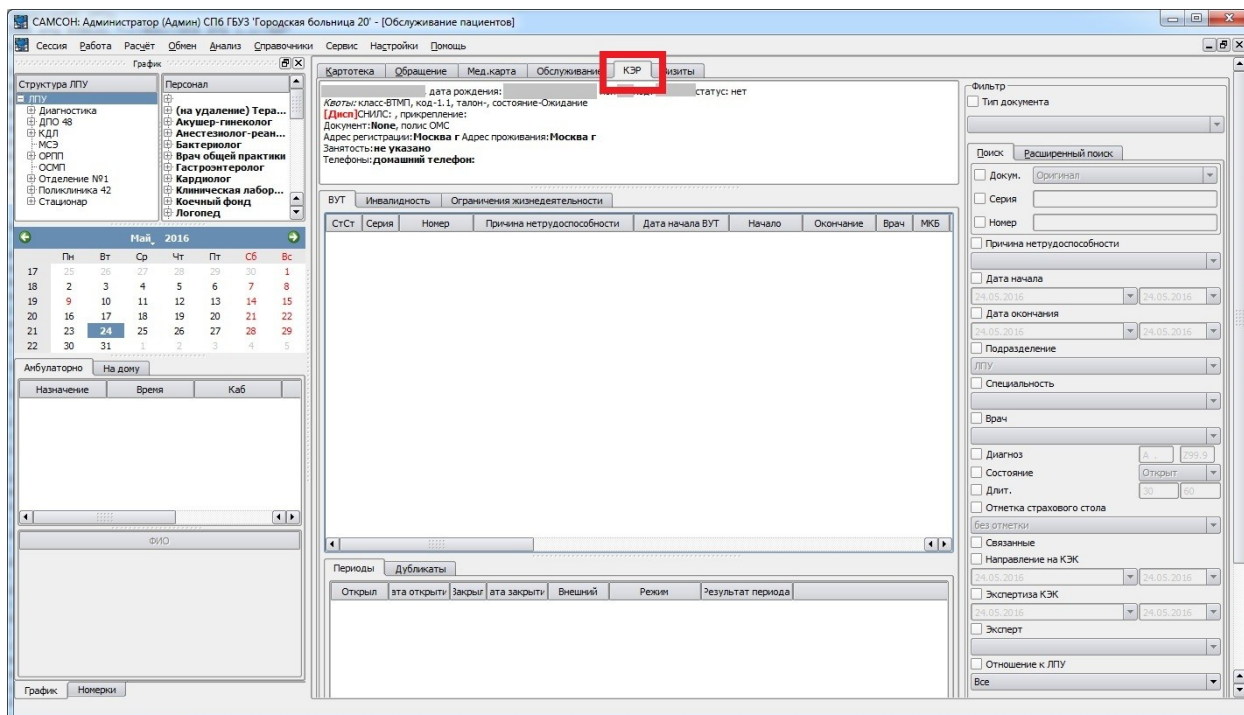


Рисунок 7 – Вкладка КЭР

Откроется окно **ВУТ** (рис. 8). В нем слева заполняем необходимые поля. Если место работы указано в регистрационной карточке пациента, то поле «**Место работы**» заполнится автоматически. В таблице указываем дату начала и окончания периода нетрудоспособности, результат по периоду, врача, закончившего период. Если необходимо указать председателя врачебной комиссии, то указываем врача в поле Эксперт.

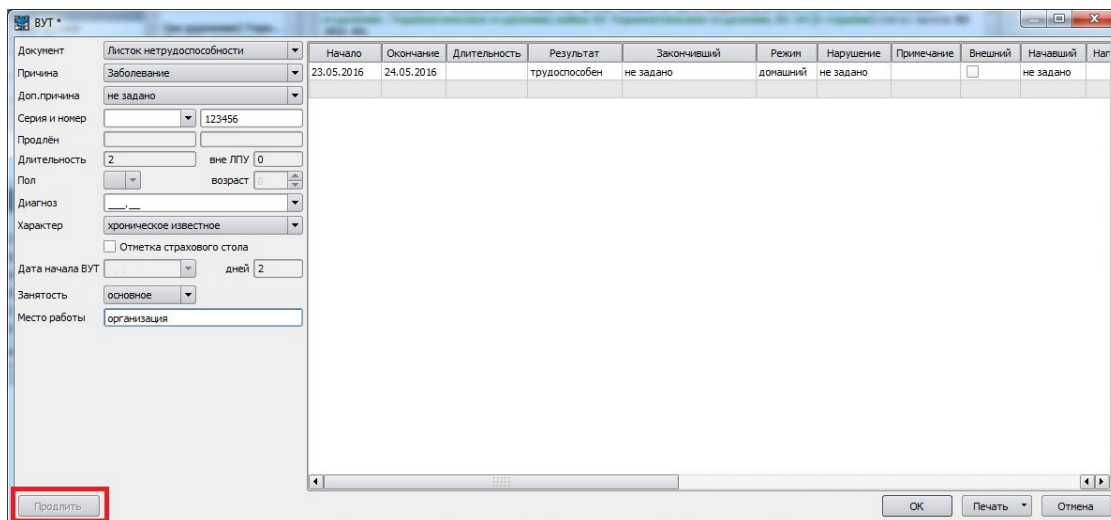


Рисунок 8 – Окно регистрации ВУТ

Работа

Обслуживание пациентов

КЭР

Если больничный лист необходимо продлить, то сделать это можно нажав на кнопку «**Продлить**». Обычно это требуется в случае, если у пациента вводится больше трех периодов нетрудоспособности.

Нажимаем кнопку «**Печать**», для того, чтобы напечатать листок нетрудоспособности. Для сохранения листка нажимаем кнопку ОК.

Удалить больничный лист можно на вкладке **ЛУД**, нажав правой кнопкой мыши на нужную запись (рис. 9).

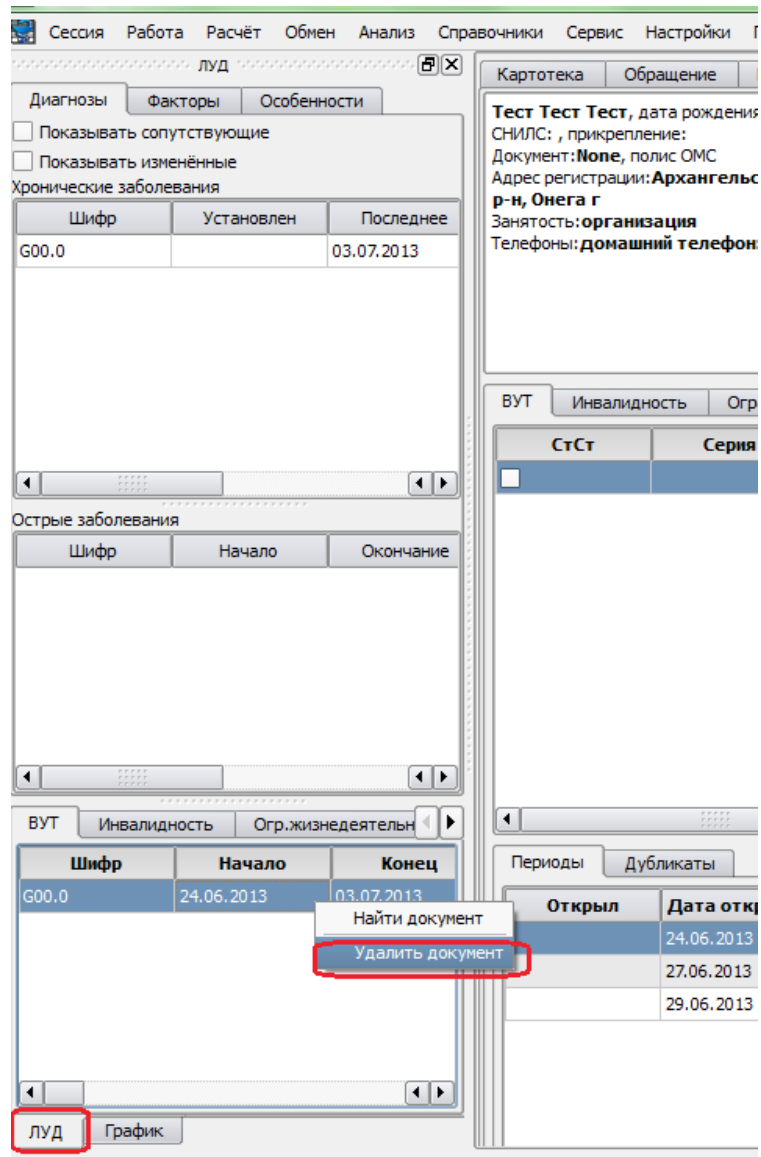


Рисунок 9 – Удаление листа нетрудоспособности на вкладке ЛУД



## Регистрация больничного листа

## Учет дубликатов

Находясь на вкладке КЭР, щелкаем правой кнопкой мыши по выбранной записи, выбираем «Учесть дубликат документа» (рис. 10).

## Работа

## Обслуживание пациентов

## КЭР

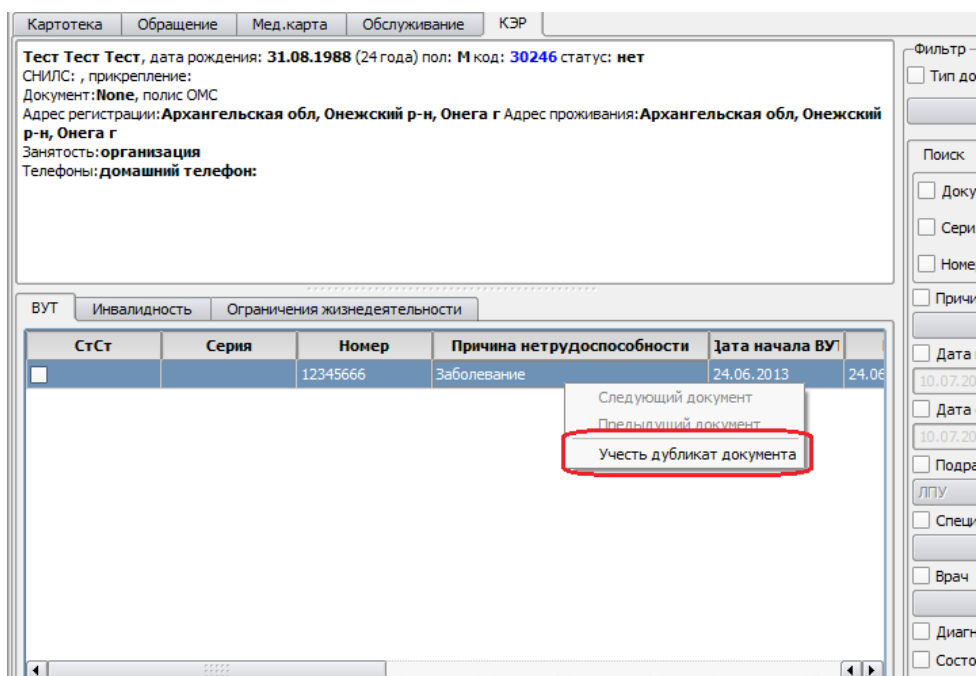


Рисунок 10 – Учет дубликатов больничных листов

В появившемся окне (рис. 11) заполняем необходимые поля и нажимаем кнопку ОК.

Выдача дубликата больничного листа может быть произведена в случае порчи или утери документа, поэтому в поле «Причина выдачи» необходимо указывать 2 - «Утеря» или 3 - «Порча».

Рисунок 11 – Создание дубликата документа временной нетрудоспособности

**Работа**      **Обслуживание пациентов**      **КЭР**

Посмотреть и удалить дубликаты документа можно на вкладке «Дубликаты» (рис. 12).

СтСт	Серия	Номер	Причина нетрудоспособности	Дата начала ВУТ	Начало	Окончание	Врач
<input type="checkbox"/>		111112	Заболевание	11.04.2016	19.04.2016	22.04.2016	

СтСт	Серия	Номер	Дата
<input type="checkbox"/>		111111-2	29.04.2016

Рисунок 12 – Дубликаты листа временной нетрудоспособности

Регистрация больничного листа

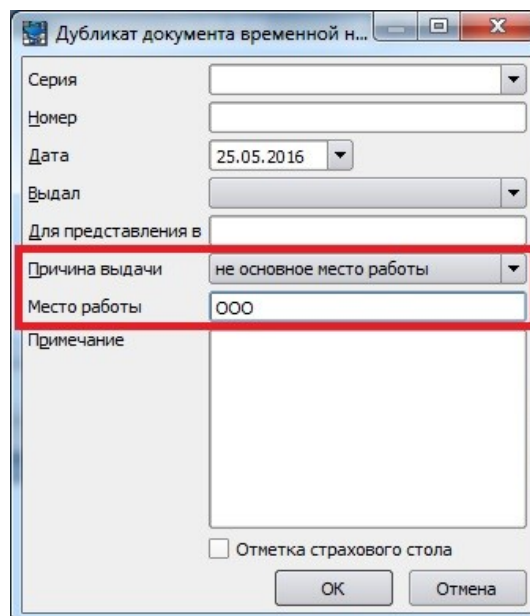
По совместительству

Если был зарегистрирован больничный лист для предоставления по основному месту работы, то для внесения листа по совместительству необходимо создать дубликат (см. раздел «Регистрация больничного листа - Учет дубликатов»), в котором нужно указать в строке «Причина выдачи» - 1 - «не основное место работы» и заполнить «Место работы» (рис. 13).

Работа

Обслуживание пациентов

КЭР



Дубликат документа временной н...

Серия

Номер

Дата 25.05.2016

Выдал

Для представления в

Причина выдачи не основное место работы

Место работы ООО

Примечание

Отметка страхового стола

ОК Отмена

Рисунок 13 – Создание больничного листа для предоставления по месту работы по совместительству

В случае, если в системе регистрируется только больничный лист для предоставления по неосновному месту работы, а по основному месту работы больничный лист не предоставляется, то необходимо при регистрации или редактировании листа нетрудоспособности в окне ВУТ в строке «Занятость» указать «совместитель» и заполнить строку «Место работы» (рис. 14).

Дубликат при этом надо создавать только при необходимости печати еще одного листа нетрудоспособности в другое место работы по совместительству (см. раздел «Печать больничного листа – По совместительству»).

Работа

Обслуживание пациентов

КЭР

Рисунок 14 – Создание больничного листа для предоставления по месту работы по совместительству

Печать больничного листа

По основному месту работы

Чтобы напечатать больничный лист, можно нажать F6 на клавиатуре или на кнопку «Печать» на вкладке **КЭР**, выделив нужную запись (рис. 15). В открывшемся диалоговом окне можно выбрать необходимые пункты для печати.

Работа

Обслуживание пациентов

КЭР

Рисунок 15 – Печать больничного листа для предоставления по основному месту работы

Если при создании больничного листа или при настройке поле «**Занятость**» не было заполнено, то на печать будет выведено место работы, указанное в регистрационной карточке пациента. Если в регистрационной карте место работы не заполнено, то надо заполнить «**Место работы**» в окне ВУТ (рис. 16).

Рисунок 16 – Указание основного места работы

Если были учтены дубликаты, но при этом надо распечатать основной лист, данные для которого были заполнены в окне ВУТ, то необходимо вернуться на вкладку «**Периоды**» (рис. 17).

**Работа**      **Обслуживание пациентов**      **КЭР**

СтСт	Серия	Номер	Причина нетрудоспособности	Дата начала ВУТ	Начало	Окончание	Врач	МКБ	Предь
		111112	Заболевание	11.04.2016	19.04.2016	22.04.2016		A01.0	111111

Открыл	Дата открыт	Закрыл	Дата закрыт	Внешний	Режим	Результат период
	19.04.2016		22.04.2016	<input type="checkbox"/>		нетрудоспособен

Рисунок 17 – Вкладка периоды

Печать больничного листа

По совместительству

Работа

Обслуживание пациентов

КЭР

Печать листа по совместительству происходит аналогично печати дубликата (см. раздел «**Печать больничного листа – Печать дубликата**»), в случаях, если:

- 1) Больничный лист предоставляется и по основному месту работы, и по месту работы по совместительству;
- 2) В окне ВУТ в строке «**Занятость**» указано «**совместитель**», но надо распечатать лист для предоставления по другому месту работы по совместительству.

В случае если необходимо распечатать лист по совместительству, данные для которого указаны в окне ВУТ, то процедура печати аналогична печати листа для предоставления по основному месту работы (см. раздел «**Печать больничного листа – По основному месту работы**»).

Печать больничного листа

Печать дубликата

Для печати дубликата листа нетрудоспособности, необходимо перейти на вкладку «**Дубликаты**», выбрать нужный лист и нажать кнопку печати (рис. 18).

Работа

Обслуживание пациентов

КЭР

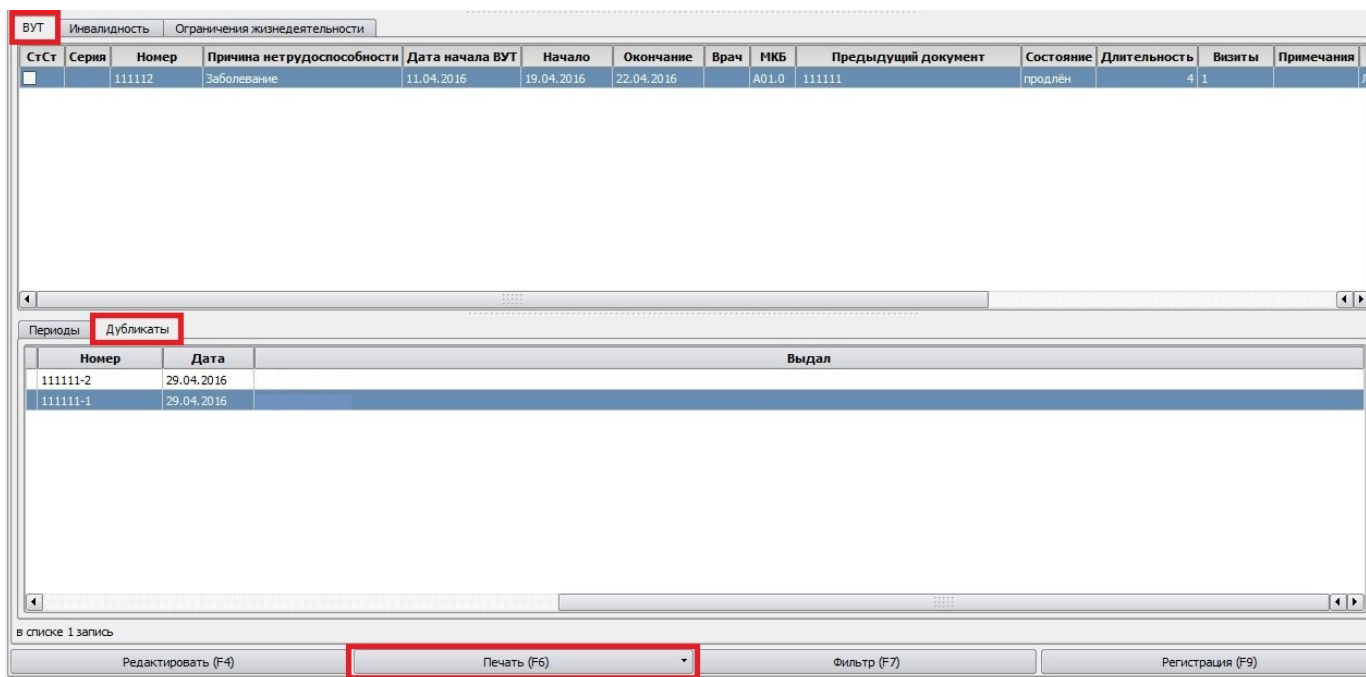


Рисунок 18 – Печать из вкладки дубликаты

Если необходимо распечатать дубликат листа нетрудоспособности для предоставления по не основному месту работы, в случае когда лист нетрудоспособности был утерян или испорчен, а в системе он уже зарегистрирован как дубликат документа по основному месту работы, то необходимо при печати документа в диалоговом окне «Печать дубликата» установить отметку «Дубликат листа по совместительству». (рис. 19). Этот дубликат в системе никак не учитывается.

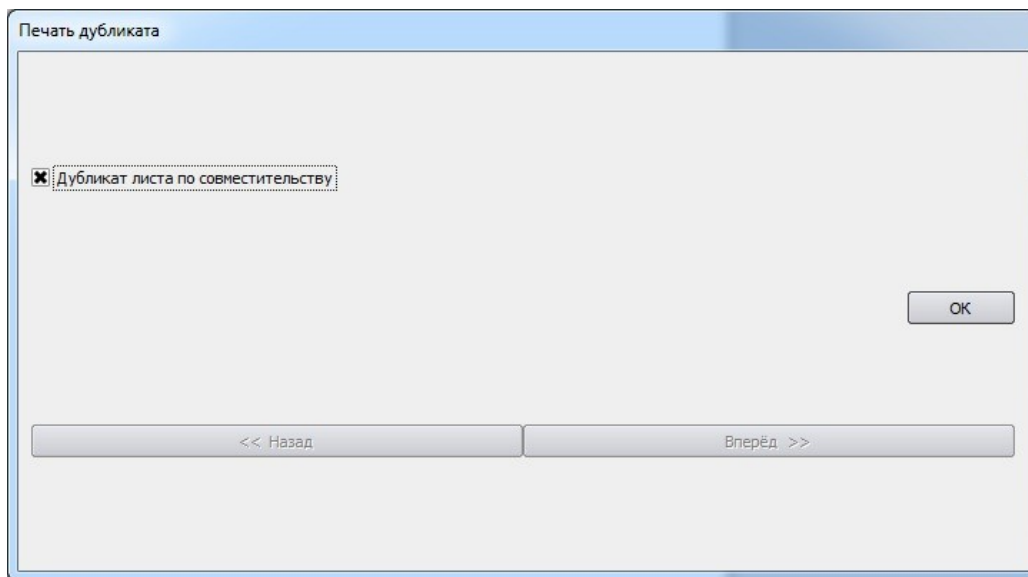


Рисунок 19 – Печать дубликата листа по совместительству, оформленного как дубликат

Печать больничного листа

Настройка шаблона печати

Для того, чтобы настроить печать листа нетрудоспособности, необходимо в меню «**Настройки**» выбрать пункт «**Шаблоны печати**» (рис. 20).

Настройки

Шаблоны печати

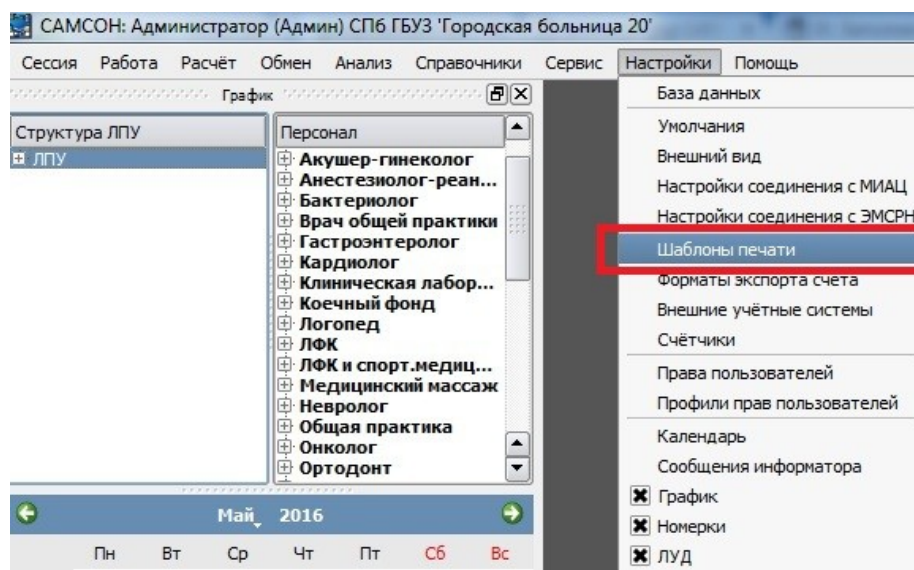


Рисунок 20 – Настройки шаблонов печати



Если необходимо добавить новый шаблон печати, то в открывшемся окне «Шаблоны печати» нажимаем кнопку «Вставка» или F9 на клавиатуре. Если шаблон печати уже создан и его нужно отредактировать, то необходимо выбрать нужный шаблон и нажать кнопку «Правка» или клавишу F4 на клавиатуре(рис. 21).

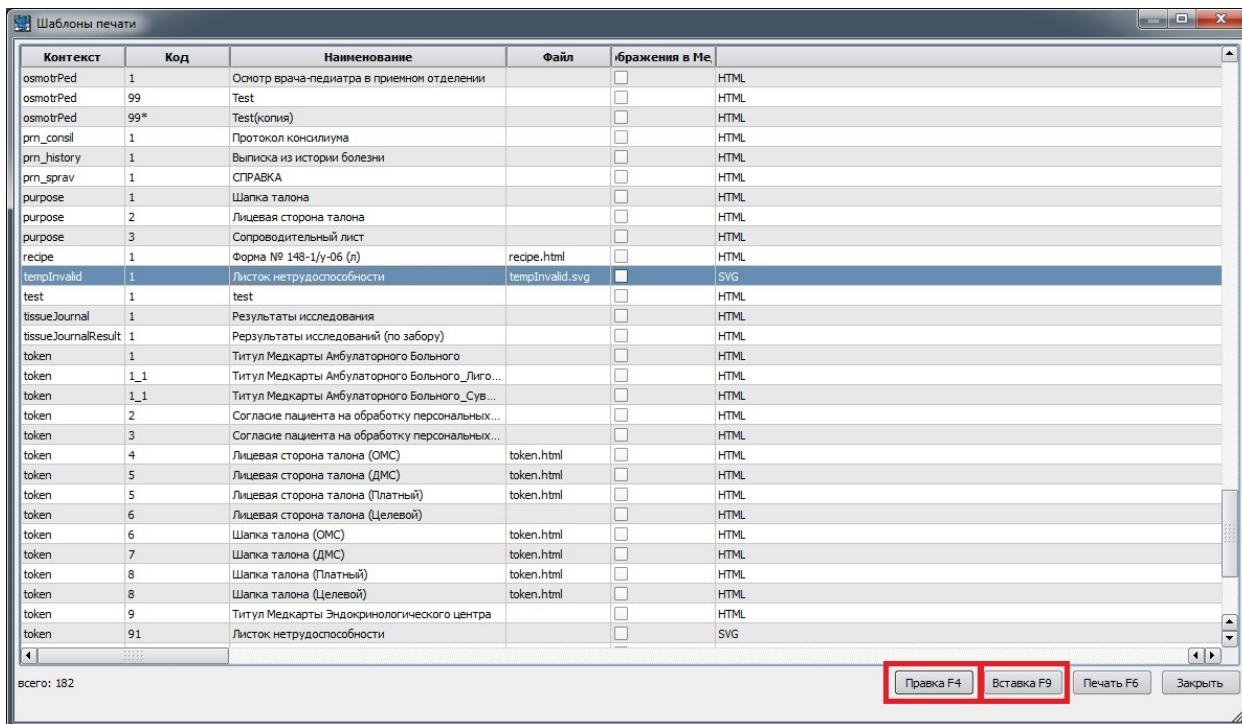


Рисунок 21 – Шаблоны печати

Настройки

Шаблоны печати

Листок нетрудоспособности

В настройках шаблона печати должны быть указаны (рис. 22):

**Блок 1** – «Контекст» - TempInvalid;

**Блок 2** – «Код» - код шаблона в произвольной форме;

**Блок 3** – «Наименование» - «Лист нетрудоспособности»;

**Блок 5** – «Тип» - SVG.

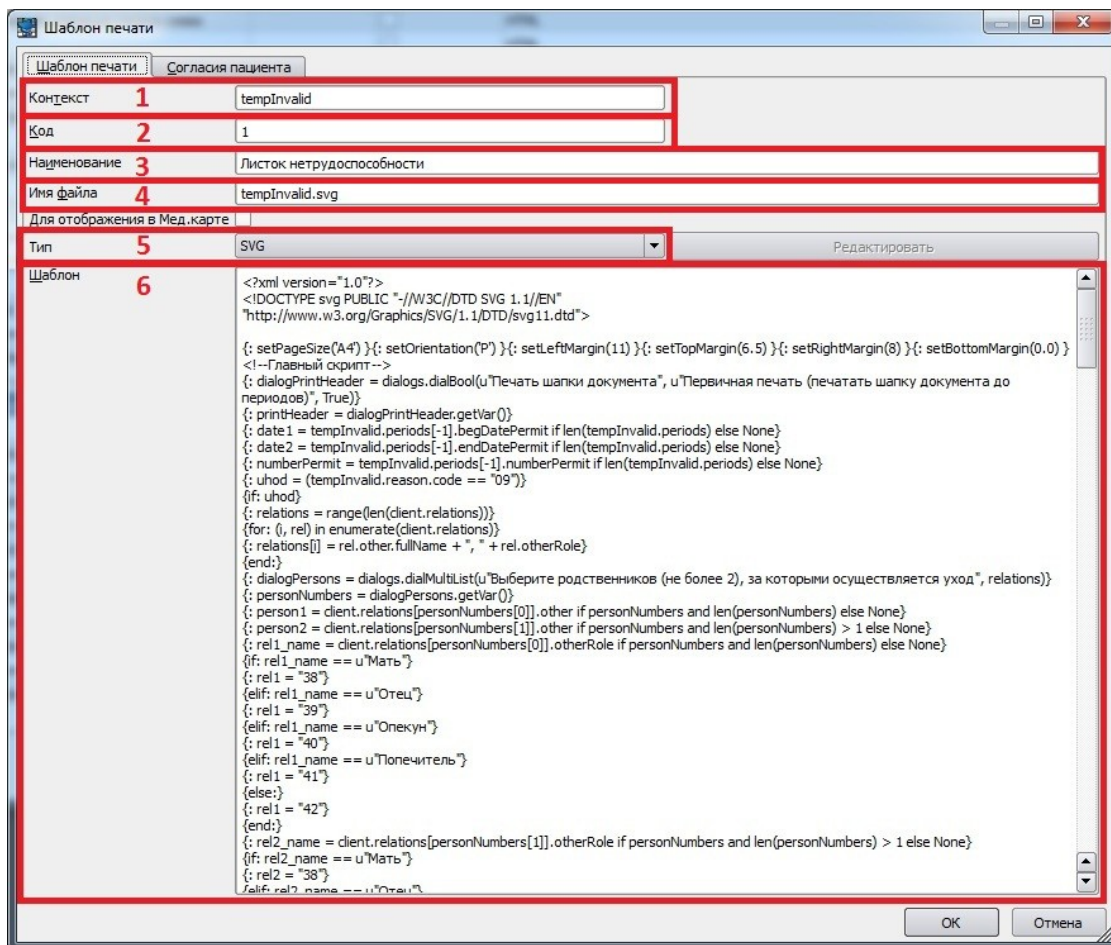


Рисунок 22 – Редактирование шаблона печати

В строках «Код» и «Наименование» указываются, соответственно, код и наименование, по которым в дальнейшем можно будет найти данный шаблон.

Если есть svg-файл с шаблоном печати, хранящийся в папке с шаблонами, то можно указать имя этого файла в строке «Имя файла» (**блок 4**). Уточнить директорию, в которой находятся шаблоны документов, можно открыв меню «Настройки» - «Умолчания» (рис. 23). В открывшемся окне необходимо открыть вкладку «Прочие настройки». Путь к шаблонам будет указан в строке «Директория с шаблонами документов» (рис. 24).

Если файл с шаблоном отсутствует, то в поле «Шаблон» (**блок 6**) необходимо вставить текст из раздела «Печать больничного листа – Листинг шаблона печати».

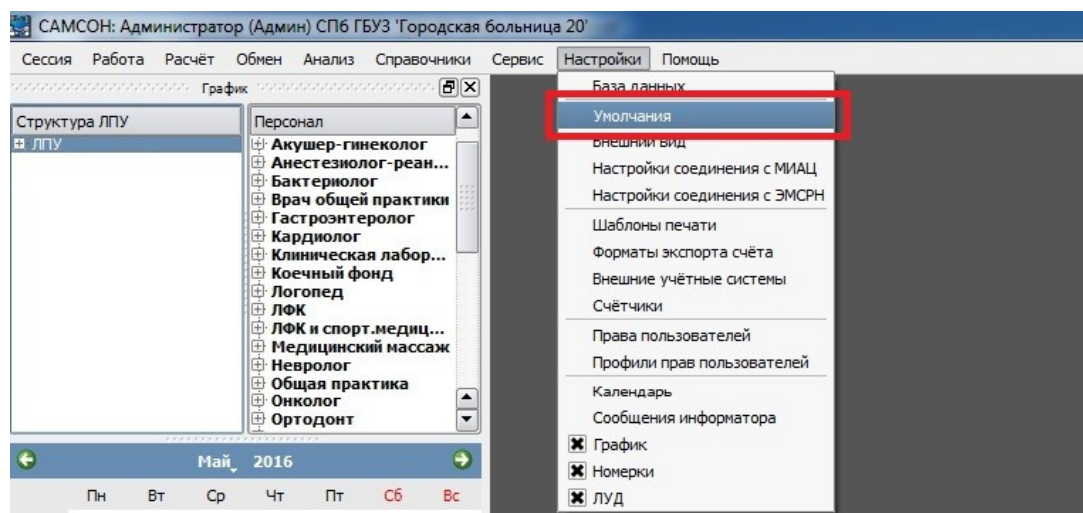


Рисунок 23 – Настройки умолчаний

Настройки

Умолчания

Прочие настройки

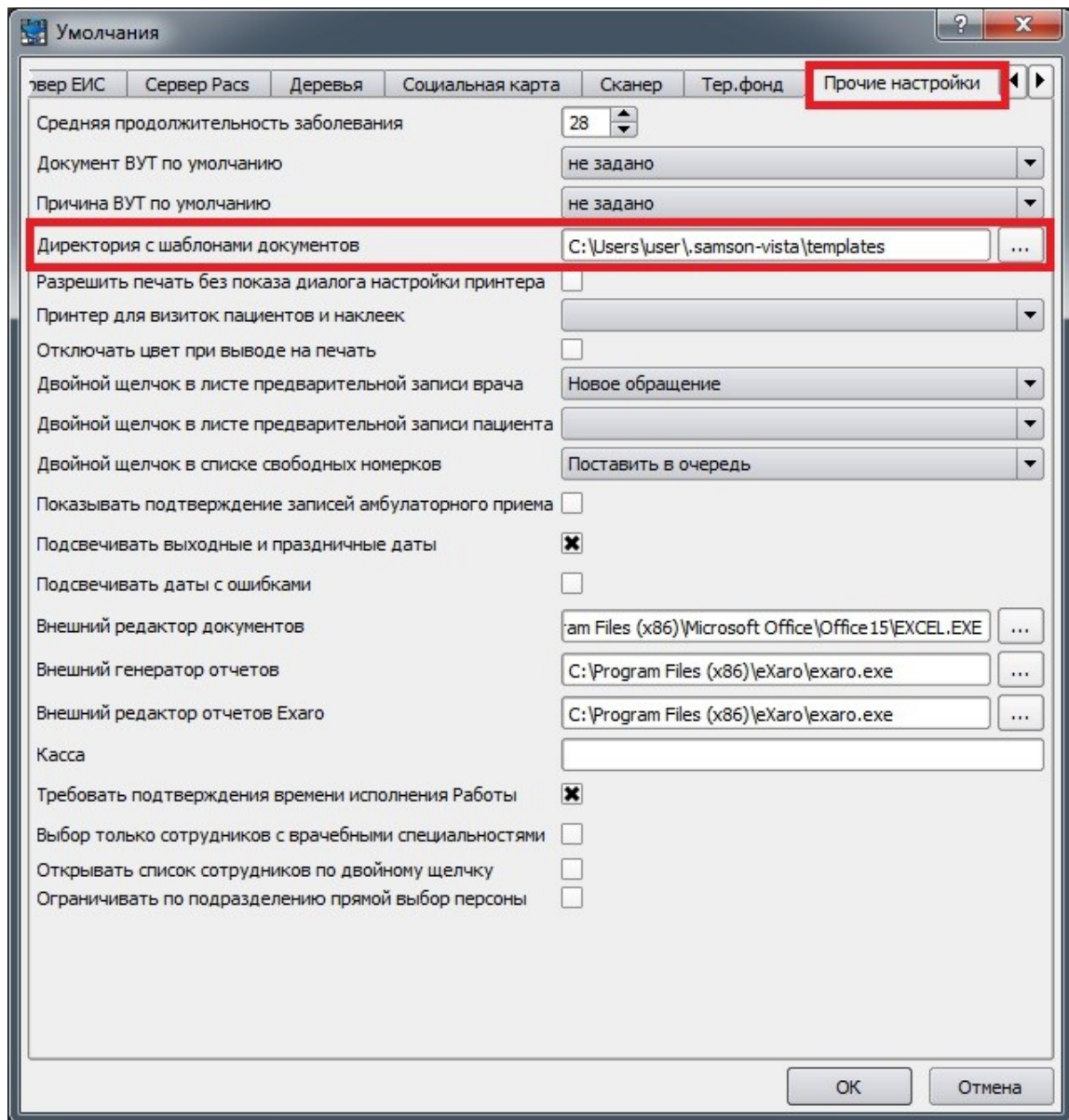


Рисунок 24 – Настройка директории с шаблонами документов

```

<?xml version="1.0"?>

<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"

"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">

{: setPageSize('A4') }{: setOrientation('P') }{: setLeftMargin(11) }{: setTopMargin(8) }{:
setRightMargin(10) }{: setBottomMargin(0.0) }

<!--Главный скрипт-->

<!--Если необходимо распечатать дубликат листа для предоставления по месту работы по
совместительству,

то в появившемся диалоговом окне можно выбрать печать дубликата дубликата -->

{: if: isDuplicate == 1 and tempInvalid.duplicateReason.code == "1"}

{: dialogDuplicate = dialogs.dialBool(u"Печатьдубликата",
u"Дубликатлистапосовместительству", False)}

{: Duplicate = (dialogDuplicate.getVar() == True)}

{elif: isDuplicate}

{:Duplicate = 1}

{else:}

{:Duplicate = 0}

{end:}

{: dialogPrintHeader = dialogs.dialBool(u"Печатьшапкидокумента", u"Первичнаяпечать
(печататьшапкудокументадопериодов)", True)}

{: printHeader = dialogPrintHeader.getVar()}

{: date1 = tempInvalid.periods[-1].begDatePermit if len(tempInvalid.periods) else None}

{: date2 = tempInvalid.periods[-1].endDatePermit if len(tempInvalid.periods) else None}

{: numberPermit = tempInvalid.periods[-1].numberPermit if len(tempInvalid.periods) else
None}

```

```

{: uhod = (tempInvalid.reason.code == "09")}

{if: uhod}

{: relations = range(len(client.relations))}

{for: (i, rel) in enumerate(client.relations)}

{: relations[i] = rel.other.fullName + ", " + rel.otherRole}

{end:}

{: dialogPersons = dialogs.dialMultiList(u"Выберите родственников (не более 2),
закоторыми осуществляется уход", relations)}

{: personNumbers = dialogPersons.getVar()}

{: person1 = client.relations[personNumbers[0]].other if personNumbers and
len(personNumbers) else None}

{: person2 = client.relations[personNumbers[1]].other if personNumbers and
len(personNumbers) > 1 else None}

{: rel1_name = client.relations[personNumbers[0]].otherRole if personNumbers and
len(personNumbers) else None}

{if: rel1_name == u"Мать"}

{: rel1 = "38"}

{elif: rel1_name == u"Отец"}

{: rel1 = "39"}

{elif: rel1_name == u"Опекун"}

{: rel1 = "40"}

{elif: rel1_name == u"Попечитель"}

{: rel1 = "41"}

{else:}

{: rel1 = "42"}

{end:}

{: rel2_name = client.relations[personNumbers[1]].otherRole if personNumbers and
len(personNumbers) > 1 else None}

```

```
{if: rel2_name == u"Мать"}
{: rel2 = "38"}
{elif: rel2_name == u"Отец"}
{: rel2 = "39"}
{elif: rel2_name == u"Опекун"}
{: rel2 = "40"}
{elif: rel2_name == u"Попечитель"}
{: rel2 = "41"}
{else:}
{: rel2 = "42"}
{end:}
{else:}
{: person1 = None}
{: person2 = None}
{: rel1 = None}
{: rel2 = None}
{end:}
{if: len(templInvalid.periods)}
{: periodstrs = []}
{for: (i, period) in enumerate(templInvalid.periods)}
{if: i < 3}
{: periodstrs = periodstrs + ["%d: %s - %s"%(i, period.begDate.date.toString("dd.MM.yyyy"),
period.endDate.date.toString("dd.MM.yyyy")) ,]}
{end:}
{end:}
{: dialogPeriods = dialogs.dialMultiList(u"Впечатывать периоды:", periodstrs)}
{: periodNumbers = dialogPeriods.getVar()}}
```

```

{else:}

{: periodNumbers = []}

{end:}

{if: len(periodNumbers)}

{: specialists = [(i, period.expert) for (i, period) in enumerate(tempInvalid.periods) if i in
periodNumbers and len(period.expert.shortName)]}

{: spec_strs = ["%s: %s"%(periodstrs[i], expert.shortName) for (i, expert) in specialists]}

{: dialogExperts = dialogs.dialMultiList(u"Впечатыватьэкспертов:", spec_strs)}

{: expertNumbers = dialogExperts.getVar()}

{: expertNumbers = [specialists[i][0] for i in expertNumbers]}

{else:}

{: expertNumbers = []}

{end:}

{: dialogOrg = dialogs.dialBool(u"Выводитьорганизациюилиподразделение?",
u"ВыводитьданныеобЛПУпобазовомуподразделению", False)}

{: printOrgStructure = (dialogOrg.getVar() == True); }

{: organisation = currentOrganisation.title if not printOrgStructure else
currentOrgStructure.name}

{: organisation = organisation.upper()}

<!--удаляем из названия организации символы ",.-: -->

{for: letter in ",.-:\\""}

{: organisation = organisation.replace(letter, "")}

{end:}

{: address = currentOrganisation.address if not printOrgStructure else
currentOrgStructure.address}

{: address = address.upper()}

<!--удаляем из адреса символы ",.-: -->

{for: letter in ",.-:\\""}

```



```

{: address = address.replace(letter, "")}

{end:}

{: address = address.replace("-", " ")}

{: address = address.replace(u"ПР ", "") }

{: address = address.replace(u"УЛ ", "") }

{: address = address.replace(u" Д ", " ")}

{: OGRN = currentOrganisation.OGRN if not isDuplicate else ""}

{: date = tempInvalid.date if Duplicate else currentDate} <!-- было isDuplicate -->

{: lastName = client.lastName.upper()}

{: firstName = client.firstName.upper()}

{: patrName = client.patrName.upper()}

{if: tempInvalid.placeWork}

{: placeWork = tempInvalid.duplicatePlaceWork if (isDuplicate and
tempInvalid.duplicateReason.code == "1") else tempInvalid.placeWork}

{else:}

{:placeWork = tempInvalid.client.work.shortName}

{end:}

{: placeWork = placeWork.upper()}

{: sovместит = tempInvalid.busyness == 1 and(isDuplicate and
tempInvalid.duplicateReason.code == "1")} <!--было tempInvalid.busyness == 2 or (...)-->

{: number = tempInvalid.number if sovместит else ""}

{def: shortn(person)}

{: result = person.lastName + " " + (person.firstName[0] if len(person.firstName) else "") + " " +
(person.patrName[0] if len(person.patrName) else "")}

{: result = result.upper()}

{: return result}

{end:}

```

```

<!--приступитьк работе: -->

{if: len(templInvalid.periods) and templInvalid.periods[-1].result.code == "1"}

{: pristupDate = templInvalid.periods[-1].endDate.date.addDays(1)}

{else:}

{: pristupDate = None}

{end:}

{if: templInvalid.prev}

{: prev_number = templInvalid.prev.number}

{else:}

{: prev_number = " "}

{end:}

<!--Печататькорешок -->

{: dialogKoreshok = dialogs.dialBool(u"Печатькорешка",
u"Печататькорешоклистанетрудоспособности", True)}

{: printKoreshok = dialogKoreshok.getVar()}

<!--Расстояниемеждубуквами:-->

{: L=11.3385}

{def: writeText(x, y, s)}

{for: i,c in enumerate(s)}

<text x="{x+i*L}" y="{y}" textLength="{L}">{c}</text>

{end:}

{end:}

{def: writeTextMM(x, y, s)}

{: writeText(x*72/25.4, y*72/25.4, s)}

{end:}

<!--Конец главного скрипта-->

<svg xmlns="http://www.w3.org/2000/svg"

```

```

xmlns:xlink="http://www.w3.org/1999/xlink"

width="210mm"

height="297mm"

viewBox="0 0 595 842"

version="1.2"

baseProfile="tiny">

<g transform="translate(45,50), rotate(-0.3)" style="font-family:'Courier'; font-size:10">

<image x="-350" y="0" width="100%" height="100%" xlink:href="vut_List.jpg"/>

</g>

<g transform="translate(5,11)" style="font-family:'Courier New'; font-size:7;font-weight:bold">

<!-- было: transform="translate(50,50) " style="font-family:'Courier'; font-size:10" -->

{if: printHeader}

<!--Галочки -->

{: writeTextMM(67-1, 10-1, "v" if (not Duplicate and not tempInvalid.prev) else "")} <!-- было
not isDuplicate and -->

{: writeTextMM(67-1, 15-1, "v" if (Duplicate) else "")}

<!-- пред. номер -->

{: writeTextMM(101, 11.375, prev_number[:14]) }

<!--Название организации-->

{: writeTextMM(53, 18.125, organisation[:38])}

<!--Адрес организации -->

{: writeTextMM(53, 24.875, address[:38]) }

<!--Дата выдачи -->

{: writeTextMM(69, 31.625, "%.2d"%date.date.day()) }

{: writeTextMM(79, 31.625, "%.2d"%date.date.month()) }

{: writeTextMM(89, 31.625, "%.4d"%date.date.year()) }

{: writeTextMM(109, 31.625, OGRN[:38]) }

```

```

<!--ФИОпациента -->

{: writeTextMM(53, 39, lastName) }

{: writeTextMM(53, 45, firstName) }

{: writeTextMM(53, 51, patrName) }

<!--Датарожденияпациента -->

{: writeTextMM(9, 59, "%.2d"%client.birthDate.date.day()) }

{: writeTextMM(19, 59, "%.2d"%client.birthDate.date.month()) }

{: writeTextMM(29, 59, "%.4d"%client.birthDate.date.year()) }

{: writeTextMM(57.25, 59, "v" if client.sexCode == 1 else "") }

{: writeTextMM(65.25, 59, "v" if client.sexCode == 2 else "") }

{: writeTextMM(129, 59, templInvalid.reason.code[:2] if templInvalid.reason else "") }

{: writeTextMM(141, 59, templInvalid.extraReason.code[:3] if templInvalid.extraReason else "") }

<!--Местоработыпациента -->

{: writeTextMM(9, 66.375, placeWork[:29]) }

{: writeTextMM(21.5, 73.125, "v" if templInvalid.busyness == 1 and not sovместit else "") }

{: writeTextMM(53.5, 73.125, "v" if sovместit else "") }

{: writeTextMM(77.5, 73.125, number[:12]) }

{: writeTextMM(153.5, 73.125, ("v" if templInvalid.busyness == 3 and not sovместit else "")) }

{if: str(date1) != ""}

<!--Дата и номер путевки -->

{: writeTextMM(13, 80.5, "%.2d"%date1.date.day() if date1 else "") }

{: writeTextMM(23, 80.5, "%.2d"%date1.date.month() if date1 else "") }

{: writeTextMM(33, 80.5, "%.4d"%date1.date.year() if date1 else "") }

{: writeTextMM(57, 80.5, "%.2d"%date2.date.day() if date2 else "") }

{: writeTextMM(67, 80.5, "%.2d"%date2.date.month() if date2 else "") }

{: writeTextMM(77, 80.5, "%.4d"%date2.date.year() if date2 else "") }

```

```

{: writeTextMM(101, 80.5, numberPermit[:6] if numberPermit else "") }

{end:}

<!--Родственники -->

{: writeTextMM(13, 87.25, "%.2d"%person1.ageTuple[3] if person1 else "")}

{: writeTextMM(25, 87.25, "%.2d"%(person1.ageTuple[2]%12) if person1 else "")}

{: writeTextMM(37, 87.25, rel1[:2] if person1 else "")}

{: writeTextMM(49, 87.25, person1.fullName[:39].upper() if person1 else "") }

{: writeTextMM(13, 92.25, "%.2d"%person2.ageTuple[3] if person2 else "")}

{: writeTextMM(25, 92.25, "%.2d"%(person2.ageTuple[2]%12) if person2 else "")}

{: writeTextMM(37, 92.25, rel2[:2] if person2 else "")}

{: writeTextMM(49, 92.25, person2.fullName[:39].upper() if person2 else "") }

{end:}

<!--Период 1 -->

{if: periodNumbers and 0 in periodNumbers}

{: period = templInvalid.periods[0]}

{: writeTextMM( 7, 142.5, "%.2d"%period.begDate.date.day()) }

{: writeTextMM( 17, 142.5, "%.2d"%period.begDate.date.month()) }

{: writeTextMM( 27, 142.5, "%.4d"%period.begDate.date.year()) }

{: writeTextMM( 43, 142.5, "%.2d"%period.endDate.date.day()) }

{: writeTextMM( 53, 142.5, "%.2d"%period.endDate.date.month()) }

{: writeTextMM( 63, 142.5, "%.4d"%period.endDate.date.year()) }

{: writeTextMM( 79, 142.5, period.endPerson.speciality.name[:9].upper()) }

{: writeTextMM(115, 142.5, shortn(period.endPerson)[:14]) }

{if: 0 in expertNumbers}

{: writeTextMM( 79, 147.5, period.expert.post.name[:9].upper()) }

{: writeTextMM(115, 147.5, shortn(period.expert)[:14]) }

```

```

{end:}

{end:}

<!--Период 2 -->

{if: periodNumbers and 1 in periodNumbers}

{: period = templInvalid.periods[1]}

{: writeTextMM( 7, 152.5, "%.2d"%period.begDate.date.day() ) }

{: writeTextMM( 17, 152.5, "%.2d"%period.begDate.date.month() ) }

{: writeTextMM( 27, 152.5, "%.4d"%period.begDate.date.year() ) }

{: writeTextMM( 43, 152.5, "%.2d"%period.endDate.date.day() ) }

{: writeTextMM( 53, 152.5, "%.2d"%period.endDate.date.month() ) }

{: writeTextMM( 63, 152.5, "%.4d"%period.endDate.date.year() ) }

{: writeTextMM( 79, 152.5, period.endPerson.speciality.name[:9].upper() ) }

{: writeTextMM(115, 152.5, shortn(period.endPerson)[:14]) }

{if: 1 in expertNumbers}

{: writeTextMM( 79, 157.5, period.expert.post.name[:9].upper() ) }

{: writeTextMM(115, 157.5, shortn(period.expert)[:14]) }

{end:}

{end:}

<!--Период 3 -->

{if: periodNumbers and 2 in periodNumbers}

{: period = templInvalid.periods[2]}

{: writeTextMM( 7, 162.5, "%.2d"%period.begDate.date.day() ) }

{: writeTextMM( 17, 162.5, "%.2d"%period.begDate.date.month() ) }

{: writeTextMM( 27, 162.5, "%.4d"%period.begDate.date.year() ) }

{: writeTextMM( 43, 162.5, "%.2d"%period.endDate.date.day() ) }

{: writeTextMM( 53, 162.5, "%.2d"%period.endDate.date.month() ) }

```

```

{: writeTextMM( 63, 162.5, "%.4d"%period.endDate.date.year()) }

{: writeTextMM( 79, 162.5, period.endPerson.speciality.name[:9].upper()) }

{: writeTextMM(115, 162.5, shortn(period.endPerson)[:14]) }

{if: 2 in expertNumbers}

{: writeTextMM( 79, 167.5, period.expert.post.name[:9].upper()) }

{: writeTextMM(115, 167.5, shortn(period.expert)[:14]) }

{end:}

{end:}

<!--Приступить к работе с -->

{if: pristupDate}

{: writeTextMM(49, 174, "%.2d"%pristupDate.day()) }

{: writeTextMM(59, 174, "%.2d"%pristupDate.month()) }

{: writeTextMM(69, 174, "%.4d"%pristupDate.year()) }

{end:}

<!--Корешок -->

{if: printKoreshok}

<!--Галочкиипредыдущийномер -->

{: writeTextMM(66-1, 248.5-1, " " if (not isDuplicate and not tempInvalid.prev) else "")}

{: writeTextMM(66-1, 248.5-1, " " if isDuplicate else "")}

{: writeTextMM(103, 255, prev_number[:14]) }

<!--ФИО -->

{: writeTextMM(13, 258, lastName) }

{: writeTextMM(13, 264, firstName) }

{: writeTextMM(13, 270, patrName) }

<!--Местоработы -->

{: writeTextMM(13, 277, placeWork[:29]) }

```

```
{: writeTextMM(21.5, 283.5, ("v" if templInvalid.busyness == 1 and not sovместit else "")) }  
{: writeTextMM(53.5, 283.5, ("v" if sovместit else "")) }  
{: writeTextMM(76, 284.5, number[:12]) }  
<!-- ФИОВрача -->  
{: writeTextMM(141, 264.5, shorten(templInvalid.periods[-1].endPerson) if  
len(templInvalid.periods) else "" ) }  
<!-- № истории болезни и дата выдачи -->  
{: writeTextMM(164.5, 271.5, "%d"%client.id) }  
{: writeTextMM(164.5, 277.5, "%.2d"%date.date.day()) }  
{: writeTextMM(174.5, 277.5, "%.2d"%date.date.month()) }  
{: writeTextMM(184.5, 277.5, "%.4d"%date.date.year()) }  
{end:}  
</g>  
</svg>
```