

Автоматизация бизнес-процессов

Заключительный этап

Инженерный тур

Краткое описание задачи

Задача командного тура финала профиля «Автоматизация бизнес-процессов» Олимпиады Национальной технологической инициативы состоит в разработке приложения по автоматизации управления сети отелей и формирования предложений туристам и турагентствам в зависимости от их статуса и выбранного туристического направления.

За два дня участники должны разработать бизнес-приложение, полностью отвечающее требованиям технического задания.

Легенда

Что может быть прекраснее путешествия? Красота природы, архитектура, встречи с новыми людьми несут незабываемые впечатления, дают заряд бодрости и вдохновения.

Два друга детства, Иван Виноградов и Кирилл Самоцветов, исколесив всю нашу страну, решили дарить радость от путешествий другим, сделав комфорт неотъемлемой частью путешествий.

На накопленные сбережения они открыли несколько небольших, но очень комфортных отелей для любителей путешественников. Со временем сеть разрослась. Появилось много идей по развитию бизнеса и туризма в целом, т. к. друзьями движет не только желание заработать, но и любовь к путешествиям, к нашей необъятной стране, где есть, что посмотреть путешественникам. Таким образом, возникла потребность в создании эффективной системы управления отелями с правильно построенными бизнес-процессами, что позволит развивать свой бизнес и туризм по России в целом.

Друзья начали искать исполнителей для реализации данной идеи. Их знакомая, владелица туристического агентства, Алла Волкова посоветовала своего одноклассника Гришу, который в свое время ей очень помог, разработав приложение для автоматизации ведения бизнеса туристического агентства. Но вы же знаете, что разработчик из Гриши «не очень», и он опять обращается к вам за помощью. . .

Техническое задание

- Участникам финального тура предлагается создать приложение для автоматизации ведения бизнеса.
- Задача разделена на 7 сессий, каждая сессия — это решение отдельной подзадачи.
- Сессии открываются последовательно.
- Во всех сессиях подразумевается, что информационная система должна работать без сбоев. Выдавать пользователям информацию об ошибках на понят-

ном пользователю языке. Приложение должно запускаться простым, понятным пользователю способом.

- Разрабатываемое решение должно содержать тестовые данные и сопроводительную записку, объясняющую логику работы пользователя.

Правила отправки решения

Готовое решение должно быть скомпилировано и иметь запускаемый файл с именем `sessionX.exe`, где `X` — номер сессии. Для разработчиков на платформе «1С: Предприятие 8» результат работы должен быть представлен выгрузкой информационной базы с именем `sessionX.dt`.

Создание приложения для автоматизации учета отдельного бизнеса

Создание приложения и его наполнение справочной информацией, которая включает в себя:

- структуру хранения информации об отелях;
- структуру номерного фонда каждого отеля;
- структуру хранения и отображения информации о состоянии номеров (занят, свободен (чистый/грязный) и др.).

Длительность сессии: 2 академических часа.

Техническое задание

1. Разработать структуру хранения информации об отелях в виде справочника «Отели».
 - 1.1. «Название отеля» — строка до 150 символов, хранит наименование отеля.
 - 1.2. «Регион» — ссылка на элемент другой таблицы (справочника/классификатора), где хранится список всех регионов.
2. Разработать структуру номерного фонда каждого отеля.
 - 2.1. Разработать структуру хранения информации о номерном фонде отеля (структура хранения подчинена справочнику «Отели»).
 - 2.2. Основные характеристики номера.

Номер: Категория: Количество мест: Статус:

«Номер» — тип данных «строка» до 10 символов. «Категория» — список значений, редактируемый пользователем. В отелях есть номера следующих категорий:

- Стандарт (первая категория).
- Люкс.
- Апартамент.

«Количество мест» — тип «число» в номерах:

- Категория «Стандарт» и «Люкс» — до 2-х человек.
- Категория «апартамент» — до 4-х мест.

«Статус» — список значений, не редактируемый пользователем. Статусы номеров будут следующие:

- Занят.
- Занят (грязный).
- Свободен (чистый).
- Свободен (грязный).

При реализации механизма статусов занятости номеров необходимо учесть следующие моменты:

- Занятость учитывается только по датам — продаем количество ночей. Т. е. если гость приехал 13.03, а выезд у него 15.03, то считаем, что номер занят 13.03, 14.03 — продано 2 ночи. 15.03 — номер освобождается, можно в него заселить другого гостя после выездной уборки.
- Если номер занят, необходимо отображать соответствующие даты в карточке номера.

3. Создать отчет о загрузке номерного фонда, отвечающий на вопрос: «Что есть в наличии (какие номера и сколько) на определенную дату?».

3.1. Отчет необходимо составить в разрезе категорий.

3.2. Отчет открывается из карточки отеля и отображает загрузку номерного фонда **только** этого отеля.

Тестирование

1. Заполнить базу данными. Минимальный набор данных определен ниже.

1.1. 2 отеля в различных регионах.

1.2. В каждом отеле заполнить структуру номерного фонда — не менее 10 номеров:

- 1–2 номера категории «апартамент»;
- 2–3 номера категории «люкс»;
- 5–7 номеров категории «стандарт».

1.3. Проставить статусы занятости номеров. У 2–3 номеров (по каждому отелю) должен быть статус «Занят» и определены даты занятости.

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

Штраф за отсутствие пояснительной записки — 1 балл.

Критерии	Базовый балл	Прибавка	Макс. балл
Созданная база данных (информационная база)	10		10
Созданная таблица (справочник) «Отели»	5		5

Критерии	Базовый балл	Прибавка	Макс. балл
Реквизит таблицы (справочника) «Название отеля» — строка до 150 символов, хранит наименование отеля	2		2
Реквизит таблицы (справочника) «Регион»	2	Если реализована ссылка на справочник «Регионы» +2	4
Созданная структура хранения информации о номерном фонде отеля	10	Если структура подчинена таблице (справочнику «Отели») +5	15
Реквизит (характеристика) «Номер» — тип данных «строка» до 10 символов	2		2
Реквизит (характеристика) «Категория» — список значений, редактируемый пользователем	2	Заполненные категории: <ul style="list-style-type: none"> • Стандарт (первая категория). • Люкс. • Апартамент. +2	4
Реквизит (характеристика) «Статус» — список значений, редактируемый пользователем	2	Заполненные статусы: <ul style="list-style-type: none"> • Занят. • Занят (грязный). • Свободен (чистый). • Свободен (грязный). +2	4
Реквизит (характеристика) «Дата заезда» — тип «Дата»	2		2
Реквизит (характеристика) «Дата выезда» — тип «Дата»	2		2
Отображение дат заезда и выезда в зависимости от статуса		+3 есть попытка реализации отображения дат в зависимости от статуса (поля для ввода дат отображаются только при статусах «Занят» или «Занят (грязный)») +10 алгоритм отображения дат работает корректно	10
Контроль ввода дат		+3 попытка реализации механизма проверки, чтобы дата заезда не была больше даты выезда +10 алгоритм проверки, чтобы дата заезда не была больше даты выезда, работает корректно	10

Критерии	Базовый балл	Прибавка	Макс. балл
Созданный отчет о загрузке номерного фонда	5	+2 Данные в отчете формируются на определенную дату, но есть расхождения с реальными данными в базе +7 Данные в отчете формируются на определенную дату и корректны +3 Отчет, открываемый из карточки отеля, отображает данные по конкретному отелю	15
Тестирование. Заполнение данными. Ввод информации о двух отелях	1		1
Тестирование. Заполнение данными. В каждом отеле заполнить структуру номерного фонда — не менее 10 номеров: <ul style="list-style-type: none"> • 1–2 номера категории «апартамент» • 2–3 номера категории «люкс» • 5–7 номеров категории «стандарт» 	5		5
Тестирование. Заполнение данными. Проставить статусы занятости номеров. У 2–3 номеров (по каждому отелю) должен быть статус «Занят» и определены даты занятости.	4		4
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		+5	5

Основные этапы решения

Этапы решения приведены на платформе «1С:Предприятие 8».

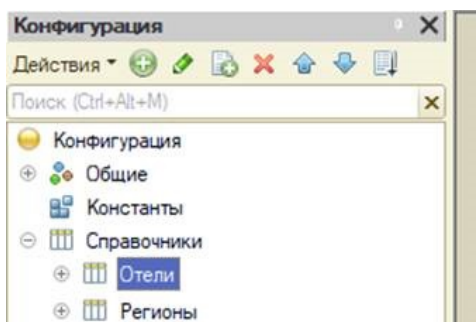
Если данные материалы используются для подготовки к Олимпиаде следующего года, то рекомендуем в качестве программного обеспечения использовать учебную версию платформы. Скачать ее можно совершенно бесплатно здесь: <https://uc1.1c.ru/uchebnaya-versiya-1s/>.

Устанавливается она совершенно стандартно с помощью мастера установки (как и все остальные программы).

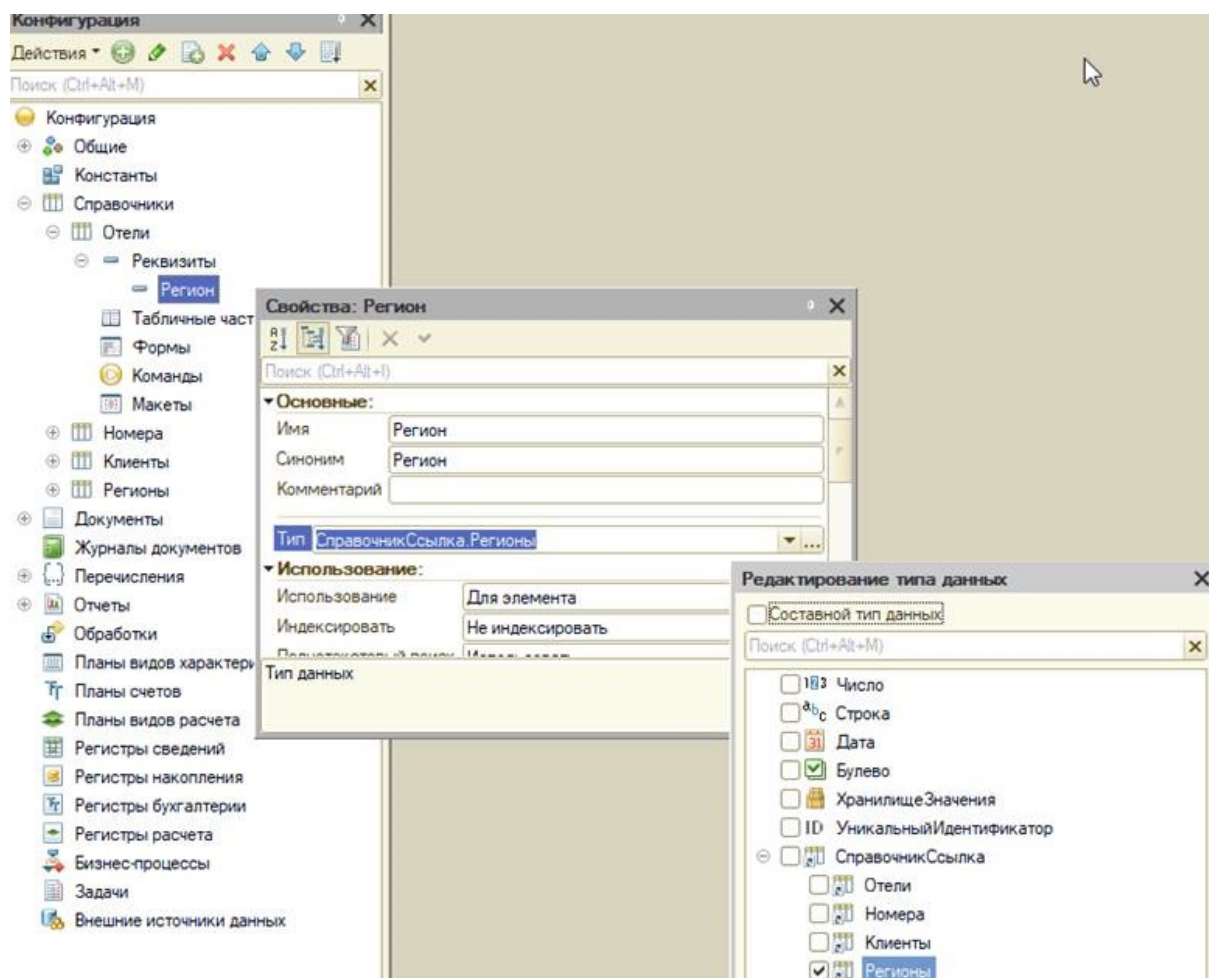
После запуска платформы (после установки создается соответствующий ярлычок) необходимо создать информационную базу — каркас для создания будущего приложения. Как добавить информационную базу описано в разборе задач второго этапа Олимпиады.

Информационную базу нужно открыть в режиме «Конфигуратор» для разработки нашего приложения. Далее необходимо действовать строго по этапам технического задания (ТЗ):

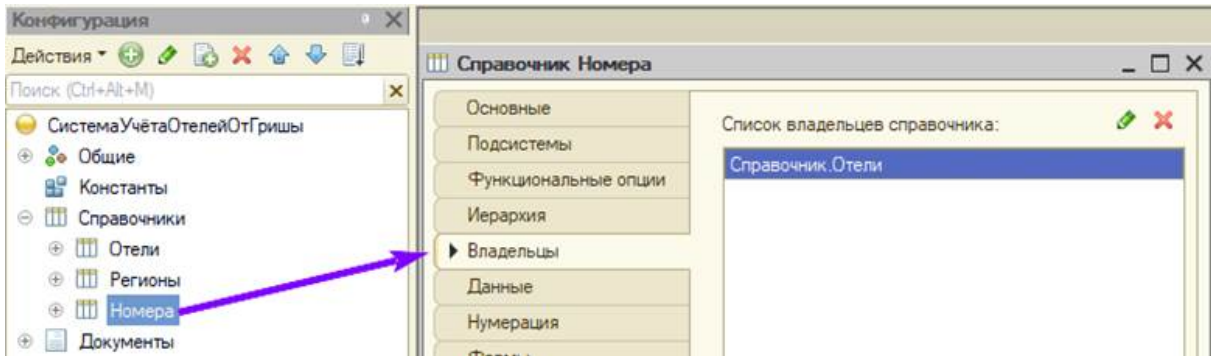
1. Разработать структуру хранения информации об отелях в виде справочника «Отели».
 - 1.1. «Название отеля» — строка до 150 символов, хранит наименование отеля.
 - 1.2. «Регион» — ссылка на элемент другой таблицы (справочника/классификатора), где хранится список всех регионов.
- Т. е. нам надо создать два справочника «Отели» и «Регионы» и указать связь между ними через поле «Регион» в справочнике «Отели».



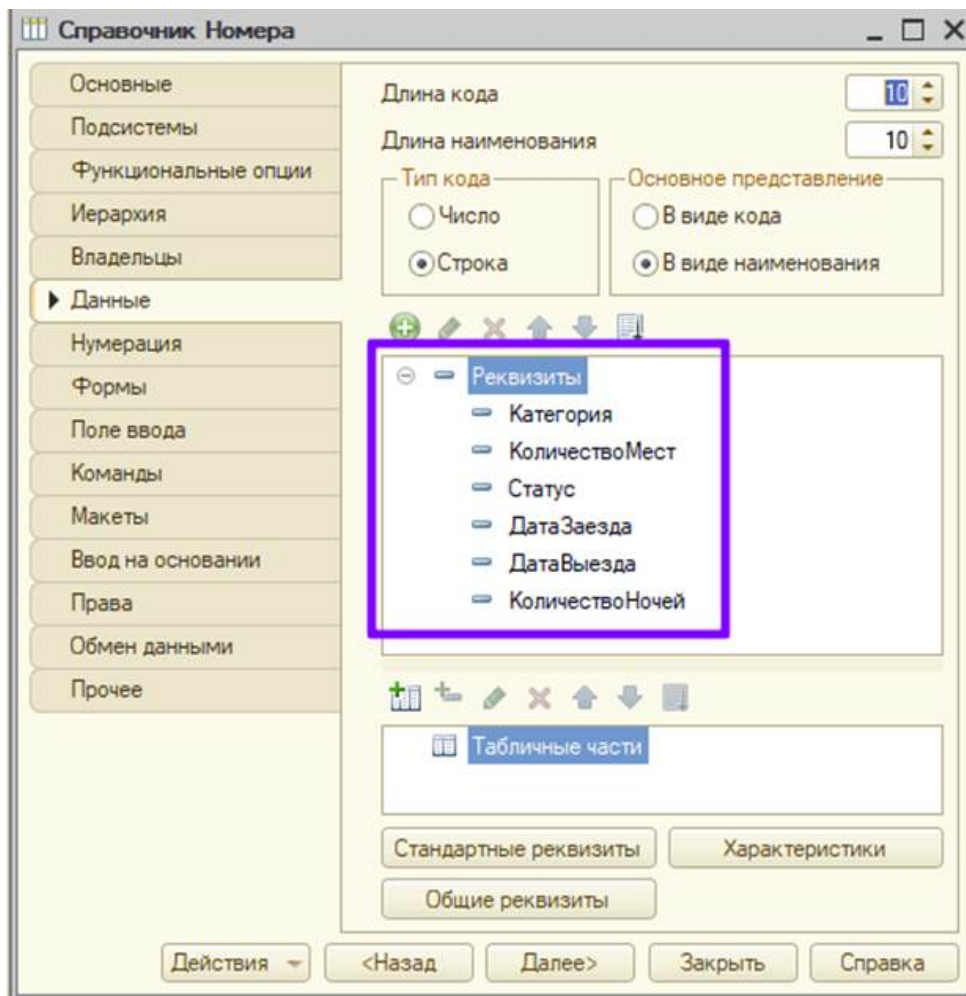
Справочник «Регионы» будет только со стандартными реквизитами. В справочнике «Отели» будет поле «Регион» с типом данных — ссылка на справочник «Регионы».



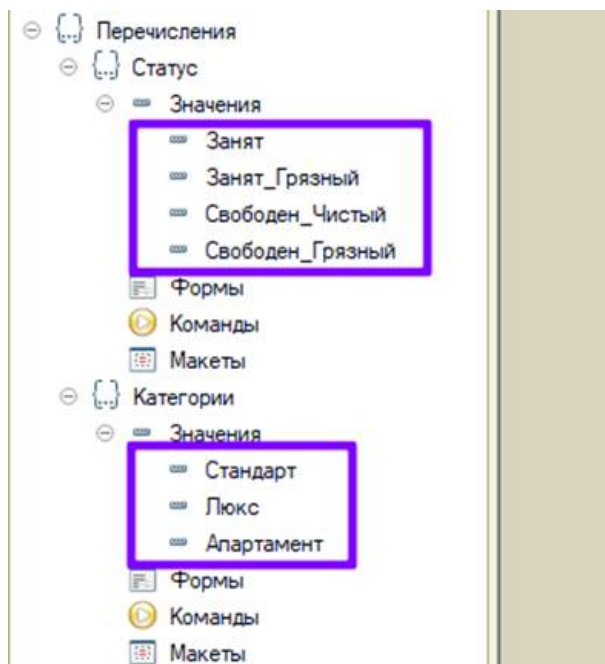
- Следующим этапом необходимо разработать структуру хранения номеров в «Отеле». Структуру хранения номеров можно реализовать двумя способами — либо в виде справочника, либо в виде регистра сведений. Здесь важно указать связь между справочником «Отели» и данной структурой: номера должны «принадлежать» конкретному отелю. Это можно сделать через свойство «Владелец».



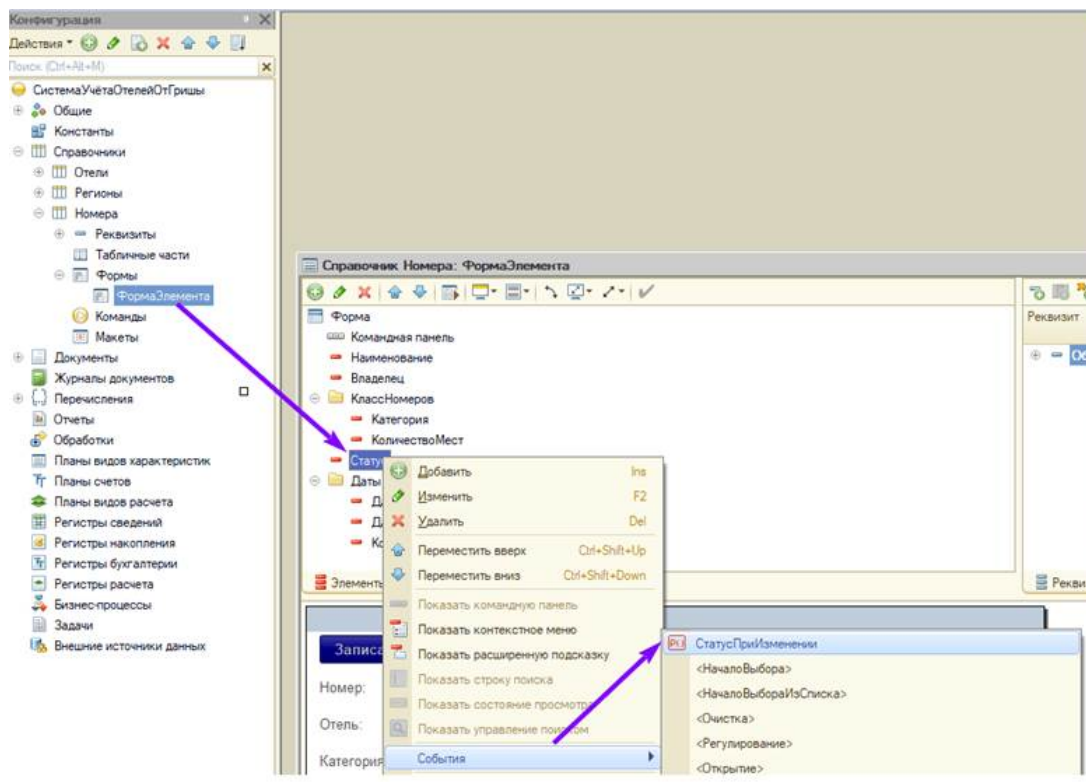
- Далее мы формируем структуру справочника «Номера» согласно техническому заданию, обязательно указывая при создании того или иного реквизита в свойствах его тип данных.



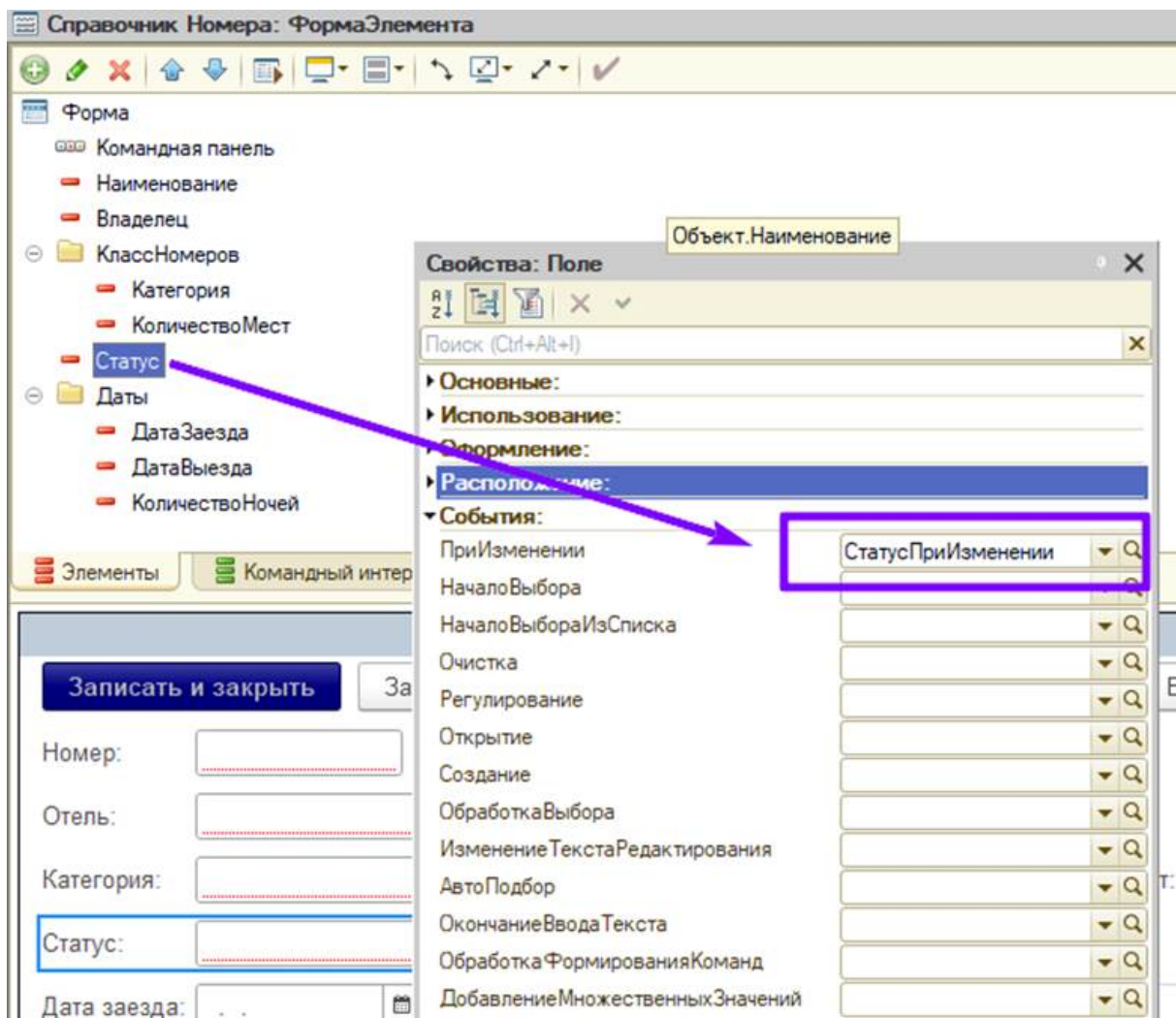
- При создании реквизитов «Категория» и «Статус» мы увидим, что потребуется список значений для хранения категорий. Поскольку список будет редактироваться крайне редко, то можно воспользоваться реквизитом «Перечисление».



5. Далее нужно создать механизм отображения дат заезда и выезда в карточке номера при выборе статуса «Занят». Т. е. у нас происходит событие — смена статуса. Это событие необходимо описать. Поскольку событие происходит на форме объекта, то логично создать новую форму объекта (автоматическая уже не подойдет, т. к. нам надо описывать свои события), и в ней прописать алгоритм данного события.



Полный код одного из вариантов решения приведен ниже. Но это не единственный вариант. Также обращаем внимание, что в «1С:Предприятие 8» недостаточно прописать код в модуле формы — важно связать процедуры описания событий с самими элементами форм (событие — свойство элемента формы).



```

1  &НаСервере
2  Процедура КатегорияПриИзмененииНаСервере()
3      Если Объект.Категория = Перечисления.Категории.Стандарт Или Объект.Категория
4          ↪ = Перечисления.Категории.Люкс Тогда
5
6          Объект.КоличествоМест = 2;
7
8      Иначе
9
10         Объект.КоличествоМест = 4;
11
12     КонецЕсли;
13 КонецПроцедуры
14
15 &НаКлиенте
16 Процедура КатегорияПриИзменении(Элемент)
17     КатегорияПриИзмененииНаСервере();
18 КонецПроцедуры
19
20 &НаСервере
21 Процедура СтатусПриИзмененииНаСервере()
22     Если Объект.Статус = Перечисления.Статус.Занят Или Объект.Статус =
23         ↪ Перечисления.Статус.Занят_Грязный Тогда
24
25         Элементы.Даты.Видимость = Истина;

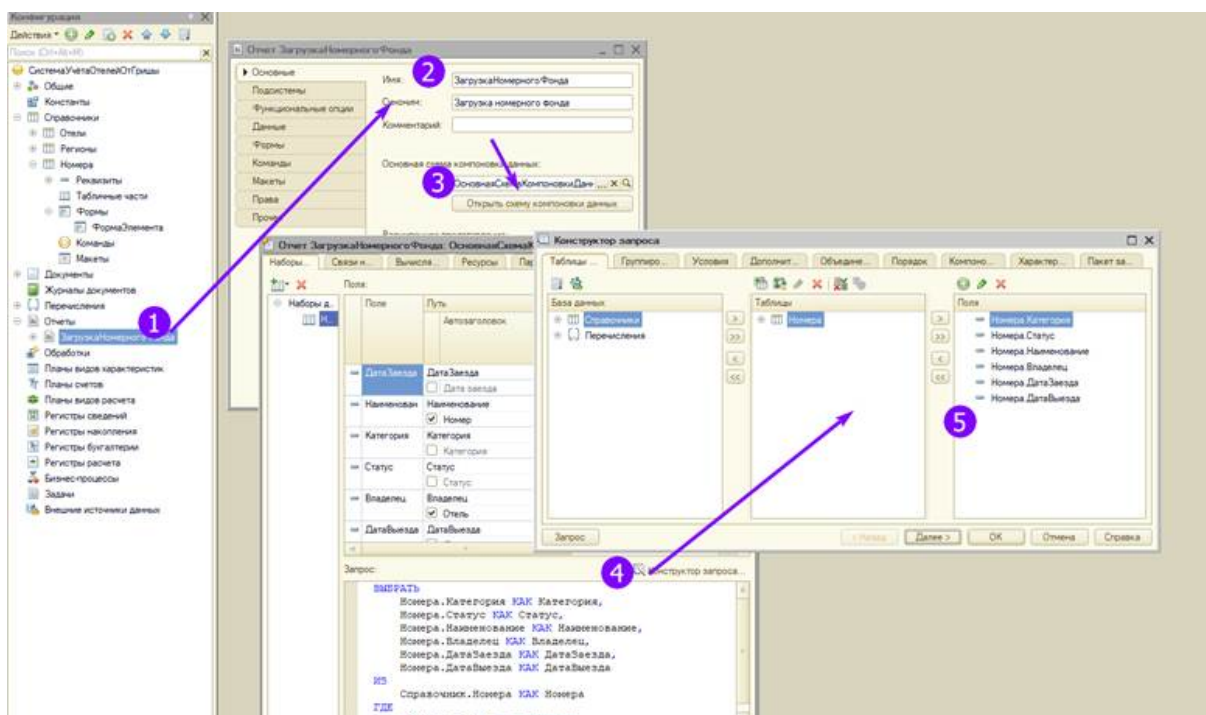
```

```

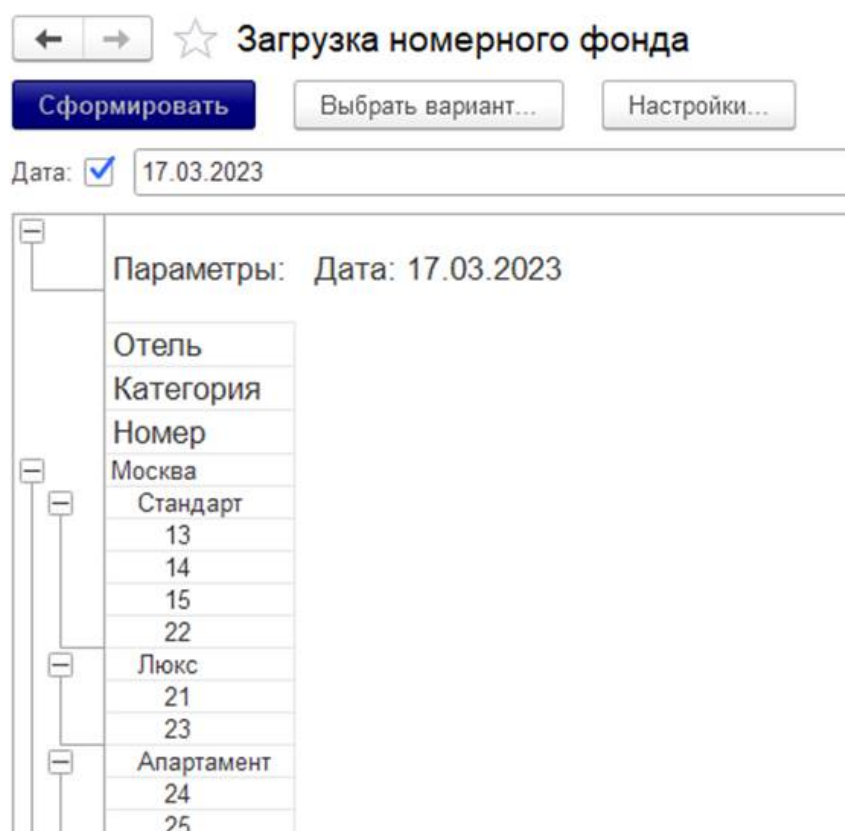
26     Иначе
27
28         Элементы.Даты.Видимость = Ложь;
29
30     КонецЕсли;
31 КонецПроцедуры
32
33 &НаКлиенте
34 Процедура СтатусПриИзменении(Элемент)
35     СтатусПриИзмененииНаСервере();
36 КонецПроцедуры
37
38 &НаКлиенте
39 Процедура ДатаВыездаПриИзменении(Элемент)
40     Если Объект.ДатаЗаезда="" Тогда
41         Сообщить("Заполните дату заезда.");
42         Объект.КоличествоНочей=0;
43     ИначеЕсли Объект.ДатаВыезда<Объект.ДатаЗаезда Тогда
44         Сообщить("Поправьте дату выезда или заезда");
45         Объект.КоличествоНочей=0;
46     Иначе
47         Объект.КоличествоНочей=(Объект.ДатаВыезда-Объект.ДатаЗаезда)/3600/24;
48     КонецЕсли;
49 КонецПроцедуры
50
51 &НаКлиенте
52 Процедура ДатаЗаездаПриИзменении(Элемент)
53     Если Объект.ДатаВыезда="" Тогда
54         Сообщить("Заполните дату выезда.");
55         Объект.КоличествоНочей=0;
56
57     ИначеЕсли Объект.ДатаВыезда<Объект.ДатаЗаезда Тогда
58         Сообщить("Поправьте дату выезда или заезда");
59         Объект.КоличествоНочей=0;
60
61     Иначе
62         Объект.КоличествоНочей=(Объект.ДатаВыезда-Объект.ДатаЗаезда)/3600/24;
63     КонецЕсли;
64 КонецПроцедуры
65
66 &НаСервере
67 Процедура ПередЗаписьюНаСервере(Отказ, ТекущийОбъект, ПараметрыЗаписи)
68     Если Объект.Статус = Перечисления.Статус.Занят Или Объект.Статус =
69     ↪ Перечисления.Статус.Занят_Грязный Тогда
70         Если Объект.ДатаВыезда < Объект.ДатаЗаезда Тогда
71             Сообщить("Поправьте дату выезда или заезда.");
72
73             Отказ = Истина;
74         КонецЕсли;
75     КонецЕсли;
76 КонецПроцедуры

```

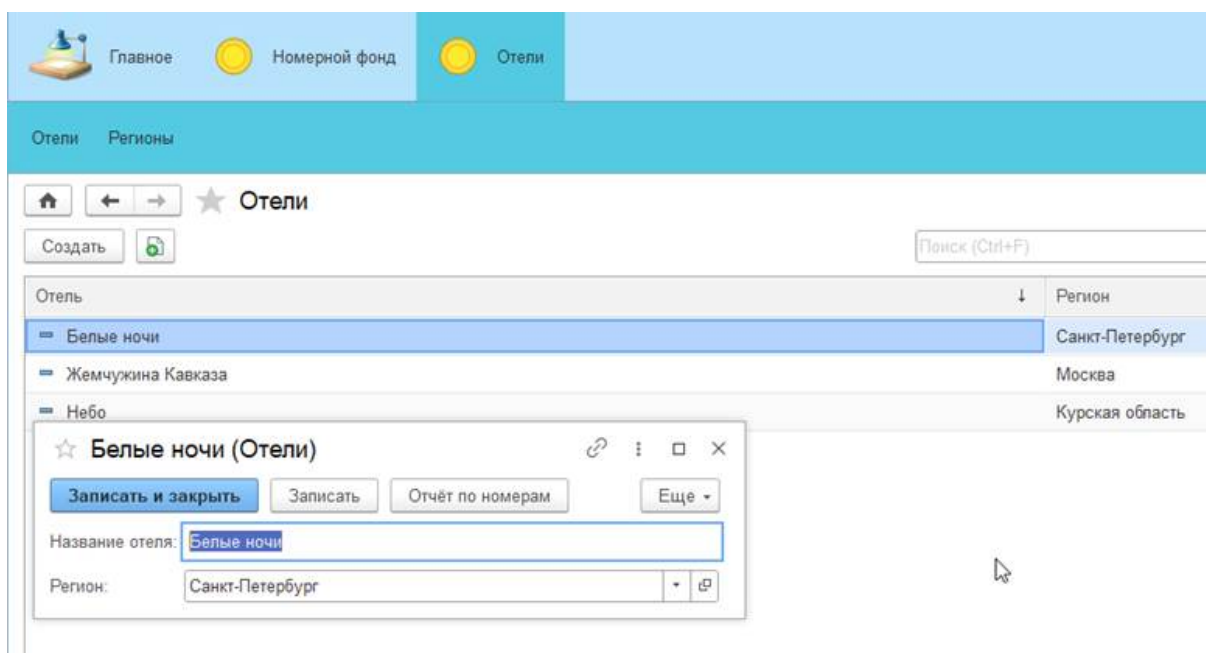
6. Следующим пунктом необходимо создать отчет о занятости номерного фонда того или иного отеля. Отчет можно сделать при помощи объекта «Отчет» и специального конструктора «Схемы компоновки данных».



Примерный вид отчета.



7. В приложении необходимо ввести тестовые данные — для проверки себя. Пример заполненной базы данных.



8. Также в задаче обращается внимание на внешний вид приложения: умение настраивать цветовую гамму, располагать объекты удобным образом, добавлять пиктограммы, изображения и т. д. На это нужно обращать внимание. Конечно, баллы не убираются за то, что был выбран не тот цвет, но если используется стандартный интерфейс, то это минус, баллы согласно критериям.

Создание механизма бронирования

Номера в отеле предварительно бронируются. Цель данной сессии — разработать механизм бронирования номеров в отеле на определенные даты. Забронированные номера необходимо учитывать структуре загрузки номерного фонда.

Длительность сессии: 2 академических часа.

Техническое задание

1. Создать структуру хранения информации о постояльцах отеля в виде справочника «Гости».

Ф. И. О. (текстовая строка до 150 символов)

Телефон (текстовая строка, предусмотреть ввод в определенном формате, например: +7(9xx) xxx-xx-xx)

2. В номера могут заселяться до 4-х человек, при этом платательщик может быть один — либо физическое лицо, либо юридическое лицо (компания оплачивает своему сотруднику проживание в отеле, либо турагентство оплачивает за своих клиентов). Поэтому необходимо создать структуру хранения информации о клиентах в виде справочника «Клиенты».

Платательщик (текстовая строка до 150 символов)

Вид платательщика (список значений из двух элементов — физ. лицо/юр. лицо, нередактируемый пользователем)

-
3. Разработать «Журнал бронирования», где будут храниться сведения о брони номеров на определенные даты. Бронирование номера — это определенное событие, которое происходит в определенную дату. Бронирование можно реализовать в виде документа.
 4. Структура объекта будет следующей.

Дата (тип «Дата»)
Заказчик (ссылка на справочник «Клиенты»)
Отель (ссылка на справочник «Отели»)
Бронируемый номер (доступны для выбора только номера со статусом «Свободен» и «Свободен (грязный)»)
Дата заезда (тип «Дата»)
Дата выезда (тип «Дата»)
Стоимость (тип «Число», два знака после запятой)

5. Номер бронируется по предоплате. Т. е. оплата — подтверждение брони. Необходимо реализовать данный механизм.
 - 5.1. Добавить в документ возможность подтвердить бронь.
 - 5.2. При подтверждении брони появляется документ оплаты со следующей структурой (соответствующий объект предварительно нужно создать).

Основание (ссылка на документ бронирования)
Сумма (подставляется целиком из документа бронирования, с возможностью корректировки)

- 5.3. После подтверждения бронирования статус номера меняется «Занят» с указанием соответствующих дат занятости. Статус номера можно менять либо автоматически, либо интерфейсно (менеджером вручную). Разработчик выбирает способ смены самостоятельно. Если статус номера меняется интерфейсно, необходимо создать удобный механизм смены статуса.
- 5.4. Если оплаты нет (бронь не подтверждена), то статус номера не меняется.

Тестирование

1. Создать не менее 2-х документов бронирования с подтвержденной бронью.
2. Создать не менее одного документа с неподтвержденной бронью.
3. Посмотреть, как изменяются данные в отчете по загрузке временного фонда.

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

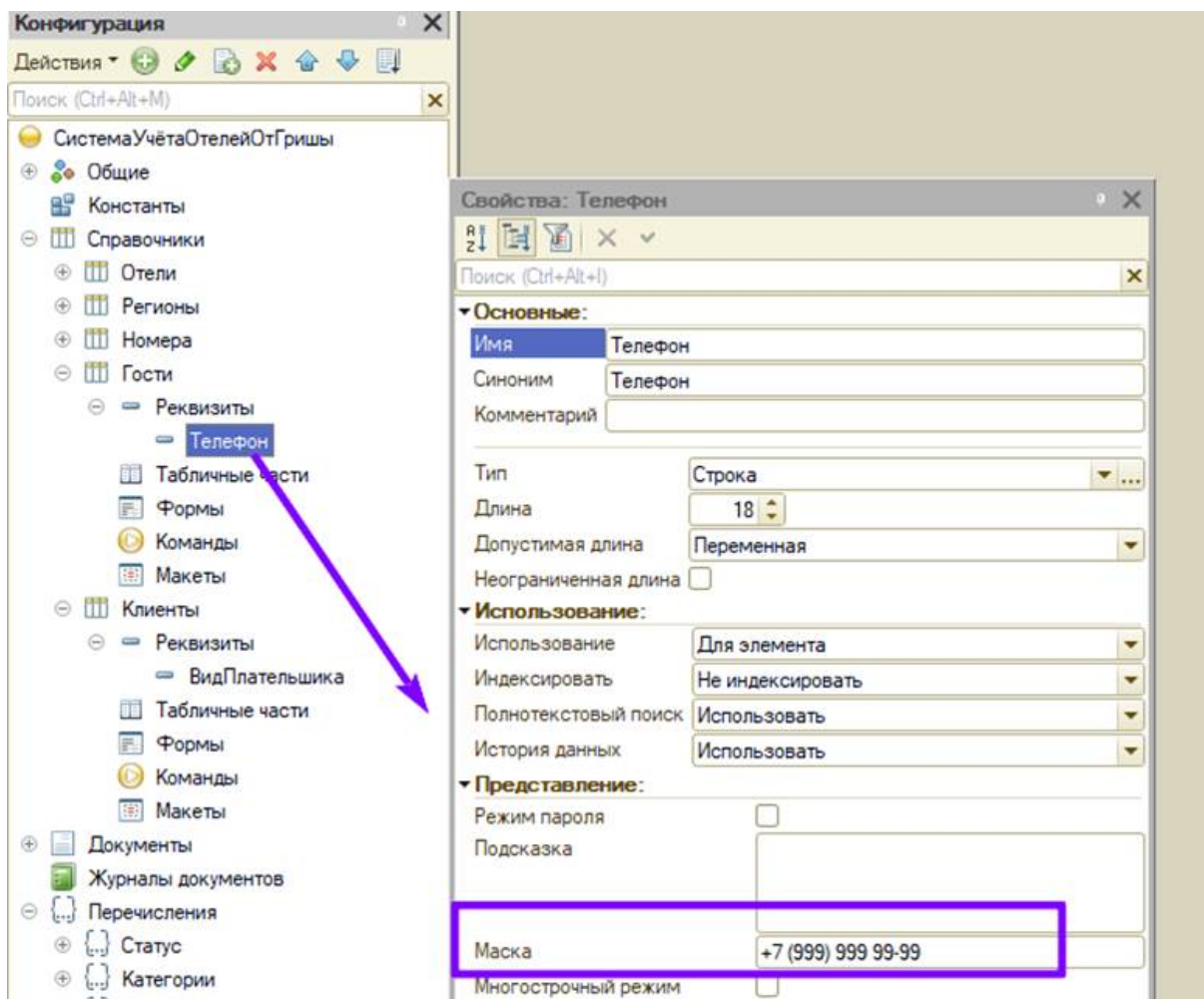
Штраф за отсутствие пояснительной записки — 1 балл.

Критерии	Базовый балл	Прибавка	Макс. балл
Созданная таблица (справочник) «Гости»	5		5
Реквизит (характеристика) «Телефон»	2	Если реализован ввод телефона в определенном формате +6	8
Созданная таблица (справочник) «Клиенты»	5		5
Реквизит (характеристика) «Вид плательщика» — список значений, не редактируемый пользователем	2	Введены виды плательщика: <ul style="list-style-type: none"> • Физическое лицо. • Юридическое лицо. +2	4
Созданная таблица (документ) для хранения информации о забронированных номерах	5		5
Реквизит «Заказчик»	2	Если реализована ссылка на справочник «Клиенты» +2	4
Реквизит «Отель»	2	Если реализована ссылка на справочник «Отели» +2	4
Реквизит (характеристика) «Дата заезда» — тип «Дата»	2		2
Реквизит (характеристика) «Дата выезда» — тип «Дата»	2		2
Реквизит «Бронируемый номер»	2	Доступны для выбора только номера со статусом «Свободен» и «Свободен (грязный)» +8	10
Реквизит «Стоимость» (тип «Число», 2 знака после запятой)	2		2
Созданная таблица (документ) для хранения информации об оплате	5		5
Реквизит таблицы (документа) «Основание»	2	Если реализована ссылка на документ бронирования +2	4
Реквизит таблицы (документа) «Сумма»	2	Подставляется из документа бронирования с возможностью корректировки +8	10
Реализация механизма подтверждения брони из карточки документа бронирования (например, по кнопке/ссылке «Подтвердить бронь» открывается документ оплаты, после ввода документа, появляется признак «Бронь подтверждена»). Решение может быть разным, описывается в пояснительной записке		+2 если есть попытка реализации, но с ошибками или некорректной работой +10 если все работает корректно	10

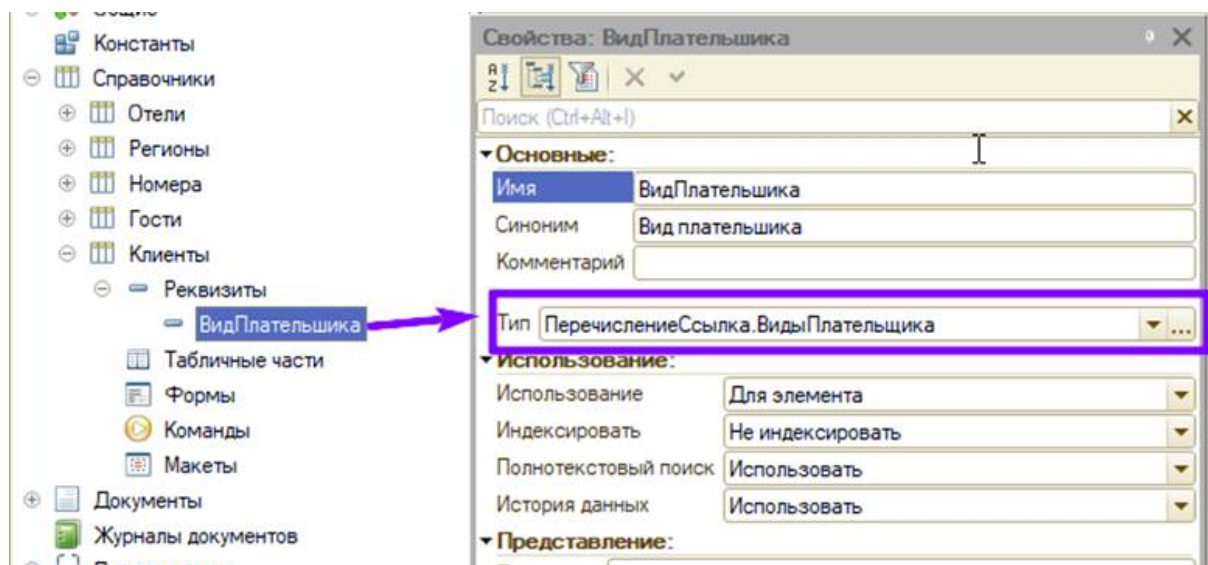
Критерии	Базовый балл	Прибавка	Макс. балл
Реализация механизма смены статуса номера на «Занят» (варианты решения могут быть разные, необходимо описать в пояснительной записке)		+2 если есть попытка реализации с некорректной работой +10 механизм работает корректно	10
Тестирование – ввод документов бронирования и оплаты		+5 если документы введены по условиям тестирования в ТЗ	5
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		до +5, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	5

Основные этапы решения

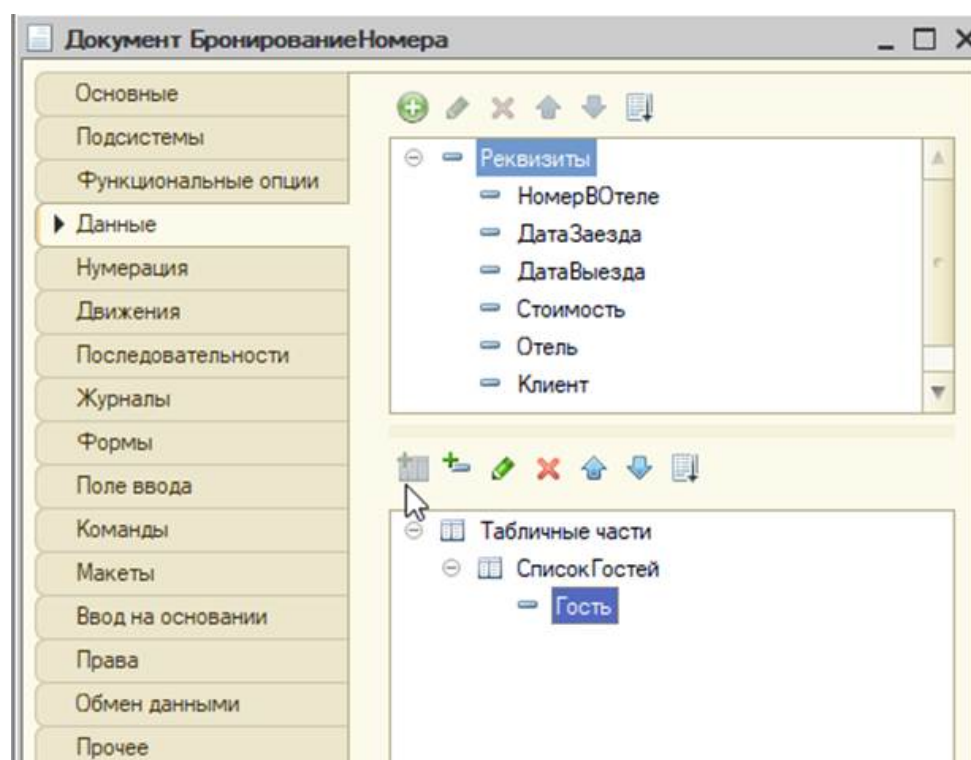
1. Создать два справочника — «Гости» и «Клиенты». В первом справочнике будет храниться информация о тех, кто приезжает, во втором — кто оплачивает. У справочника «Гости» должен быть реквизит «Телефон». Причем необходимо создать маску для ввода телефона — чтобы его можно было ввести в определенном формате, удобном для пользователя.



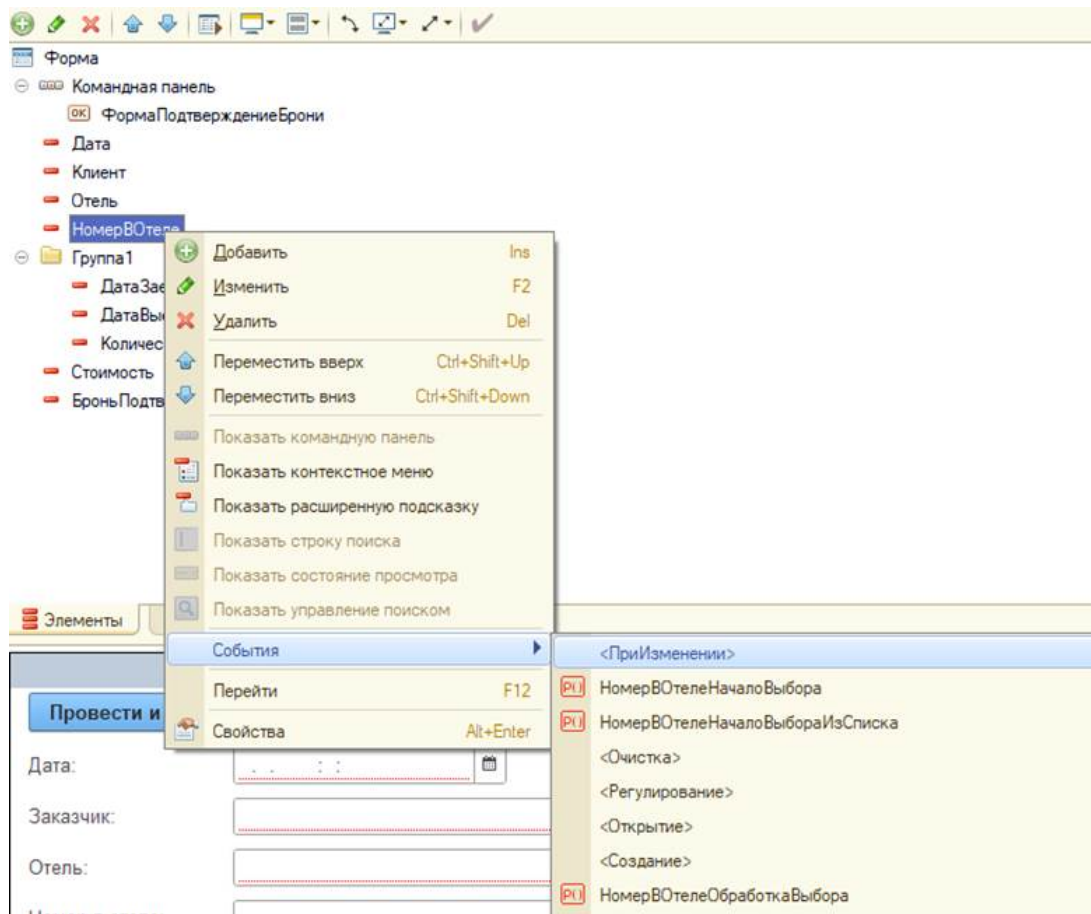
У справочника «Клиенты» должен быть реквизит «ВидПлательщика» (выбирается один из двух вариантов — физическое или юридическое лицо) с типом ссылка на соответствующее перечисление (его надо создать дополнительно).



2. Далее создаем объект, предназначенный о хранении данных по бронированию номеров. Бронирование номеров — это событие. Поэтому для хранения таких данных необходимо создать документ «БронированиеНомера» с реквизитами, перечисленными в ТЗ.



Здесь же необходимо описать алгоритм доступности номеров для выбора со статусами «Свободен_чистый» и «Свободен_грязный». Поскольку выбор происходит на форме, то это означает, что нам нужно создать форму документа и описать алгоритм выбора номера.



Приводим один из вариантов программного кода.

```

1  &НаСервере
2  Функция ПолучитьСтатусСвободенГрязный()
3      Возврат Перечисления.Статусы.СвободенГрязный;
4  КонецФункции
5
6  &НаСервере
7  Функция ПолучитьСтатусСвободенЧистый()
8      Возврат Перечисления.Статусы.СвободенЧистый;
9  КонецФункции
10
11 &НаСервере
12 Функция ПолучитьОтель()
13     Возврат Объект.Отель;
14 КонецФункции
15
16 &НаСервере
17 Функция ПроверкаОтеля(Отель)
18     Возврат Отель = Справочники.Отели.ПустаяСсылка();
19 КонецФункции
20
21 &НаКлиенте
22 Процедура НомерВОтелеНачалоВыбора(Элемент, ДанныеВыбора, СтандартнаяОбработка)
23     СтандартнаяОбработка = Ложь;
24     Если ПроверкаОтеля(Объект.Отель) Тогда
25         Сообщить("Для выбора номера необходимо указать отель");
26         Возврат;
27     КонецЕсли;
28     Оповещение = Новый
29         ↳ ОписаниеОповещения("ОбработкаВыбораРезультата",ЭтотОбъект);

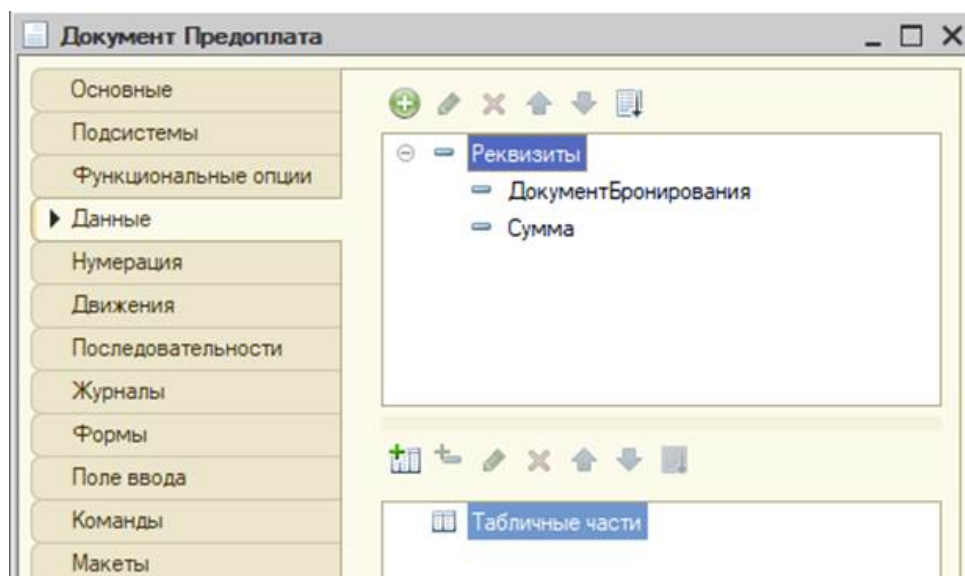
```

```

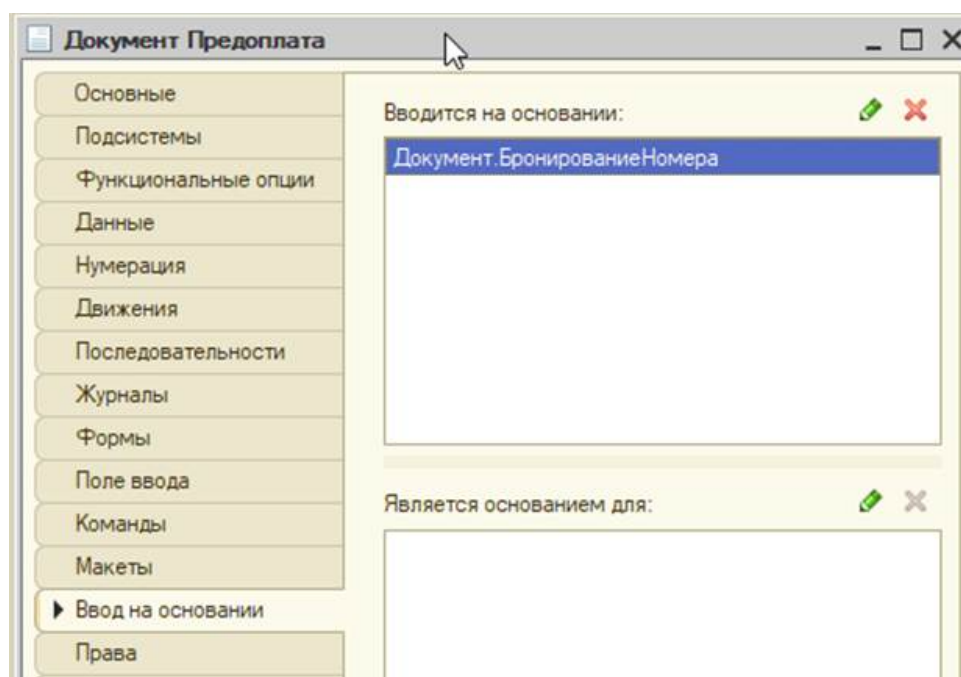
30     ОтборПараметры = Новый Структура;
31     ОтборПараметры.Вставить("Статус", ПолучитьСтатусСвободенГрязный());
32     ОтборПараметры.Вставить("Статус", ПолучитьСтатусСвободенЧистый());
33     ОтборПараметры.Вставить("Отель", ПолучитьОтель());
34     ОтборПараметры.Вставить("ЭтоГруппа", Ложь);
35     ПараметрыФормы = Новый Структура("Отбор", ОтборПараметры);
36     ОткрытьФорму("Справочник.Номера.ФормаВыбора", ПараметрыФормы,,,Оповещение,
    ↪     РежимОткрытияОкнаФормы.БлокироватьВесьИнтерфейс);
37     КонецПроцедуры
38
39     &НаКлиенте
40
41     Процедура ОбработкаВыбораРезультата(РезультатЗакрытия, ДополнительныеПараметры)
    ↪     Экспорт
42         Объект.НомерВОтеле = РезультатЗакрытия;
43     КонецПроцедуры
44
45     &НаКлиенте
46     Процедура НомерВОтелеОбработкаВыбора(Элемент, ВыбранноеЗначение,
    ↪     СтандартнаяОбработка)
47         Если Не НомерВОтелеОбработкаВыбораНаСервере(ВыбранноеЗначение) Тогда
48             СтандартнаяОбработка = ложь;
49             Сообщить("Выбранный номер не удовлетворяет условиям");
50         КонецЕсли;
51     КонецПроцедуры
52
53     &НаСервере
54     Функция НомерВОтелеОбработкаВыбораНаСервере(ВыбранноеЗначение)
55         Если ВыбранноеЗначение.Отель = Объект.Отель и
56             (ВыбранноеЗначение.Статус = Перечисления.Статусы.СвободенГрязный или
57             ВыбранноеЗначение.Статус = Перечисления.Статусы.СвободенЧистый) Тогда
58             Возврат Истина
59         Иначе
60             Возврат Ложь;
61         КонецЕсли;
62     КонецФункции

```

3. Далее необходимо описать механизм ввода оплаты на основании документа бронирования. Очевидно, оплата — это тоже событие, поэтому для хранения данных будем использовать документ.



Очевидно, что тип данных реквизита «ДокументБронирования» — ссылка на документ «БронированиеНомера». Документ оплаты вводится на основании документа бронирование, необходимо отразить это в свойствах и описать механизм записи данных при вводе на основании.



4. После подтверждения бронирования статус номера меняется «Занят» с указанием соответствующих дат занятости. Статус номера можно менять либо автоматически, либо интерфейсно (менеджером вручную). Разработчик выбирает способ смены самостоятельно. Если статус номера меняется интерфейсно, необходимо создать удобный механизм смены статуса. Если оплаты нет (бронь не подтверждена), то статус номера не меняется.
5. Также в этой сессии необходимо ввести тестовые данные и поработать над интерфейсом.

Создание механизма оформления заезда/выезда

Разработать механизм оформления заезда/выезда с удобным интерфейсом для пользователя.

Удобный интерфейс предполагает некий рабочий стол с быстрым доступом к основным операциям за небольшое количество кликов.

Длительность сессии: 2 академических часа.

Техническое задание

1. Создать механизм оформления заезда в отель.
 - 1.1. Заезд гостя в отель — это событие. Событие происходит всегда в определенную дату. Это необходимо учитывать при выборе объекта для реализации механизма.

1.2. Т. к. номера в отеле предварительно бронируются, заезд «привязывается» к созданному ранее документу бронирования. Другими словами, документ заезда создается на основании документа бронирования.

1.3. Структура записи о заезде будет следующая.

Дата оформления заезда (тип «Дата»)
Отель (тип – ссылка на справочник «Отели»)
Занимаемый номер (доступны для выбора только номера со статусами «Свободен (чистый)» и «Свободен (грязный)»)
Дата выезда (тип «Дата»)
Список гостей (таблица из одного столбца, каждый элемент таблицы – ссылка на справочник «Гости»)

1.4. При оформлении заезда меняется статус выбранного номера на «Занят» (если этот статус был установлен документом бронирования, он не меняется).

1.5. Добавить информацию о гостях номера в отчет о загрузке номерного фонда. Т. е. при нажатии на ячейку с конкретным номером можно было бы увидеть расшифровку — кем занят и в какие даты.

2. Реализовать механизм выезда из отеля.

2.1. Выезд из отеля — это тоже событие, происходящее в определенную дату. Реализовать механизм можно по аналогии с механизмом заезда, учитывая, что это обратная операция.

- Структура записи о выезде будет следующая.

Дата выезда (тип «Дата»)
Отель (тип — ссылка на справочник «Отели»)
Освобождаемый номер (доступны для выбора только номера со статусом «Занят» и «Занят (грязный)»)
Список ушедших гостей (таблица из одного столбца, каждый элемент таблицы — ссылка на справочник «Гости»)

- При выборе номера, который освобождается, нужно сделать так, чтобы список ушедших гостей заполнялся автоматически.
- При оформлении выезда статус указанного номера меняется на «Свободен (грязный)».

3. Создать удобный интерфейс (рабочий стол) для менеджера по оформлению заезда выезда из определенного отеля.

3.1. Заезд — кликом по кнопке открывается форма список «К оформлению», состоящий из списка брони номеров с датой заезда на текущую дату. Если гость предварительно не бронировал номер, должна быть предусмотрена возможность оформления заезда (например, по кнопке, без документа бронирования).

- 3.2. Выезд — кликом открывается список номеров «К освобождению». В этот список попадают номера, у которых дата выезда — текущая дата. Около каждого номера к освобождению стоит кнопка «Оформить выезд». При нажатии на кнопку открывается форма оформления выезда.

Тестирование

1. Зарегистрировать в системе не менее 2-х документов заезда и 1-го документа выезда.
2. Проверить, как меняется информация в отчете о загрузке номерного фонда при регистрации документов.

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

Штраф за отсутствие пояснительной записки — 1 балл.

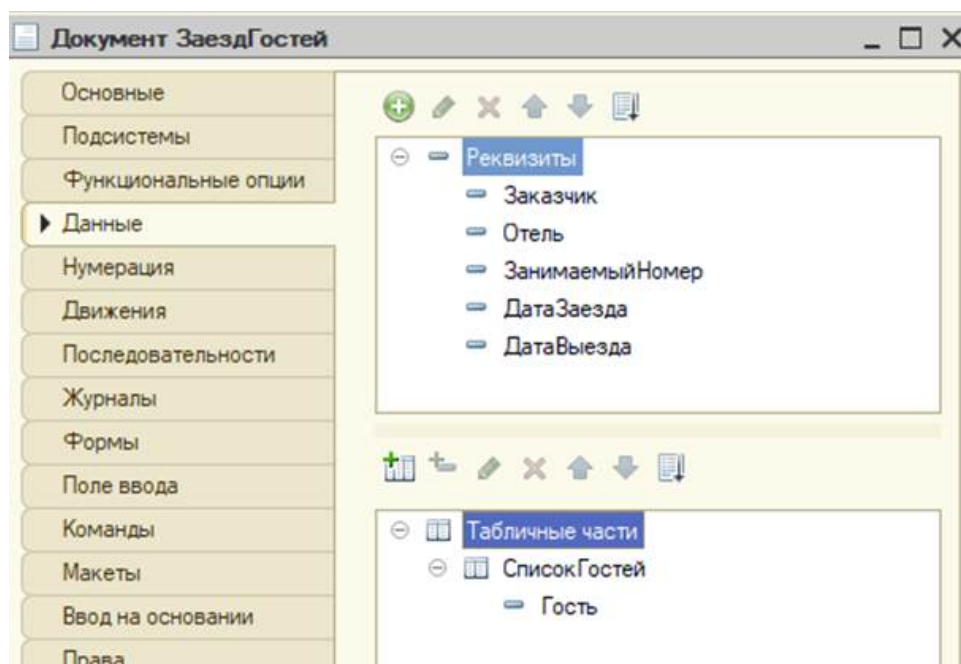
Критерии	Базовый балл	Прибавка	Макс. балл
Созданная таблица (документ) для хранения информации о заезде гостя	5	Если документ вводится на основании документа бронирования, но скопирована не вся информация +2 Документ вводится на основании документа бронирования, и вся информация скопирована +5	10
Реквизит «Отель»	2	Если реализована ссылка на справочник «Отели» +2	4
Реквизит (характеристика) «Дата заезда» — тип «Дата»	2		2
Реквизит (характеристика) «Дата выезда» — тип «Дата»	2		2
Созданная таблица «Список гостей» с одной колонкой — «Гость»	3	Если реализована ссылка на справочник «Гости» +3	6
При оформлении заезда меняется статус выбранного номера на «Занят» (если этот статус был установлен документом бронирования, он не меняется)		Если при вводе документа заезда на основании документа бронирования статус не изменяется +2	2
Реализация механизма добавления информации о гостях номера в отчет о загрузке номерного фонда. Т. е. при нажатии на ячейку с конкретным номером можно было бы увидеть расшифровку — кем занят, и в какие даты.		Если есть попытка создания расшифровки занятости +2 Расшифровка работает корректно +10	10

Критерии	Базовый балл	Прибавка	Макс. балл
Созданная таблица (документ) для хранения информации о выезде гостя	5	Если документ вводится на основании документа заезда, но скопирована не вся информация +2 Документ вводится на основании документа бронирования, и вся информация скопирована +5	10
Реквизит «Отель»	2	Если реализована ссылка на справочник «Отели» +2	4
Освобождаемый номер (доступны для выбора только номера со статусом «Занят» и «Занят (грязный)»)	2	Если реализован механизм смены статуса номера +8 (любой вариант — программный или интерфейсный)	10
Созданная таблица «Список гостей» с одной колонкой — «Гость»	3	Если реализована ссылка на справочник «Гости» +2	5
Созданный рабочий стол для менеджера по оформлению заезда выезда из определенного отеля	5		5
Реализованный механизм оформления заезда — перечень документов бронирования с подтвержденной бронью и возможностью оформить заезд, например, по кнопке		Если механизм есть, но работает с ошибками, например, попадают не только документы с подтвержденной бронью +2 Если механизм работает корректно +10	10
Реализована возможность оформить заезд без предварительного бронирования (например, по кнопке «Оформить заезд без бронирования», при нажатии на которую появляется документ заезда и возможность оплаты)		Есть попытка реализации +3 Механизм оформления заезда работает корректно (вводится документ заезда и на его основании оплата) +10	
Реализованный механизм оформления выезда — перечень документов заезда, у которых указана дата выезда, равная текущей		Если механизм есть, но работает с ошибками, например, попадают или наоборот не попадают документы, соответствующие критерию +2 Если механизм работает корректно +10	10
Тестирование — ввод документов заезда и выезда		+5 если документы введены по условиям тестирования в ТЗ	5

Критерии	Базовый балл	Прибавка	Макс. балл
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		До +5, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	5

Основные этапы решения

1. Заезд гостя в отель — это событие. Это событие оформляется документом «ЗаездГостей». Его нужно создать в приложении. Реквизиты описаны в ТЗ.



2. Документ заезда может оформляться по предварительному бронированию, т. е. вводиться на основании документа «БронированиеНомера».
3. При оформлении заезда меняется статус выбранного номера на «Занят» (если этот статус был установлен документом бронирования, он не меняется). В отчет о загрузке номерного фонда необходимо добавить информацию о гостях занятых номеров. Варианты отображения могут быть различные, достаточно вот такого варианта.

← → ☆ Загрузка номерного фонда

Сформировать Выбрать вариант... Настройки...

Отель: Жемчужина

Параметры: Отель: Жемчужина

Номер	Отель	Свободен занят	Дата заезда	Дата выезда	Клиент
11	Жемчужина	Свободен(чистый)			
11	Жемчужина	Свободен(чистый)			
12	Жемчужина	Свободен(чистый)			
12	Жемчужина	Свободен(чистый)			
13	Жемчужина	Занят	17.03.2023	19.03.2023	Фомина Анастасия Сергеевна
13	Жемчужина	Свободен(чистый)			

4. Выезд гостя — это событие, которое также необходимо описать в виде документа со структурой, описанной в ТЗ. Обратите внимание, что выезд гостя оформляется на основании документа заезда. Т. е. мы должны связать цепочку событий, связанных с определенными заказчиками и гостями (Бронирование — Оплата — Заезд — Выезд).

Документ ВыездГостей

Основные
Подсистемы
Функциональные опции
Данные
Нумерация
Движения
Последовательности
Журналы
Формы
Поле ввода
Команды
Макеты
Ввод на основании
Права

Реквизиты

- Отель
- Освобождаемый номер

Табличные части

- Выезжающие
 - Гость

Документ ВыездГостей

Вводится на основании:

Документ.ЗаездГостей

Является основанием для:

5. Важный пункт, где проверяется и умение работать с интерфейсом, и навыки программирования — создание «Интерфейса менеджера», где бы отображались документы бронирования с текущей датой (для быстрого ввода документа заезда) и документы заезда с текущей датой выезда — для быстрого оформления выезда. Рабочий стол можно было создать в виде обработки, либо другим каким-то способом.

← → Рабочий стол менеджера

Отель: Москва

Заезд Выезд

Поиск (Ctrl+F)

Отель	Дата заезда	Дата выезда	Заказчик
Москва	07.04.2023	10.04.2023	Соколов Максим Сергеевич

6. Также в этой сессии необходимо ввести тестовые данные и поработать над интерфейсом.

Ценообразование и оплата

В отелях присутствуют номера трех категорий:

- Стандарт.
- Люкс.
- Апартамент.

Стоимость каждого номера за сутки различается. Более того, стоимость номера той же категории может различаться в разных отелях.

Необходимо разработать механизм определения цены на номер конкретной категории в том или ином отеле и механизм подстановки автоматической стоимости проживания в документы бронирования и оплаты.

Длительность сессии: 2 академических часа.

Техническое задание

1. Реализовать механизм для назначения и хранения цен на номера различных категорий в различных отелях. Структура хранения информации о ценах будет иметь приблизительно следующий вид.

Отель:	Небо	Жемчужина
Категория:		
Стандарт	2500 р./сут	3800 р./сут
Люкс	3500 р./сут	5000 р./сут
Апартамент	5000 р./сут	7000 р./сут

- 1.1. Способ реализации выбирается разработчиком самостоятельно. В «1С» для создания таких таблиц удобно использовать объект под названием «Регистр сведений».
- 1.2. Структура хранилища цен будет следующая.

«Отель» — ссылка на справочник «Отели»

«Категория» — список значений, не редактируемых пользователем

«Цена» — тип «Число», два знака после запятой

- 1.3. Следует предусмотреть, что цены могут изменяться. Т. е. сегодня номер категории «Стандарт» в отеле «Небо» стоит 2500 р./сут., через неделю цена этого номера составит 3000 р./сут.
2. Создать отчет «Прайс-лист», отображающий стоимость проживания в сутки в том или ином отеле в разрезе категорий.
 - 2.1. Отчет должен представлять собой таблицу, подобную таблице п. 1 данного «Технического задания» .
 - 2.2. Отчет строится на определенную дату, т. к. цены могут изменяться.
3. Реализовать механизм автоматической подстановки цены в документ бронирования (возможность ручного редактирования необходимо оставить) и расчет общей стоимости проживания.
 - 3.1. Добавить в документ бронирования поле «Цена» (тип «число», 2 знака после запятой) и реализовать автоматическую подстановку цены.
 - 3.2. Реализовать автоматический расчет общей стоимости проживания, который осуществляется по формуле:

$$\text{Стоимость проживания} = \text{Цена} \times \text{Количество дней.}$$

Тестирование

1. Ввести сведения о ценах на номера в отелях. На две даты — любую прошедшую и текущую.
2. Сформировать прайс-лист на разные даты.
3. Проверить механизм автоматической подстановки цены в документ бронирования — создать документ бронирования и на его основании ввести документ оплаты.

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

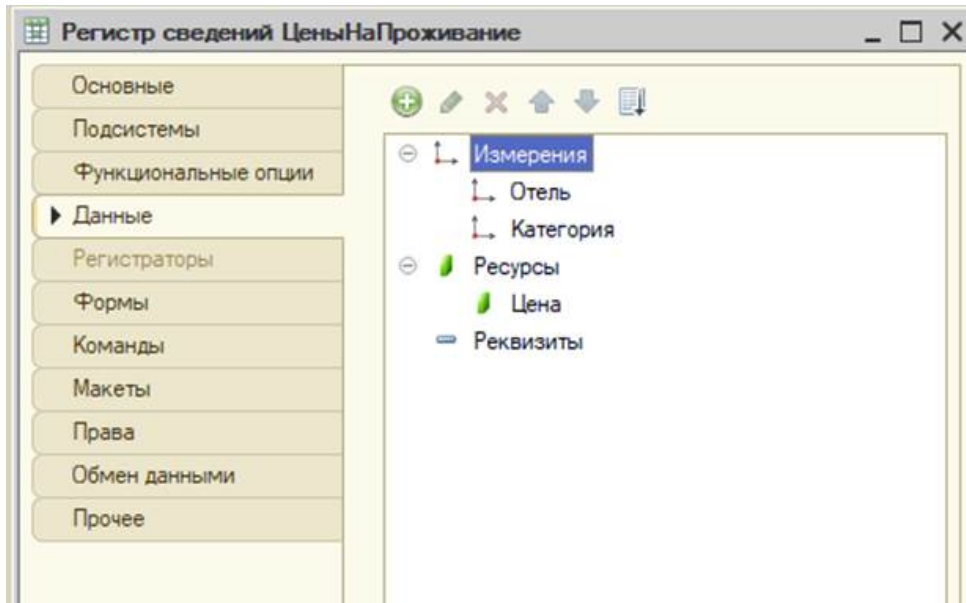
Штраф за отсутствие пояснительной записки — 1 балл.

Критерии	Базовый балл	Прибавка	Макс. балл
Созданное хранилище (регистр сведений) для хранения информации о ценах	15		15
Реквизит «Отель»	2	Если реализована ссылка на справочник «Отели» +2	4

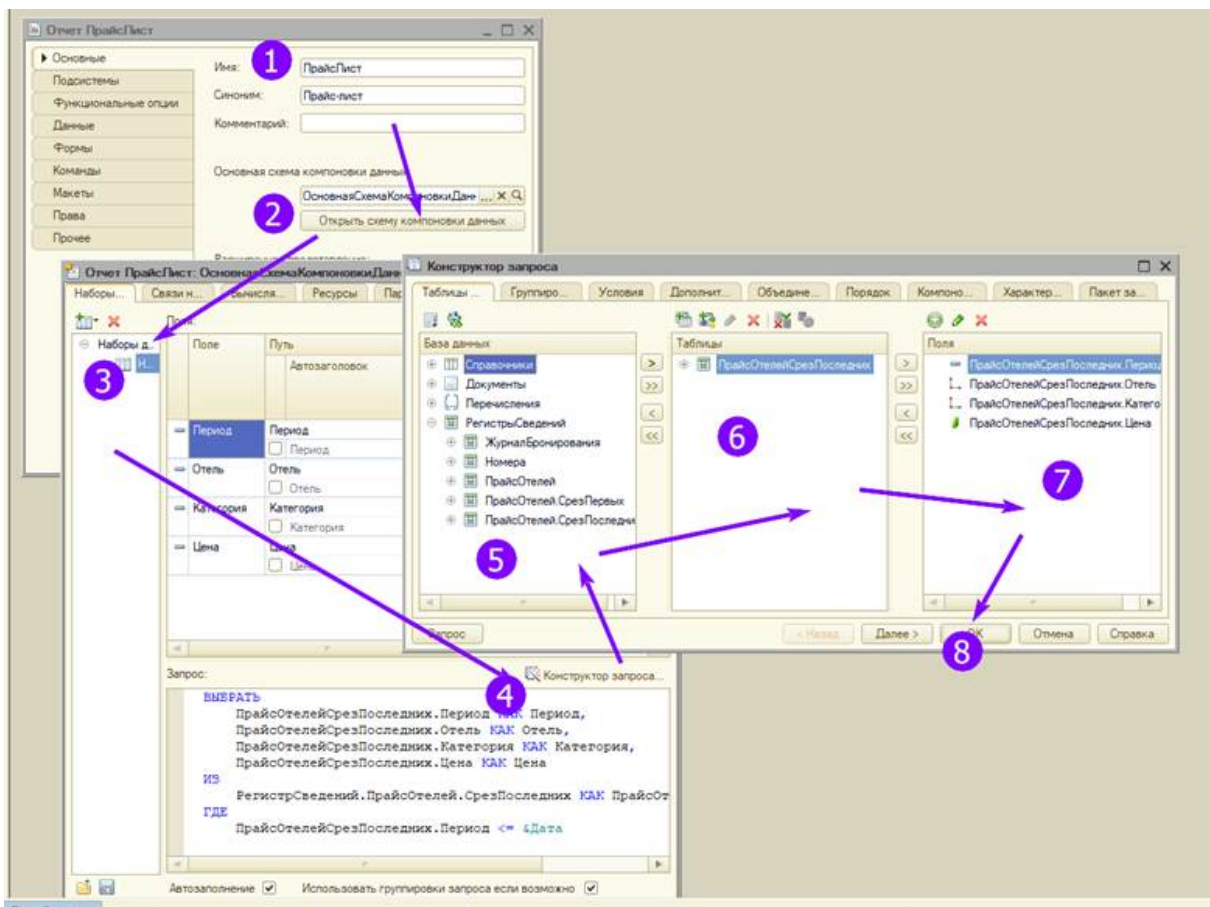
Критерии	Базовый балл	Прибавка	Макс. балл
Реквизит (характеристика) «Категория» — список значений, редактируемый пользователем.	2	Введены категории номеров: <ul style="list-style-type: none"> • Стандарт. • Люкс. • Апартамент. +2	4
Реквизит «Цена» (тип «Число», 2 знака после запятой)	2		2
Реализован механизм хранения цены в разрезе дат		+5	5
Созданный отчет «Прайс-лист»		Если есть попытка создания отчета +5 Если отчет создан, но отображает некорректные данные +9 Если отчет работает корректно +20	20
Добавление поля «Цена» в документ бронирования		Если поле добавлено, но цена не подставляется из хранилища (регистра сведений) информации о ценах +9 Если цена подставляется автоматически из хранилища +20	20
Реализация механизма расчета общей стоимости проживания в документе бронирования (цена умножить на количество ночей)		Если есть попытка реализации механизма, но с ошибками +7 Если механизм работает +20	20
Тестирование — заполнение хранилища информации о ценах данными на разные даты		+5 если тестирование произведено по правилам ТЗ	5
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		До +5, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	5

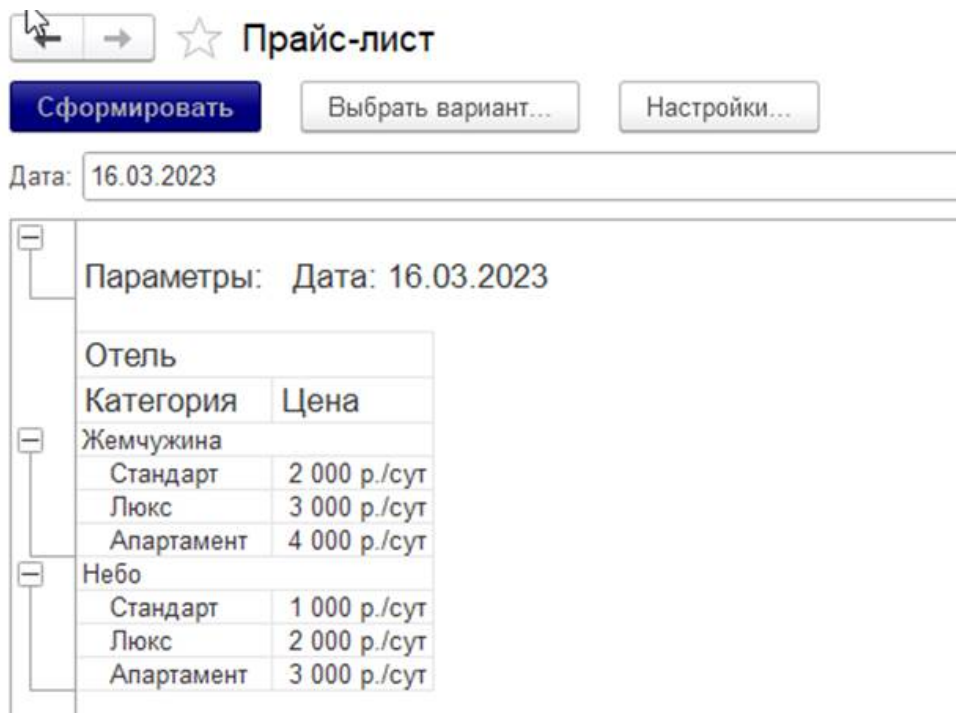
Основные этапы решения

1. В первую очередь необходимо реализовать механизм для назначения и хранения цен на номера различных категорий в различных отелях. В «1С:Предприятие» 8 для таких целей обычно используют объект «Регистр сведений».



2. Регистр будет периодическим, т. к. цены меняются. На закладке «Основные» необходимо изменить периодичность, указав, например, «В пределах дня».
3. На основании данных этого регистра необходимо построить отчет «Прайс-лист».





4. Далее необходимо реализовать механизм подстановки цен в документ бронирования из регистра сведений, где хранятся эти цены. Поскольку подстановка осуществляется в оперативной памяти компьютера, когда мы создаем документ, то все алгоритмы можно описать в модуле формы документа бронирования. Обратите внимание, что цена должна подставляться при следующих событиях: выборе номера, изменении даты заезда и выезда. Общая стоимость рассчитывается как цена, умноженная на количество ночей. Ниже пример программного кода — один из вариантов решения.

```

1  &НаСервере
2  Процедура НомерВОтелеПриИзмененииНаСервере()
3      Оте=Объект.Отель;
4      Спр=Справочники.Номера.НайтиПоНаименованию(Объект.БронируемыйНомер);
5      Кат=Спр.Категория;
6
7      Запрос = Новый Запрос;
8      Запрос.Текст =
9          "ВЫБРАТЬ
10             |      ПрайсОтелейСрезПоследних.Отель КАК Отель,
11             |      ПрайсОтелейСрезПоследних.Категория КАК Категория,
12             |      ПрайсОтелейСрезПоследних.Цена КАК Цена
13             |ИЗ
14             |      РегистрСведений.ПрайсОтелей.СрезПоследних КАК
15             ↳ ПрайсОтелейСрезПоследних";
16
17      РезультатЗапроса = Запрос.Выполнить();
18
19      ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();
20
21      Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
22          Если ВыборкаДетальныеЗаписи.Отель=Оте и
23             ↳ ВыборкаДетальныеЗаписи.Категория=Кат Тогда
24              Объект.Цена=ВыборкаДетальныеЗаписи.Цена;
25              КонецЕсли;
26      КонецЦикла;
27      Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;

```

```
26  КонецПроцедуры
27
28  &НаКлиенте
29  Процедура НомерВОтелеПриИзменении(Элемент)
30      НомерВОтелеПриИзмененииНаСервере();
31  КонецПроцедуры
32
33  &НаКлиенте
34  Процедура ЦенаПриИзменении(Элемент)
35      Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;
36  КонецПроцедуры
37
38  &НаКлиенте
39  Процедура КоличествоНочейПриИзменении(Элемент)
40      Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;
41  КонецПроцедуры
```

5. Также в этой сессии необходимо ввести тестовые данные и поработать над интерфейсом.

Разработка механизма лояльности (предоставления скидок)

Разработать механизм предоставления скидок при бронировании в отеле с различным вариантом популярности направления (популярное, среднее, мало популярное). Если отель принадлежит к категории «Малопопулярное направление», предоставить при бронировании его номера скидку в 15%.

Второй вариант предоставления скидки — «скидка в мертвый сезон», 20%.

Длительность сессии: 2 академических часа.

Техническое задание

1. Создать категории направлений.

- Популярное.
- Среднее по популярности.
- Непопулярное.

Это список значений, не редактируемый пользователем.

- 1.1. Добавить возможность указания направления для конкретного отеля.
- 1.2. Реализовать механизм расчета стоимости со скидкой 10% при выборе отеля с направлением «Непопулярное» в документе бронирования.
2. Разработать систему предоставления скидки в «мертвый сезон» по следующей схеме: если дата заезда попадает в даты мертвого сезона, определенного для каждого отеля, то клиенту предоставляется скидка в 20% от общей стоимости бронирования.
 - 2.1. Создать возможность проставления дат «мертвого сезона». Тип — «Дата».

- 2.2. Реализовать механизм расчета стоимости со скидкой в 20%, если дата заезда попадает в «мертвый сезон».

Тестирование

1. Заполнить направления для каждого отеля.
2. Создать документы бронирования номера в отеле с разными вариантами направления, и проверить правильность работы механизма расчета скидки для непопулярного направления.
3. Проставить даты «мертвого сезона» для каждого сезона.
4. Создать документы бронирования: 2 документа с датой заезда в мертвый сезон (по разным отелям) и 2 документа по разным отелям с датой заезда вне «мертвого сезона». Проверить правильность работы механизма.

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

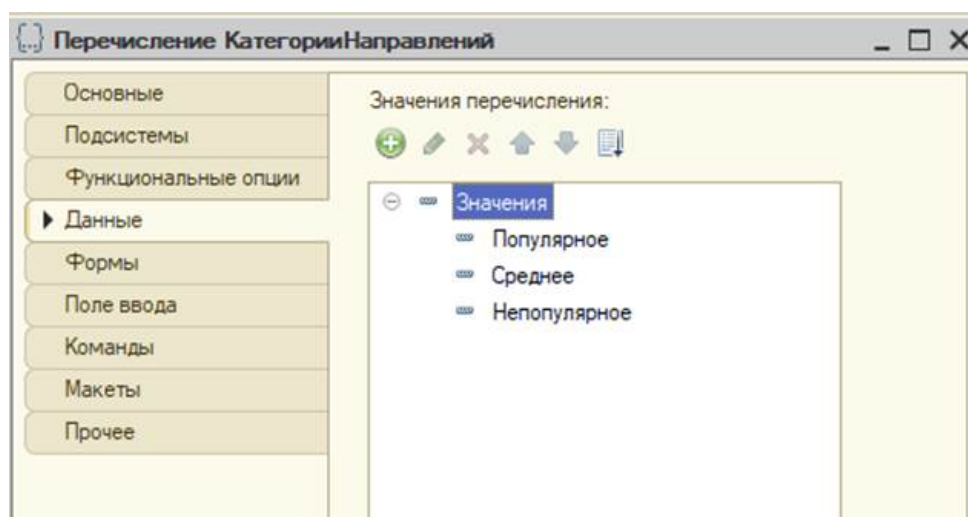
Штраф за отсутствие пояснительной записки — 1 балл.

Критерии	Базовый балл	Прибавка	Макс. балл
Созданные категории направлений (список значений, редактируемый пользователями)	5	Заполненные категории: <ul style="list-style-type: none"> • Популярное • Среднее по популярности • Непопулярное +5	10
Добавление поля «Направление» для конкретного отеля	+2	Направление — ссылка на категории направлений +3	5
Механизм расчета стоимости со скидкой 10% при выборе отеля с направлением «Непопулярное» в документе бронирования.		Есть попытка создания механизма +9 Механизм расчета скидки работает +25	25
Механизм определения дат «мертвого сезона» — тип «Дата» (варианты различные — в карточке отеля, отдельный справочник и т. д.)	5		5
Механизм предоставления скидки в 20%, если бронирование осуществляется в «мертвый сезон» (дата заезда попадает в «мертвый сезон»)		Есть попытка создания механизма +9 Механизм расчета скидки работает +25	25

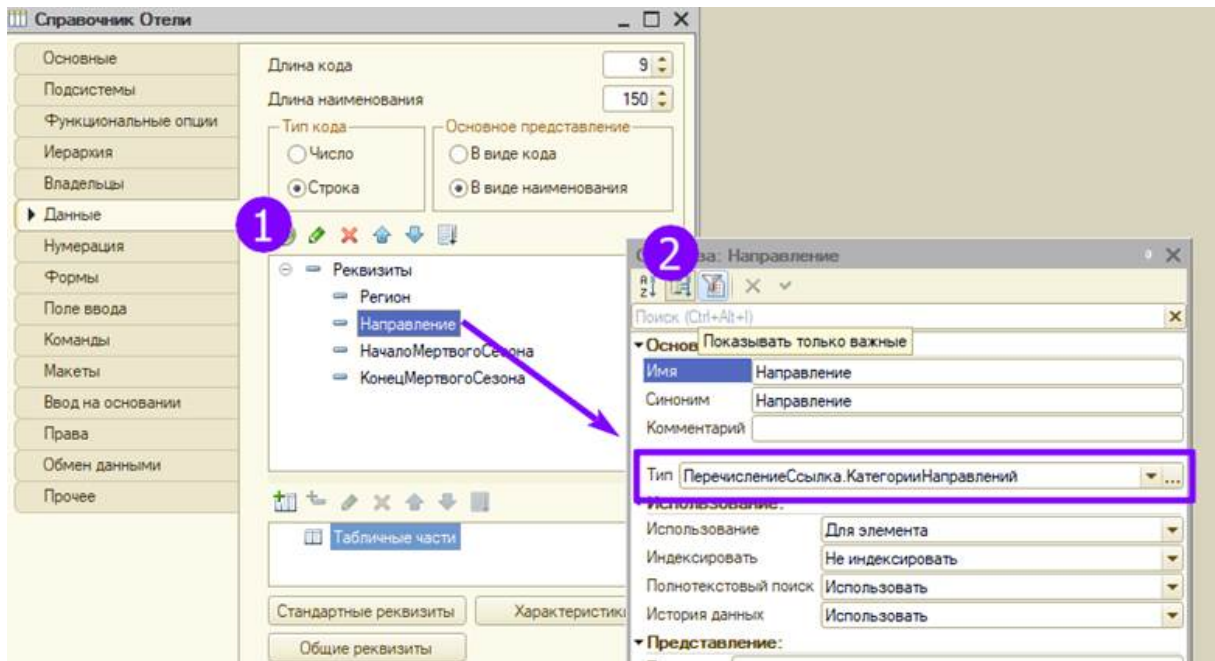
Критерии	Базовый балл	Прибавка	Макс. балл
Тестирование. Созданные документы бронирования с учетом направлений — проверка работы расчета скидки по направлениям	5		5
Тестирование. Проставленные даты мертвого сезона в каждом отеле	5		5
Тестирование. Созданные документы бронирования с датой, попадающий в «мертвый сезон» и не попадающие в «мертвый сезон» — проверка работы механизма расчета скидки в «мертвый сезон»	5		5
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		До +15, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	15

Основные этапы решения

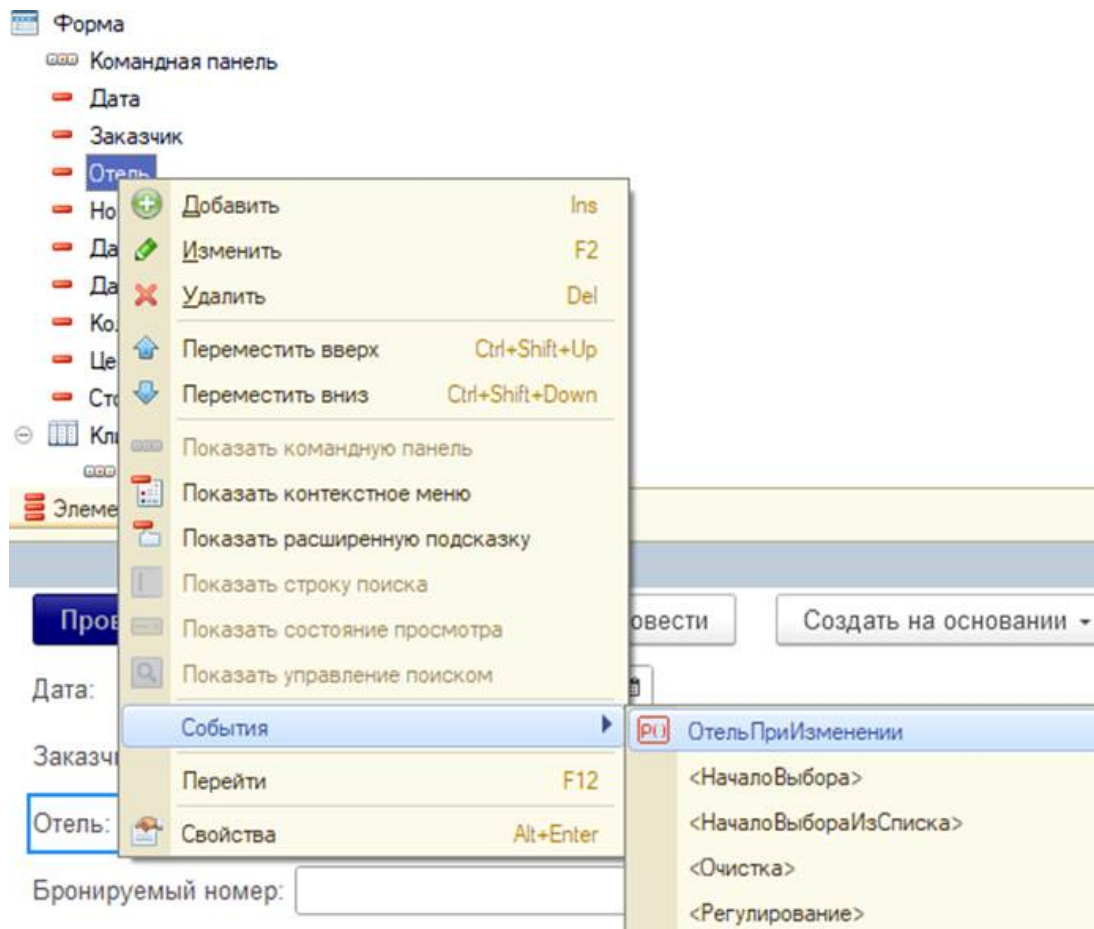
1. Создаем перечень направлений. Т. к. по ТЗ он почти не редактируется, его также можно создать в виде перечисления. Но можно сделать и справочником.



Тут же необходимо добавить для справочника «Отели» новый реквизит.



2. Далее необходимо разработать механизм предоставления скидки в 10%, если отель находится на непопулярном направлении. Очевидно скидка рассчитывается при выборе отеля. Выбор отеля в документе бронирования — это событие.



Один из вариантов описания алгоритма.

- 1 &НаСервере
- 2 Процедура Сумма()

```

3     Множитель = 1;
4     Если Объект.Отель.КатегорияНаправления =
      ↳ Перечисления.КатегорииНаправлений.Непопулярное Тогда
5         Множитель = 0.9;
6     КонецЕсли;
7     Объект.Стоимость = Объект.КоличествоДней * Объект.Цена * Множитель;
8     КонецПроцедуры
9
10    &НаКлиенте
11    Процедура ДатаВыездаПриИзменении(Элемент)
12        Если Объект.ДатаЗаезда="" Тогда
13            Сообщить("Заполните дату заезда.");
14            Объект.КоличествоДней=0;
15            Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;
16        ИначеЕсли Объект.ДатаВыезда<Объект.ДатаЗаезда Тогда
17            Сообщить("Поправьте дату выезда или заезда");
18            Объект.КоличествоДней=0;
19            Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;
20        Иначе
21            Объект.КоличествоДней=(Объект.ДатаВыезда-Объект.ДатаЗаезда)/3600/24+1;
22            Сумма();
23        КонецЕсли;
24    КонецПроцедуры
25
26    &НаКлиенте
27    Процедура ДатаЗаездаПриИзменении(Элемент)
28        Если Объект.ДатаВыезда="" Тогда
29            Сообщить("Заполните дату выезда.");
30            Объект.КоличествоДней=0;
31            Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;
32        ИначеЕсли Объект.ДатаВыезда<Объект.ДатаЗаезда Тогда
33            Сообщить("Поправьте дату выезда или заезда");
34            Объект.КоличествоДней=0;
35            Объект.Стоимость=Объект.Цена*Объект.КоличествоДней;
36        Иначе
37            Объект.КоличествоДней=(Объект.ДатаВыезда-Объект.ДатаЗаезда)/3600/24+1;
38            Сумма();
39        КонецЕсли;
40    КонецПроцедуры
41
42    &НаСервере
43    Процедура НомерВОтелеПриИзмененииНаСервере()
44        Оте=Объект.Отель;
45        Спр=Справочники.Номера.НайтиПоНаименованию(Объект.БронируемыйНомер);
46        Кат=Спр.Категория;
47
48        Запрос = Новый Запрос;
49        Запрос.Текст =
50            "ВЫБРАТЬ
51             |     ПрайсОтелейСрезПоследних.Отель КАК Отель,
52             |     ПрайсОтелейСрезПоследних.Категория КАК Категория,
53             |     ПрайсОтелейСрезПоследних.Цена КАК Цена
54             |ИЗ
55             |     РегистрСведений.ПрайсОтелей.СрезПоследних КАК
      ↳ ПрайсОтелейСрезПоследних";
56
57        РезультатЗапроса = Запрос.Выполнить();
58
59        ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();
60

```

```

61     Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
62     Если ВыборкаДетальныеЗаписи.Отель=Оте и
        ↳ ВыборкаДетальныеЗаписи.Категория=Кат Тогда
63         Объект.Цена=ВыборкаДетальныеЗаписи.Цена;
64     КонецЕсли;
65     КонецЦикла;
66     Сумма();
67 КонецПроцедуры
68
69 &НаКлиенте
70 Процедура НомерВОтелеПриИзменении(Элемент)
71     НомерВОтелеПриИзмененииНаСервере();
72 КонецПроцедуры
73
74 &НаКлиенте
75 Процедура ЦенаПриИзменении(Элемент)
76     Сумма();
77 КонецПроцедуры

```

Пример реализованного расчета.

← → ☆ Бронирование (создание) *

Провести и закрыть Записать Провести Создать на основании ▾

Дата: 07.04.2023 0:00:00 📅

Заказчик: Фролова Ксения Юрьевна ▾ 📄

Отель: Небо ▾ 📄

Бронируемый номер: 14 ▾ 📄

Дата заезда: 07.04.2023 📅

Дата выезда: 10.04.2023 📅

Количество дней: 4

Цена: 1 000,00 📄

Стоимость: 3 600,00 📄

Скидка 10%!

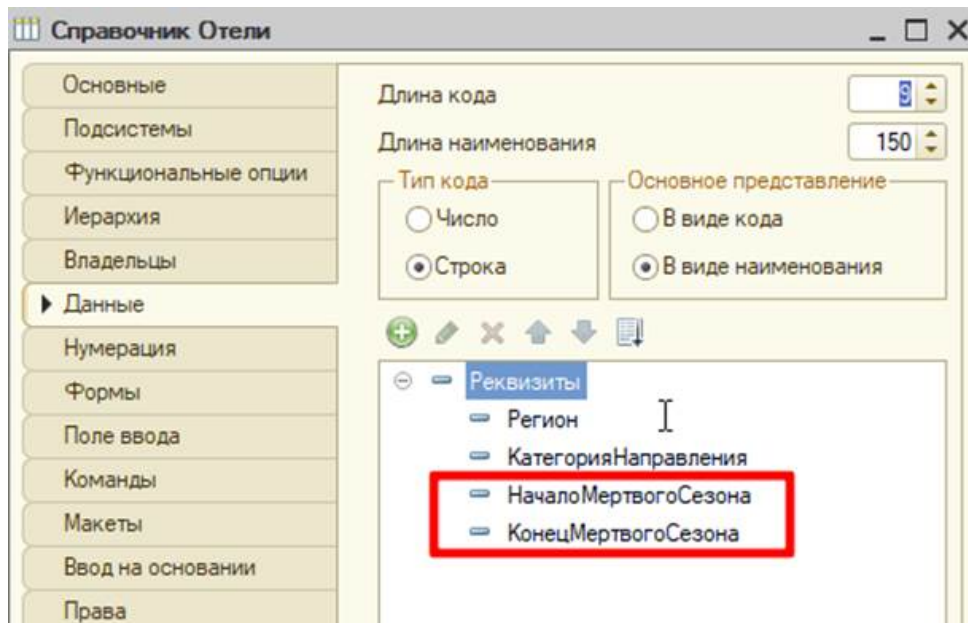
Добавить ⤴ ⤵

Гость

Фролов Алексей Сергеевич

Фролова Ксения Юрьевна

- Далее необходимо разработать скидку с учетом дат «мертвого сезона». Его даты указываются в справочнике «Отели». Необходимо их добавить.



4. Далее необходимо разработать алгоритм предоставления скидки в «мертвый сезон» в размере 20%. Скидки не суммируются. Выдаем наибольшую скидку для нашего клиента. С учетом этого, алгоритм расчета скидки (процедура «Сумма») представленный выше, приобретет следующий вид.

```

1 &НаСервере
2 Процедура Сумма()
3     Множитель = 1;
4     Если Объект.ДатаЗаезда >= Объект.Отель.НачалоМертвогоСезона И
   ↪ Объект.ДатаЗаезда <= Объект.Отель.КонецМертвогоСезона Тогда
5         Множитель = 0.8;
6     ИначеЕсли Объект.Отель.КатегорияНаправления =
   ↪ Перечисления.КатегорииНаправлений.Непопулярное Тогда
7         Множитель = 0.9;
8     КонецЕсли;
9     Объект.Стоимость = Объект.КоличествоДней * Объект.Цена * Множитель;
10 КонецПроцедуры
  
```

Пример реализованного механизма.

5. Также в этой сессии необходимо ввести тестовые данные и поработать над интерфейсом.

Разработка механизма аналитики (отчетов)

Для анализа и формирования дальнейшей политики развития гостиничного бизнеса необходимо опираться на различные показатели. Это и общеизвестные показатели, такие как сумма продаж, количество продаж (в нашем случае — количество проданных ночей), а также специфические показатели, которые используются в отельном бизнесе. К ним относятся:

Процент загрузки номерного фонда — это отношение количества проданных ночей к общему количеству номеров в отеле.

ADR (Average Daily Rate) — это показатель для расчета эффективности работы гостиницы. Показывает среднюю стоимость номера за некий промежуток времени. Показатель получается за счет деления чистой выручки от продаж номерного фонда (после вычета скидок, косвенных налогов, стоимости завтраков и т. п.) на общее количество проданных за рассматриваемый период номеров. Другими словами, данный показатель равен отношению суммы продаж к числу ночей.

RevPAR (средний доход на комнату) — один из самых важных показателей, так как он воплощает собой всю политику продаж. RevPAR равен выручке от продажи номерного фонда, поделенной на количество всех номеров в отеле. Иначе говоря, RevPAR равен ADR, умноженной на процент загрузки.

Цель данной сессии — разработать отчет с выводом выше перечисленных показателей по отелю на определенную дату.

Длительность сессии: 2 академических часа.

Техническое задание

1. Создать отчет, который выводит показатели эффективности по каждому отелю на определенную дату.
 - 1.1. Отчет будет включать следующие показатели и фильтры для отбора.

Анализ эффективности продаж	
Дата _____	
Отель _____	
Отель «Небо»	
Количество проданных ночей	
Общая сумма продаж	
% Загрузки номерного фонда	
Ср. цена ADR	
RevPAR	

- 1.2. Отчет должен формироваться на определенную дату, т. е. поле «Дата»

обязательна к заполнению. Без нее отчет формироваться не будет.

- 1.3. Поле «Отель» не обязательно к заполнению, если отель не выбран, то отображается информация по всем отелям на указанную дату.
2. Отчет необходимо делать на основании данных по документам бронирования. При выборе данных необходимо учесть некоторые особенности оформления документов.
 - 2.1. Брать нужно не все документы бронирования, а только с признаком «бронь подтверждена» (есть оплата).
 - 2.2. В документах бронирования нет сведений о количестве проданных ночей, но есть даты заезда и выезда, которые мы можем использовать для получения нужных нам данных.
3. На основании предыдущего пункта мы можем сделать вывод, что получить данные, необходимые для отчета, из документов бронирования достаточно трудоемко, поэтому удобнее создать отдельную таблицу под названием, например, «Продано ночей», в которую будут заноситься сведения, сколько номеров, на какую сумму какой категории и какого отеля были проданы на каждый день. А потом из этой таблицы данные взять в отчет.

В «1С:Предприятие 8» для таких целей удобно использовать объект «Регистр накопления».

- 3.1. В таблицу необходимо записывать следующие данные:
 - Отель (тип данных — ссылка на справочник «Отели»).
 - Категория (тип данных — ссылка на список категорий).
 - Количество проданных ночей (тип данных — число, неотрицательное, целое).
 - Сумма продаж (тип данных — число, неотрицательное, два знака после запятой).
- 3.2. Данные в таблицу записываются при проведении документа бронирования, если у него есть признак «бронь подтверждена» (есть оплата). Если такого признака нет, то данные в таблицу из документа бронирования не записываются.
4. Построить отчет на основании данных из таблицы «ПроданоНочей». Расчет перечисленных ранее показателей производится следующим образом:
 - 4.1. Количество проданных ночей — общее количество (сумма) проданных ночей по отелю на выбранную дату.
 - 4.2. Общая сумма продаж — сумма продаж по отелю на выбранную дату.
 - 4.3. % загрузки номерного фонда = (Количество проданных ночей) / (Общее количество номеров в отеле).
 - 4.4. $ADR = (\text{Общая сумма продаж}) / (\text{Количество проданных ночей})$
 - 4.5. $RevPAR = (\text{Общая сумма продаж}) / (\text{Общее количество номеров в отеле})$.

Тестирование

1. Проверить, что в таблицу «ПроданоНочей» попадают верные данные только из тех документов бронирования, у которых есть признак «бронь подтверждена».
 - 1.1. Создать и провести документ бронирования без признака «бронь подтверждена». Открыть таблицу — записи о проданных ночах быть не должно.
 - 1.2. Создать документ бронирования, ввести оплату, поставить признак «бронь

подтверждена». Открыть таблицу — запись о проданных ночах должна появиться.

2. Сформировать созданный отчет на выбранную дату.
3. Свериться с документами бронирования, что в отчет попадают верные данные (показатели по документам бронирования рассчитать вручную по документам и свериться с данными в отчете).

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

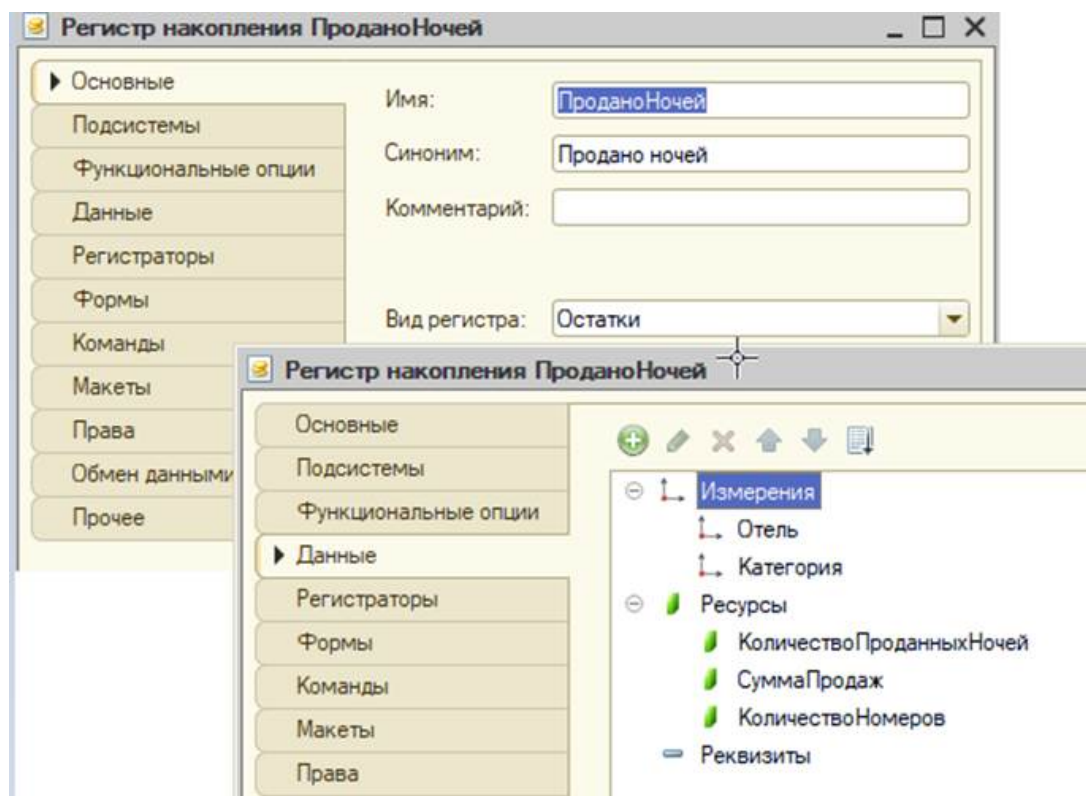
Штраф за отсутствие пояснительной записки — 1 балл.

Критерии	Базовый балл	Прибавка	Макс. балл
Созданная таблица (регистр накопления) «Продано ночей»	10		10
Реквизит «Отель»	2	Если реализована ссылка на справочник «Отели» +2	4
Реквизит «Категория»	2	Если реализована ссылка на список категорий номеров +2	4
Реквизит «Количество проданных ночей» (тип данных — число, неотрицательное, целое)	2		2
Реквизит «Сумма продаж» (тип данных — число, неотрицательное, два знака после запятой)	2		2
Реализация механизма записи данных в таблицу (регистр накопления) — данные в таблицу записываются при проведении документа бронирования, если у него есть признак «бронь подтверждена» (есть оплата). Если такого признака нет, то данные в таблицу из документа бронирования не записываются		Если есть попытка реализации механизма, но с ошибками +3 Механизм записи данных в таблицу работает корректно +10	10
Созданный отчет «Анализ эффективности продаж»	3		3
Созданный фильтр по дате	5		5
Созданный фильтр по отелю		Если есть попытка создания фильтра по отелю, но данные отбираются с ошибками +2 Если данные в зависимости от фильтра выбираются корректно +5	5

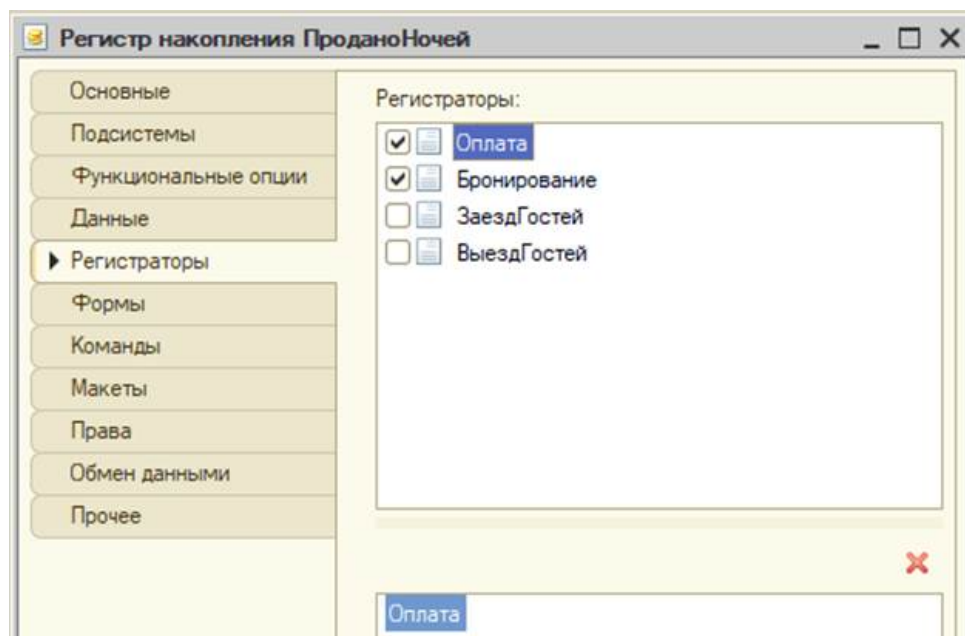
Критерии	Базовый балл	Прибавка	Макс. балл
Показатель «Количество проданных ночей» — общее количество (сумма) проданных ночей по отелю на выбранную дату		Если выводятся данные, но с ошибкой в расчете +2 Если показатель рассчитан верно +10	10
Показатель «Общая сумма продаж» — сумма продаж по отелю на выбранную дату		Если выводятся данные, но с ошибкой в расчете +2 Если показатель рассчитан верно +10	10
Показатель «% загрузки номерного фонда» равен (Количество проданных ночей) / (Общее количество номеров в отеле)		Если выводятся данные, но с ошибкой в расчете +2 Если показатель рассчитан верно +10	10
Показатель «ADR», равный (Общая сумма продаж) / (Количество проданных ночей)		Если выводятся данные, но с ошибкой в расчете +2 Если показатель рассчитан верно +10	10
Показатель «RevPAR», равен (Общая сумма продаж) / (Общее количество номеров в отеле)		Если выводятся данные, но с ошибкой в расчете +2 Если показатель рассчитан верно +10	10
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		до +5, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	5

Основные этапы решения

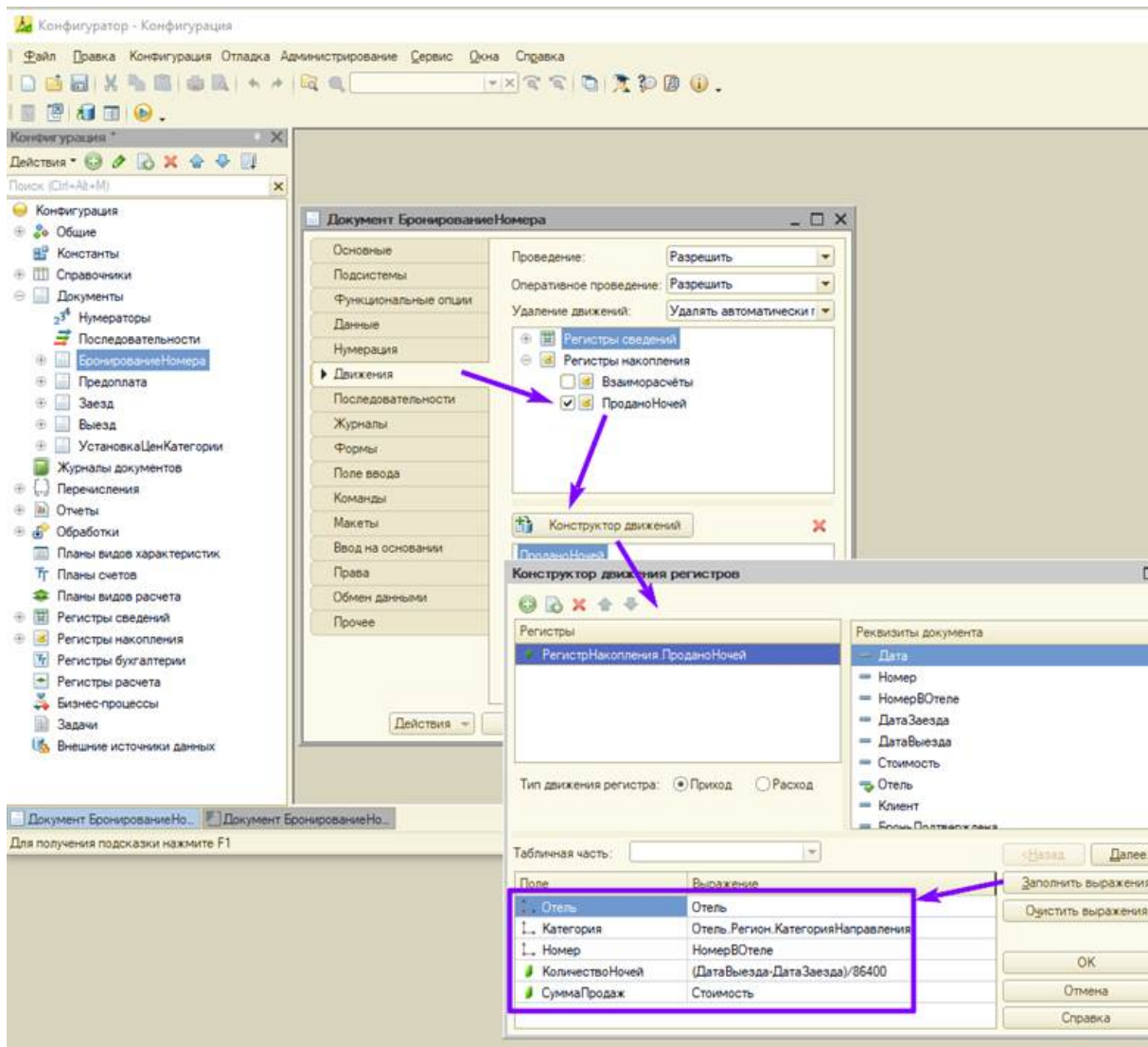
1. Создать регистр накопления для хранения суммы проданных ночей в разрезе отеля и категории. Также там можно хранить и другие данные, которые могут понадобиться при составлении отчета, например.



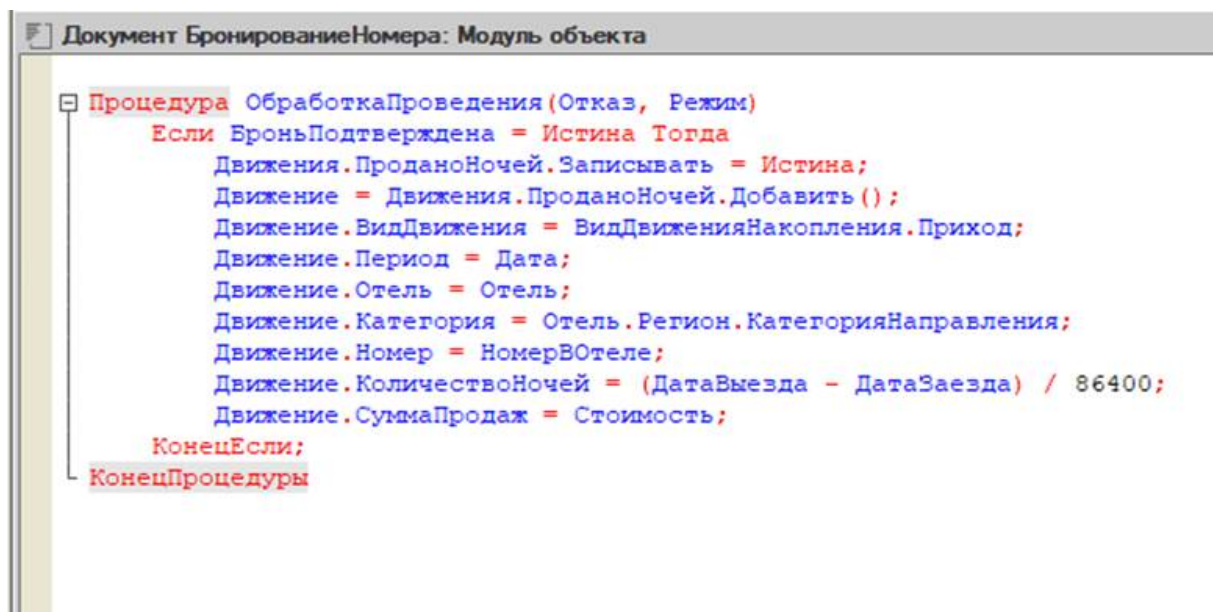
2. Регистр накопления в отличие, например, от регистра сведений не может заполняться данными вручную. «Поставщиками» данных для него являются документы. На закладке регистраторы необходимо указать эти документы.



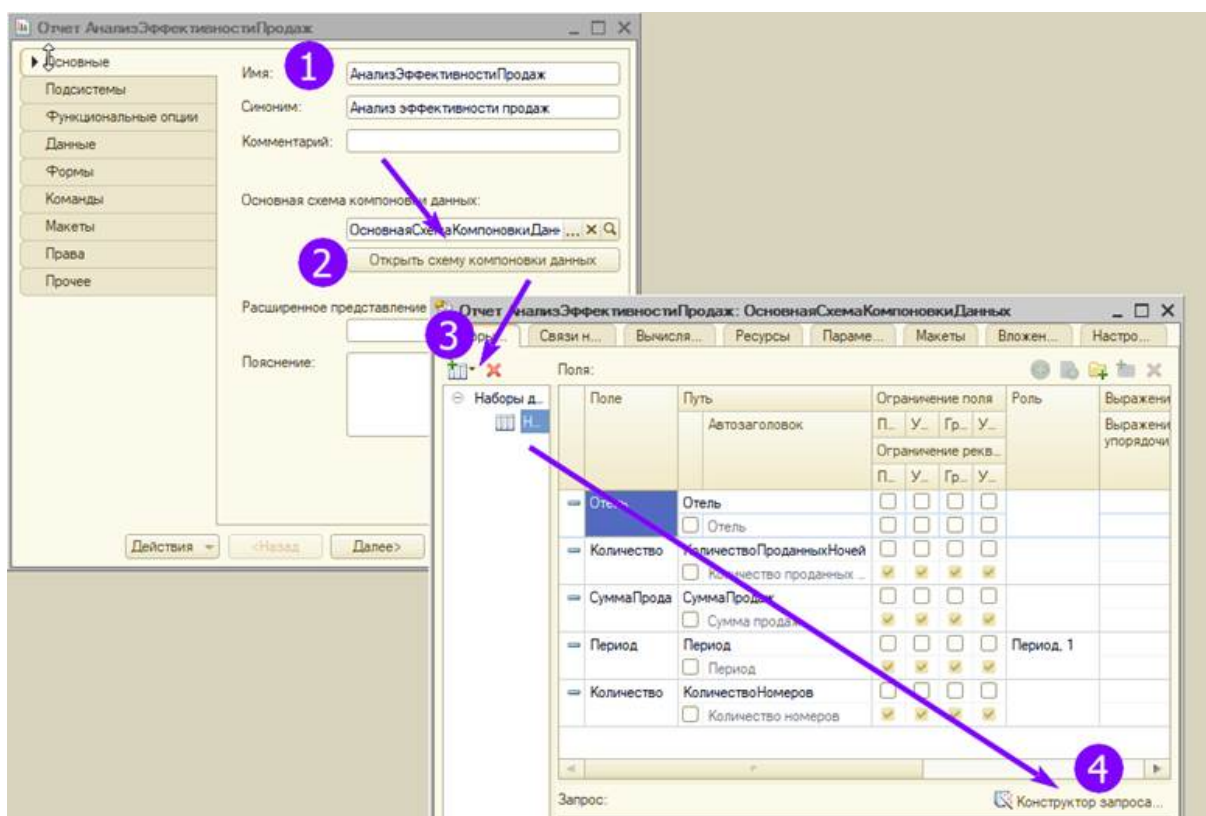
3. Далее необходимо задать правило записи данных из документов в регистр накопления. Для этого нужно открыть карточку документа и перейти на закладку «Движения». Можно не писать вручную алгоритм, а воспользоваться конструктором движений: здесь можно воспользоваться кнопкой «Заполнить выражения», тогда часть полей заполнится автоматически, а часть выражений придется заполнять автоматически.



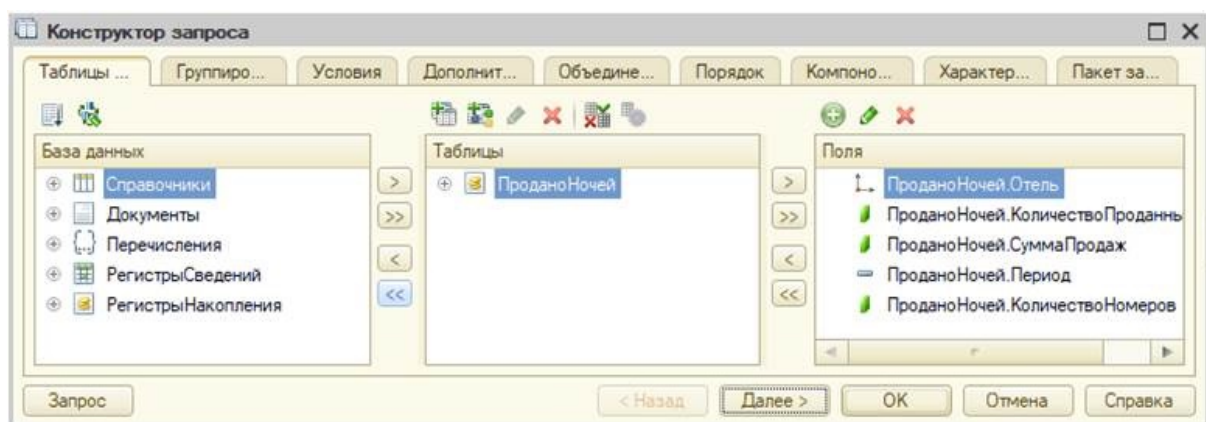
После того как мы нажмем на кнопку «ОК», система сама запишет алгоритм записи данных в регистр накопления.



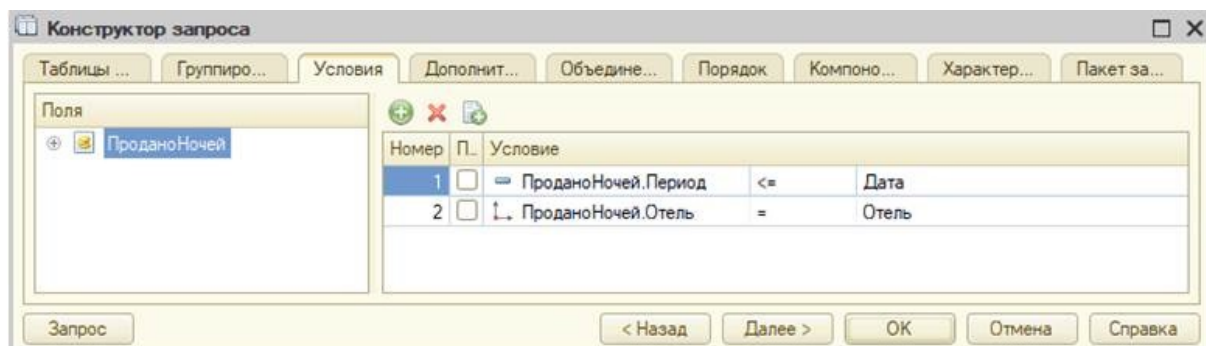
4. Далее необходимо создать отчет «Анализ эффективности продаж».



5. В качестве источника данных выбираем «Регистр накопления».



6. На закладке условия необходимо внести следующие условия.



7. После формирования запроса возвращаемся в конструктор схемы компоновки данных. Часть значений, которые должны попасть в отчет, должны быть

вычислены по формулам, приведенным в ТЗ. Формулы расчета необходимо записать на закладке «Вычисляемые поля».

Путь к данным	Выражение	Заголовок	Ограничение доступности				Выражение представления	Выражения упорядочивания	Тип значения
			Поле	Условие	Группа	Упоряд.			
КоличествоПроданныхНочей	КоличествоПроданныхНочей(О)	Количество проданных ночей	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			Число
СуммаПродаж	ОбщаяСуммаПродаж(Отели)	Сумма продаж	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			Число
ADR	СуммаПродаж / КоличествоПроданныхНочей	ADR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			Число
ПроцентЗагрузки	КоличествоПроданныхНочей / КоличествоНомеров	% загрузки номерного фонда	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			Число
RevPAR	СуммаПродаж / КоличествоНомеров	RevPAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			Число

8. На закладке «Параметры» указать параметры отчета — данные меняются в зависимости от даты и отеля.

Имя	Заголовок	Тип	Доступные значения	Дос...	Значение	Выражение
Дата	Дата	Дата		<input type="checkbox"/>		
Отель	Отель	СправочникСсылка.Отели		<input type="checkbox"/>		

9. На закладке «Настройки» необходимо будет настроить вывод данных для пользователя.

Настройки: Отчет

Пара... Выбр... Отбор Сорти... Услов... Поль... Други...

Отображать недоступные параметры (устанавливать значения недоступных п...

Параметр	Значение	Дата
<input type="checkbox"/> Дата	Произвольная д...	
<input type="checkbox"/> Отель		

Пример готового отчета.

← → ☆ Анализ эффективности продаж

Сформировать Выбрать вариант... Настройки...

Дата: Начало этого дня Отель: Небо

Параметры: Дата: 07.04.2023 0:00:00
Отель: Небо

Отель	Количество проданных ночей	Сумма продаж	% загрузки номерного фонда	ADR	RevPAR
Небо	2	1 000	0,2	500	100

10. Также в этой сессии необходимо ввести тестовые данные и поработать над интерфейсом.

Автоматизация деятельности хозяйственных служб

В рамках заключительной сессии предлагается автоматизировать деятельность хозяйственной службы — создать механизм формирования заявок на уборку номеров для горничных отеля.

Длительность сессии: 2 академических часа.

Техническое задание

- Создать структуру хранения информации о сотрудниках отеля в виде справочника «Сотрудники».
 - Все сотрудники разделяются по категориям. Категории — это отдельный справочник — полем «Наименование», тип «Строка».
 - Структура справочника «Сотрудники» будет иметь следующий вид.

ФИО сотрудника (тип данных — «Строка»)

Вид работ (тип данных — ссылка на справочник «Виды работ»)

- Создать структуру для создания и хранения заявок на уборку. Обратите внимание, что заявка на уборку, уборка — это события, которые происходят в определенную дату. В «1С:Предприятие 8» для отражения событий существует объект под названием «Документы». Необходимо создать документ «Работа», который будет иметь следующую структуру.

Дата (тип «Дата»)

ФИО горничной (ссылка на справочник «Сотрудники»)

Отель (тип — ссылка на справочник «Отели»)

Номер к уборке (тип — ссылка на объект хранения информации на номерах)

Статус (список значений, не редактируемый пользователем):

- К выполнению.
- На выполнении.
- Выполнена.

- В поле «Номер к уборке» подставляются только номера со статусом «Занят (грязный)» или «Свободен (грязный)».

3. Составить отчет, в котором выводится информация о том, сколько каждой горничной назначено заданий (к выполнению), сколько на выполнении, сколько выполнено. Отчет будет иметь следующий вид.

Анализ занятости горничных

ФИО горничной	К уборке	На уборке	Убрано

4. Создать удобный интерфейс для формирования заявок на уборку — некий интерфейс, рабочий стол, куда будут попадать номера со статусами статусом «Занят (грязный)» или «Свободен (грязный)», и там будет удобно создать заявку на уборку. Например, интерфейс может иметь такой вид.

К уборке

Отель _____

**Номер к
уборке**

102

Создать заявку на уборку

106

Создать заявку на уборку

301

Создать заявку на уборку

512

Создать заявку на уборку

- При нажатии на кнопку «Создать заявку на уборку» создается документ «Работа» с заполненными полями «Отель» и «Номер».

Тестирование

1. Завести не менее трех сотрудников с категорией «Горничная».
2. Создать документы «Работа» с различными статусами по каждой горничной.
3. Сформировать отчет и проверить правильность заполнения данных.

Критерии оценивания

Максимальное количество баллов за задачу — 100 баллов.

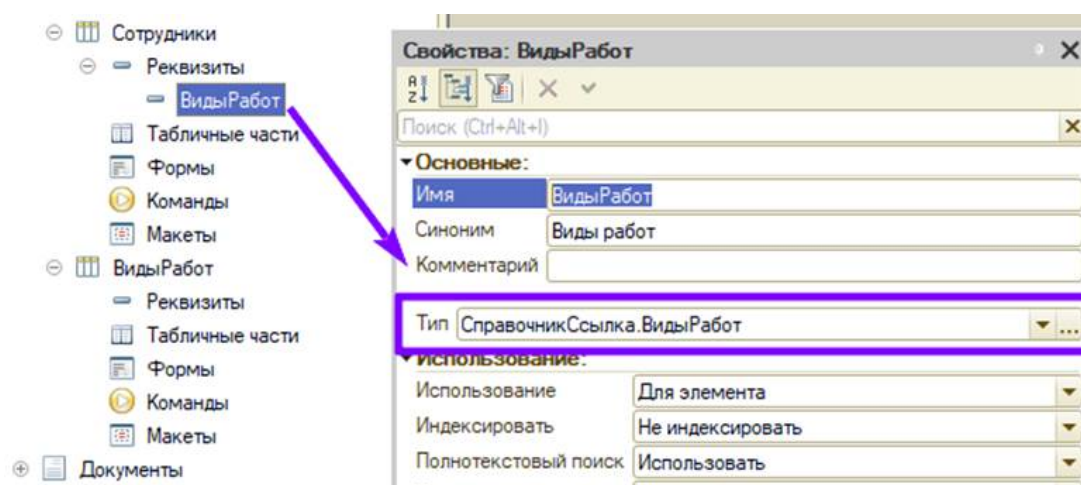
Штраф за отсутствие пояснительной записки — 1 балл.

Критерии	Базовый балл	Прибавка	Макс. балл
Созданная таблица (справочник) «Сотрудники»	5		5
Реквизит таблицы (справочника) «ФИО» — строка до 150 символов, хранит наименование отеля	2		2
Реквизит таблицы (справочника) «Вид работ»	2	Если реализована ссылка на справочник «Виды работ» +3	5
Созданная таблица (документ) для хранения информации о заявках на уборку	5		5
Реквизит «ФИО горничной»	2	Если реализована ссылка на справочник «Сотрудники» +2	4
Реквизит «Отель»	2	Если реализована ссылка на справочник «Отели» +2	4
Реквизит «Номер к уборке»	2	Доступны для выбора только номера со статусом «Занят (грязный)» и «Свободен (грязный)» +8	10
Реквизит «Статус заявки»	2	Заполненные статусы: <ul style="list-style-type: none"> • К выполнению. • На выполнении. • Выполнена. +3	5
Созданный отчет «Анализ занятости горничных»		Есть попытка создания отчета, но с ошибками +7 Отчет работает корректно +15	15

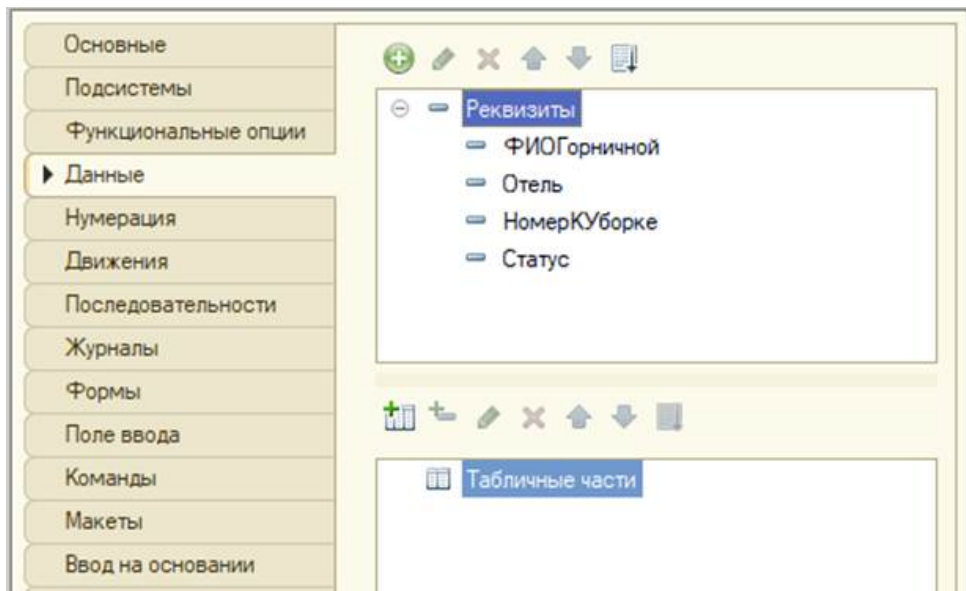
Критерии	Базовый балл	Прибавка	Макс. балл
Создать удобный интерфейс для формирования заявок на уборку – некий интерфейс, рабочий стол, куда будут попадать номера со статусами статусом «Занят (грязный)» или «Свободен (грязный)», и там будет удобно создать заявку на уборку		Есть попытка создания механизма +9 Механизм работает корректно +20	20
Внешний вид (интерфейс) на взгляд проверяющего: расположение объекта, заголовок приложения, какие-либо вставки картинок, форма и т. д.		до +15, если есть улучшения в интерфейсе, видна работа над ним. На усмотрение проверяющего	15

Основные этапы решения

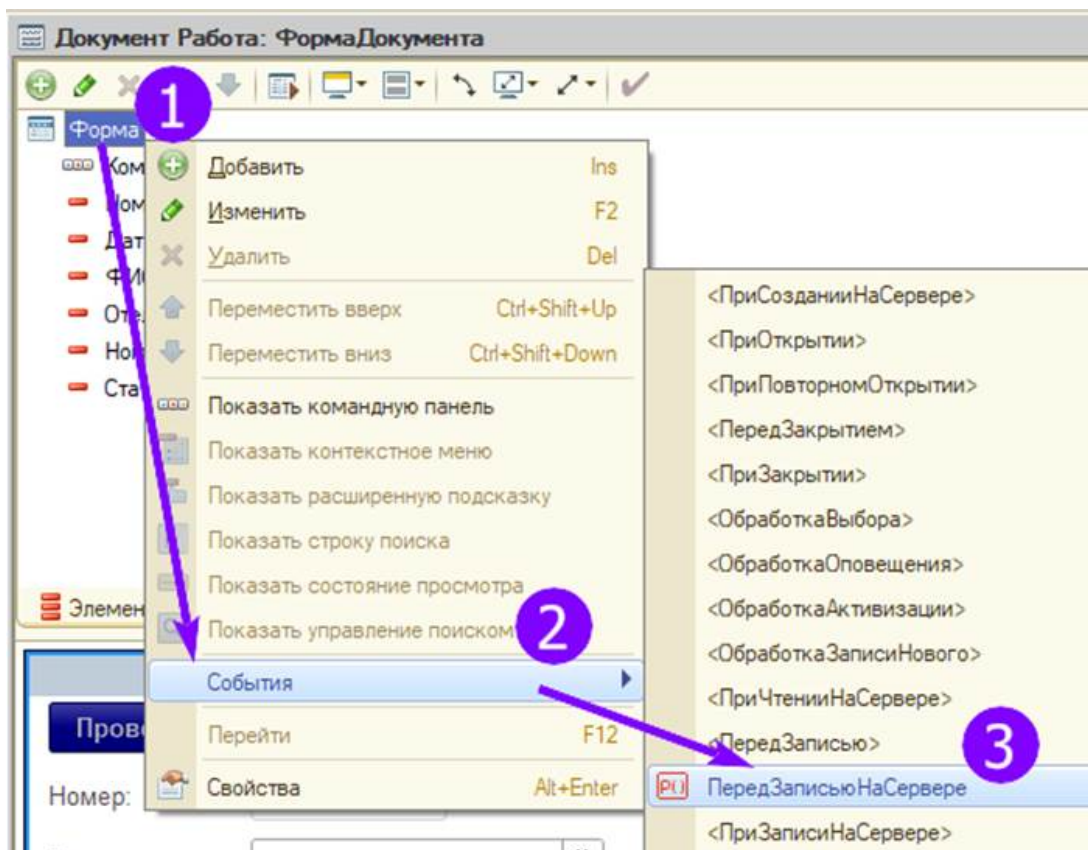
1. Сначала необходимо создать справочники «Сотрудники» и «Виды работ». Между собой эти справочники связаны по ключевому полю «Вид работ» в справочнике «Сотрудники».



2. Далее необходимо создать документ «Заявка на уборку» или «Работа», или можно назвать как вам нравится со следующими реквизитами.



3. В документе необходимо реализовать механизм, чтобы для уборки подставлялись номера со статусами «Свободен_грязный» и «Занят_грязный». Реализовать этот механизм можно также через форму документа и ее модуль.



Один из вариантов программного кода.

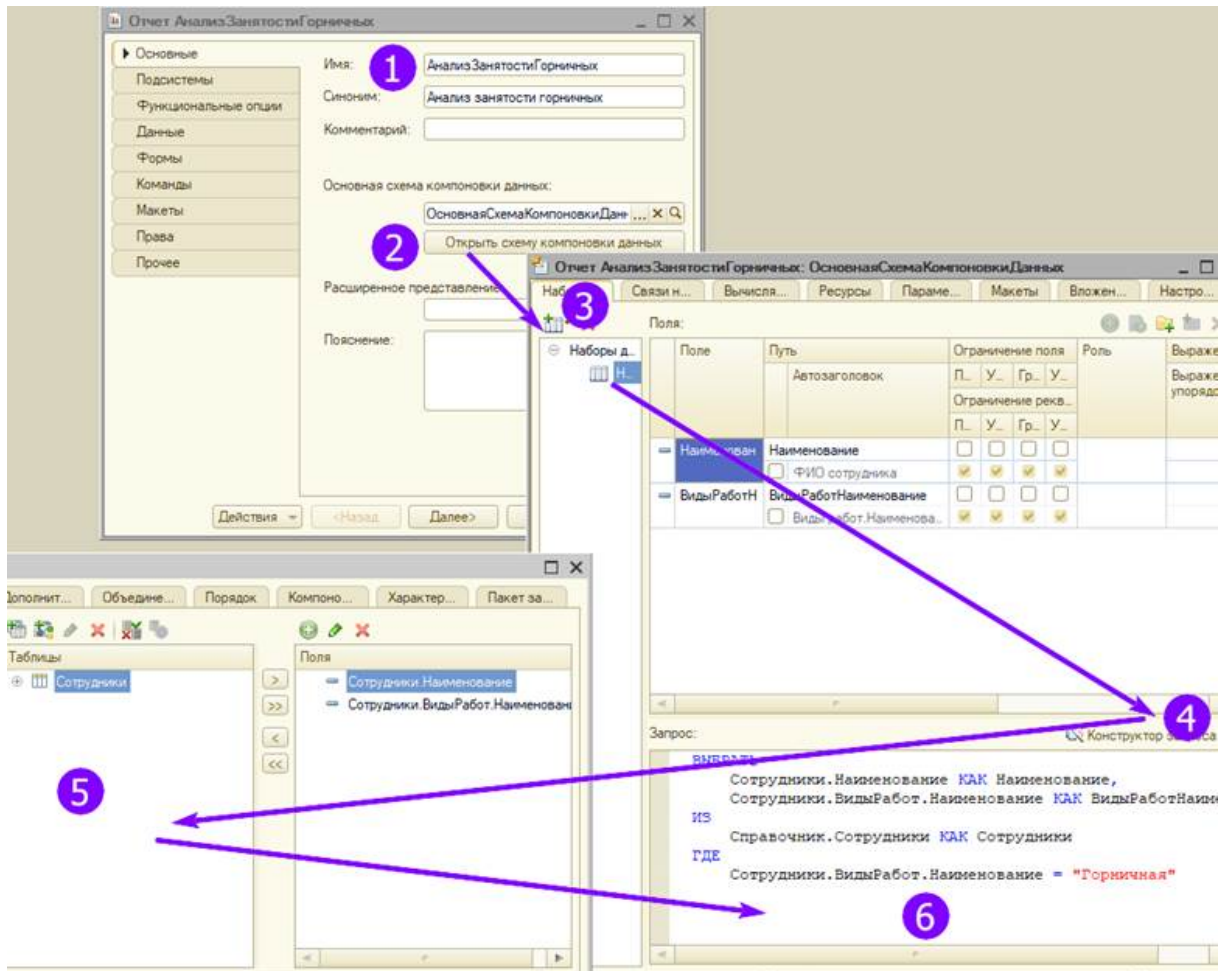
- 1 &НаСервере
- 2 Процедура ПередЗаписьюНаСервере(Отказ, ТекущийОбъект, ПараметрыЗаписи)
- 3 Если Объект.Статус=Перечисления.СтатусЗадания.Выполнена Тогда
- 4 Ном=Объект.НомерКУборке.ПолучитьОбъект();
- 5 Если Ном.Статус=Перечисления.Статус.Занят_Грязный Тогда
- 6 Ном.Статус=Перечисления.Статус.Занят;
- 7 Ном.Записать();

```

8         ИначеЕсли Ном.Статус=Перечисления.Статус.Свободен_Грязный Тогда
9             Ном.Статус=Перечисления.Статус.Свободен_Чистый;
10            Ном.Записать ();
11        КонецЕсли;
12    КонецЕсли;
13 КонецПроцедуры

```

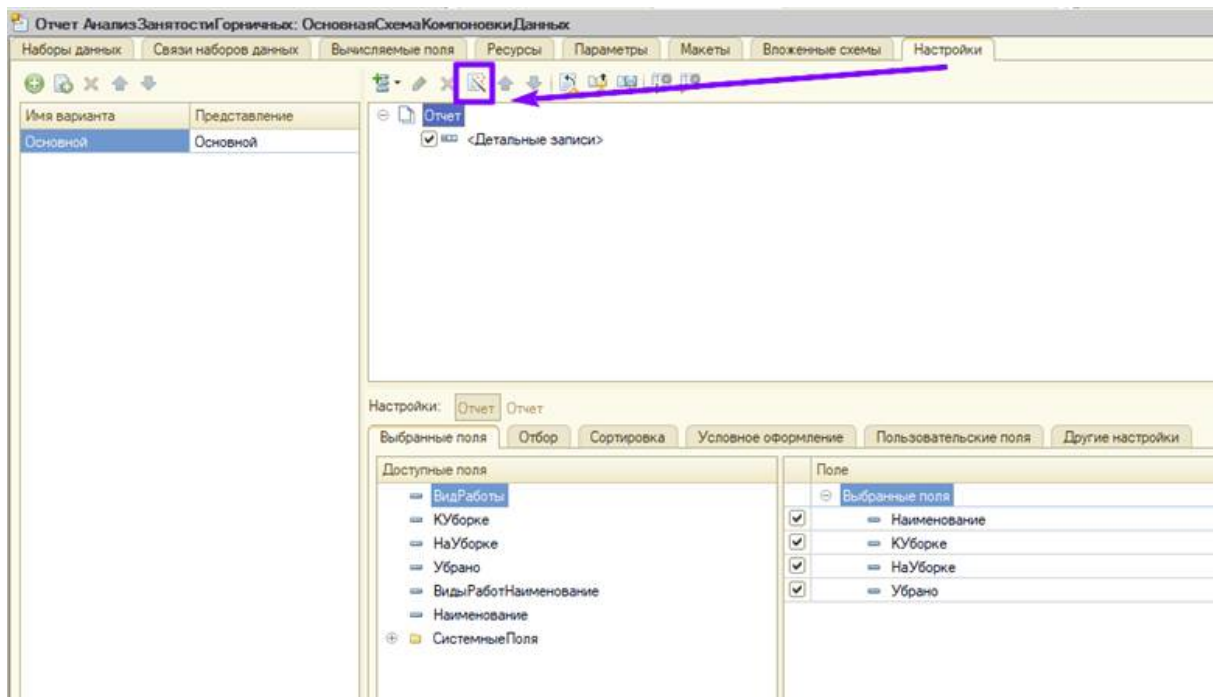
4. Далее необходимо составить отчет о занятости горничных. По традиции начинаем с формирования источника данных при помощи «Конструктора запроса».



5. Далее настраиваем поля в «Конструкторе схемы компоновки данных».

Путь к данным	Выражение	Заголовок	Ограничение доступности				Выражение представлено
			Поле	Усло_	Груп_	Упо_	
Куборке	Куборке(Наименование)	Куборке	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
НаУборке	НаУборке(Наименование)	На уборке	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Убрано	Убрано(Наименование)	Убрано	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ВидРаботы	Справочники.ВидыРабот.НайтиПоНаименованию("Горничная")	Вид работы	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

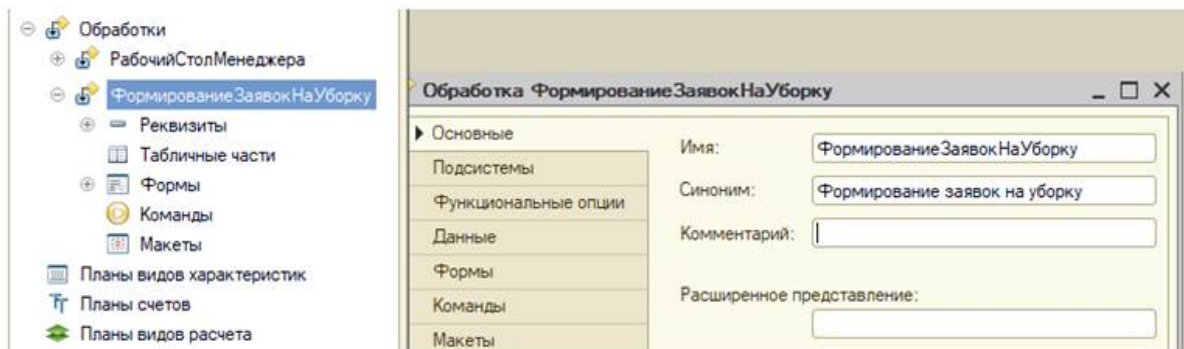
6. На закладке «Настройки» надо настроить отображение для пользователя.



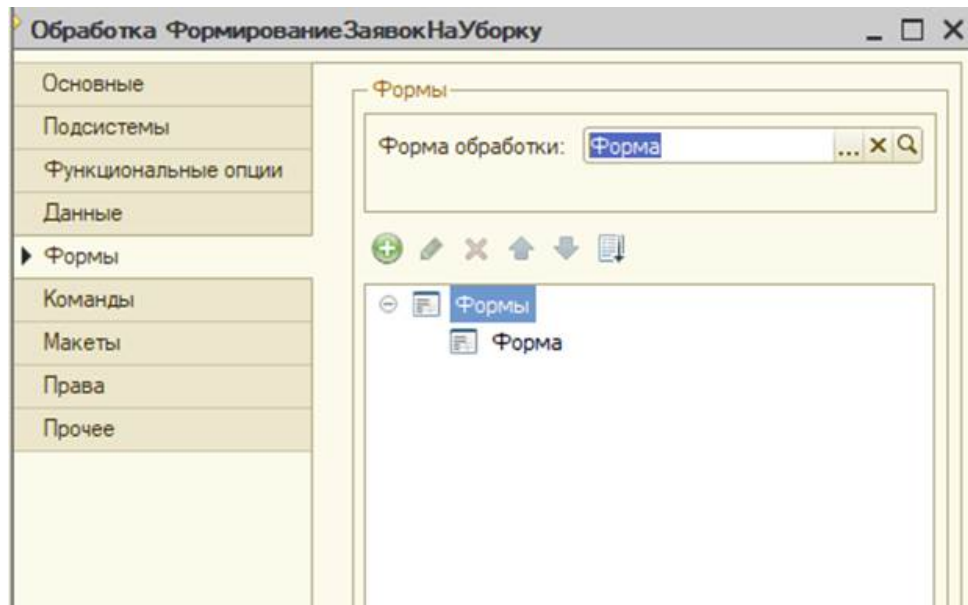
Пример созданного отчета.

ФИО сотрудника	К уборке	На уборке	Убрано
Клевцова Анастасия Семеновна	2	1	1
Дружбин Семен Иванович	1		
Сергеева Мария Ивановна	1	1	

7. В заключение предлагается создать удобный интерфейс для оформления заявок на уборку – некий интерфейс, рабочий стол, куда будут попадать номера со статусами статусом «Занят (грязный)» или «Свободен (грязный)», и там будет удобно создать заявку на уборку. Сделать такой интерфейс можно при помощи объекта «Обработка».



8. Далее необходимо создать форму этой обработки.



9. По кнопке «Лупа» необходимо открыть форму, разработать интерфейс, записать программный код для событий формы. Пример интерфейса.

Формирование заявок на уборку

Отель:

Номер	Отель	Категория	Количество мест	Статус
J402	Жемчужина	Стандарт	2	Свободен(грязный)
H201	Небо	Апартамент	3	Занят(грязный)
H101	Небо	Апартамент	3	Свободен(грязный)

Пример программного кода — как один из вариантов реализации данного механизма.

```

1  &НаСервере
2  Процедура ПриОткрытииНаСервере ()
3      ЭтаФорма.Реквизит1.Параметры.УстановитьЗначениеПараметра("Отель",
4      ↪ Справочники.Отели.НайтиПоНаименованию("Радость"));
5  КонецПроцедуры
6  &НаКлиенте
7  Процедура ПриОткрытии(Отказ)
8      ПриОткрытииНаСервере();
9  КонецПроцедуры
10
11 &НаСервере
12 Функция СоздатьЗаявкуНаУборкуНаСервере()
13     Возврат Справочники.Номера.ПустаяСсылка();
14 КонецФункции
15
16 &НаКлиенте
17 Процедура СоздатьЗаявкуНаУборку(Команда)
18     Если Элементы.Реквизит1.Видимость Тогда
19         Если Элементы.Реквизит1.ТекущиеДанные.Ссылка <>
20             ↪ СоздатьЗаявкуНаУборкуНаСервере() Тогда

```

```

20         Форма = ПолучитьФорму("Документ.Работа.Форма.ФормаДокумента");
21         Форма.СсылкаНаОбъект = Элементы.Реквизит1.ТекущиеДанные.Ссылка;
22         Форма.ОткрытьМодально();
23     КонечЕсли;
24     Иначе
25     Если Элементы.Реквизит2.ТекущиеДанные.Ссылка <>
26     ↪ СоздатьЗаявкуНаУборкуНаСервере() Тогда
27         Форма = ПолучитьФорму("Документ.Работа.Форма.ФормаДокумента");
28         Форма.СсылкаНаОбъект = Элементы.Реквизит2.ТекущиеДанные.Ссылка;
29         Форма.ОткрытьМодально();
30     КонечЕсли;
31 КонечПроцедуры
32
33 &НаСервере
34 Процедура ОтельПриИзмененииНаСервере()
35     Если Объект.Отель = Справочники.Отели.ПустаяСсылка() Тогда
36         Элементы.Реквизит1.Видимость = Ложь;
37         Элементы.Реквизит2.Видимость = Истина;
38     Иначе
39         Элементы.Реквизит1.Видимость = Истина;
40         Элементы.Реквизит2.Видимость = Ложь;
41     ЭтаФорма.Реквизит1.Параметры.УстановитьЗначениеПараметра("Отель",
42     ↪ Объект.Отель);
43 КонечЕсли;
44 КонечПроцедуры
45
46 &НаКлиенте
47 Процедура ОтельПриИзменении(Элемент)
48     ОтельПриИзмененииНаСервере();
49 КонечПроцедуры

```

10. В этой сессии, как и во всех остальных, необходимо ввести тестовые данные согласно ТЗ и привести интерфейс своего приложения к итоговому виду.