**Цель: На основании представленной дополнительной информации о предметной области и проектной специфики, произвести сопровождение имеющейся структуры базы данных.**

**Приобретаемые навыки:**

* *Умение сопровождать таблицы и столбцы;*
* *Умение сопровождать хранимые процедуры;*
* *Умение реализовывать алгоритмы в хранимых процедурах;*
* *Умение работать с переменными;*
* *Умение работать с условными операторами;*
* *Умение работать с циклами;*
* *Умение работать с перехватчиками ошибок;*
* *Умение разграничивать права доступа;*
* *Умение создавать резервную копию базы данных;*
* *Умение определять версионность реализованной базы данных.*

**Этапы выполнения работы:**

1. Сопровождение таблиц:
   1. Создание файла;



* 1. Сопровождение объектов.

Таблица 1 – Сопровождение таблиц баз данных

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Логическое название | Физическое название | Тип данных | Обязательное | Ограничения | Было | Стало | Результат |
| Класс (Class) | ID\_Class | Int | Да | Суррогатный ключ | -- Таблица Класс (Class)  CREATE TABLE Class (  ID\_Class SERIAL NOT NULL CONSTRAINT PK\_Class PRIMARY KEY,  Name\_Class VARCHAR(12) NOT NULL,  Image\_Class VARCHAR(100) NOT NULL  ); |  |  |
| Name\_Class | Varchar(12) | Да |  |
| Image\_Class | Varchar(100) | Да |  |
| Статус номера (Status\_room) | ID\_Status\_room | Int | Да | Суррогатный ключ | -- Таблица Статус номера (Status\_room)  CREATE TABLE Status\_Room (  ID\_Status\_Room SERIAL NOT NULL CONSTRAINT PK\_Status\_Room PRIMARY KEY,  Name\_Status\_room VARCHAR(12) NOT NULL  ); | -- Добавление ограничения UNIQUE на столбец Name\_Status\_room  ALTER TABLE Status\_Room  ADD CONSTRAINT UQ\_Name\_Status\_room UNIQUE (Name\_Status\_room),  ADD CONSTRAINT CH\_Name\_Status\_room CHECK (Name\_Status\_room IN ('Свободен', 'Забронирован', 'Занят', 'Недоступен')); |  |
| Name\_Status\_room | Varchar(12) | Да | Уникальный,  “Свободен”, “Забронирован”, “Занят”, “Недоступен” |
| Номер (Room) | ID\_Room | Int | Да | Суррогатный ключ | -- Таблица Номер (Room)  CREATE TABLE Room (  ID\_Room SERIAL NOT NULL CONSTRAINT PK\_Room PRIMARY KEY,  Number\_Room VARCHAR(3) NOT NULL,  Daily\_Cost\_Room INT NOT NULL,  Class\_ID INT NOT NULL REFERENCES Class(ID\_Class),  Status\_Room\_ID INT NOT NULL REFERENCES Status\_Room(ID\_Status\_Room)  ); | ALTER TABLE Room  ADD CONSTRAINT unique\_room\_number\_room UNIQUE (Room\_Number\_Room),  ADD CONSTRAINT chk\_room\_number\_room CHECK (Room\_Number\_Room >= 0),  ADD CONSTRAINT chk\_daily\_cost\_room CHECK (Daily\_Cost\_Room >= 0); |  |
| Room\_Number\_Room | Int | Да | Уникальное, >=0 |
| Class\_ID | Int | Да | Внешний ключ |
| Daily\_Cost\_Room | Int | Да | >=0 |
| Status\_Room\_ID | Int | Да | Внешний ключ |
| Комплектация (Configuration) | ID\_Configuration | Int | Да | Суррогатный ключ | -- Таблица Комплектация (Configuration)  CREATE TABLE Configuration (  ID\_Configuration SERIAL NOT NULL CONSTRAINT PK\_Configuration PRIMARY KEY,  Furniture\_Configuration VARCHAR(14) NOT NULL  ); |  |  |
| Furniture\_Configuration | Varchar(14) | Да |  |
| Комплектация номера (Room\_Configuration) | ID\_Room\_Configuration | Int | Да | Суррогатный ключ | -- Таблица Комплектация номера (Room\_Configuration)  CREATE TABLE Room\_Configuration (  ID\_Room\_Configuration SERIAL NOT NULL CONSTRAINT PK\_Room\_Configuration PRIMARY KEY,  Quantity\_Room\_Configuration INT NOT NULL,  Configuration\_ID INT NOT NULL REFERENCES Configuration(ID\_Configuration),  Room\_ID INT NOT NULL REFERENCES Room(ID\_Room)  ); | ALTER TABLE Room\_Configuration  ADD CONSTRAINT chk\_quantity\_room\_configuration CHECK (Quantity\_Room\_Configuration >= 0); |  |
| Quantity\_Room\_Configuration | Int | Да | >=0 |
| Configuration \_ID | Int | Да | Внешний ключ |
| Room\_ID | Int | Да | Внешний ключ |
| Опции (Options) | ID\_Options | Int | Да | Суррогатный ключ | -- Таблица Опции (Options)  CREATE TABLE Options (  ID\_Options SERIAL NOT NULL CONSTRAINT PK\_Options PRIMARY KEY,  Name\_Options VARCHAR(13) NOT NULL  ); |  |  |
| Name\_Options | Varchar(13) | Да |  |
| Опции номера (Room\_Options) | ID\_Room\_Options | Int | Да | Суррогатный ключ | -- Таблица Опции номера (Room\_Options)  CREATE TABLE Room\_Options (  ID\_Room\_Options SERIAL NOT NULL CONSTRAINT PK\_Room\_Options PRIMARY KEY,  Quantity\_Room\_Options INT NOT NULL,  Option\_ID INT NOT NULL REFERENCES Options(ID\_Options),  Room\_ID INT NOT NULL REFERENCES Room(ID\_Room)  ); |  |  |
| Quantity\_Room\_Options | Int | Да |  |
| Option\_ID | Int | Да | Внешний ключ |
| Room\_ID | Int | Да | Внешний ключ |
| Бронирование (Reservation) | ID\_Reservation | Int | Да | Суррогатный ключ | -- Таблица Бронирование (Reservation)  CREATE TABLE Reservation (  ID\_Reservation SERIAL NOT NULL CONSTRAINT PK\_Reservation PRIMARY KEY,  Number\_Reservation VARCHAR(16) NOT NULL,  Creation\_Date\_Reservation DATE NOT NULL,  Checkin\_Date\_Reservation DATE NOT NULL,  Checkin\_Time\_Reservation TIME NOT NULL,  Checkout\_Date\_Reservation DATE NOT NULL,  Checkout\_Time\_Reservation TIME NOT NULL,  Total\_Cost\_Reservation INT NOT NULL,  Room\_ID INT NOT NULL REFERENCES Room(ID\_Room)  ); |  |  |
| Num\_Reservation | Varchar(16) | Да | БРН-[0-9]-{9}-[0-9] {2} |
| Creation\_Date\_Reservation | Date | Да | = Текущая дата |
| Checkin\_Date\_Reservation | Date | Да | > Дата создания |
| Checkin\_Time\_Reservation | Time | Да | >= 14:00 |
| Checkout\_Date\_Reservation | Date | Да | > Дата заселения |
| Checkout\_Time\_Reservation | Time | Да | <= 12:00 |
| Room\_ID | Int | Да | Внешний ключ |
| Total\_Cost\_Reservation | Int | Да | > 0 |
| Клиент (Client) | ID\_Client | Int | Да | Суррогатный ключ | -- Таблица Клиент (Client)  CREATE TABLE Client (  ID\_Client SERIAL NOT NULL CONSTRAINT PK\_Client PRIMARY KEY,  Surname\_Client VARCHAR(50) NOT NULL,  Name\_Client VARCHAR(50) NOT NULL,  Patronymic\_Client VARCHAR(50),  Document\_Series\_Client VARCHAR(5) NOT NULL,  Document\_Number\_Client INT NOT NULL,  Date\_of\_Birth\_Client DATE NOT NULL,  Gender\_Client VARCHAR(3) NOT NULL,  Issued\_By\_Client VARCHAR(70) NOT NULL,  Issued\_Date\_Client DATE NOT NULL,  Code\_Department\_Client VARCHAR(7) NOT NULL,  Phone\_Client VARCHAR(16) NOT NULL,  Email\_Address\_Client VARCHAR(70) NOT NULL,  Login\_Client VARCHAR(50) NOT NULL,  Password\_Client VARCHAR(50) NOT NULL  ); | -- Ограничение для таблицы Client  ALTER TABLE Client  ADD CONSTRAINT CK\_Document\_Series\_Client CHECK (Document\_Series\_Client ~ '^[0-9]{2} [0-9]{2}$'),  ADD CONSTRAINT CK\_Document\_Number\_Client CHECK (Document\_Number\_Client ~ '^[0-9]{6}$'),  ADD CONSTRAINT CK\_Date\_of\_Birth\_Client CHECK (EXTRACT(YEAR FROM AGE(Date\_of\_Birth\_Client)) >= 18),  ADD CONSTRAINT CK\_Issued\_Date\_Client CHECK (Issued\_Date\_Client >= Date\_of\_Birth\_Client + INTERVAL '14 years'),  ADD CONSTRAINT CK\_Code\_Department\_Client CHECK (Code\_Department\_Client ~ '^[0-9]{3}-[0-9]{3}$'),  ADD CONSTRAINT CK\_Phone\_Client CHECK (Phone\_Client ~ '^\+7\([0-9]{3}\)[0-9]{3}-[0-9]{2}-[0-9]{2}$'),  ADD CONSTRAINT CK\_Email\_Address\_Client CHECK (Email\_Address\_Client ~ '^[^@]+@[^@]+\.[^@]+$'),  ADD CONSTRAINT CK\_Login\_Client CHECK (LENGTH(Login\_Client) >= 4 AND Login\_Client ~ '^[a-zA-Z]+$'),  ADD CONSTRAINT CK\_Password\_Client CHECK (LENGTH(Password\_Client) >= 4 AND Password\_Client ~ '[a-zA-Z]' AND Password\_Client ~ '[0-9]' AND Password\_Client ~ '[!@#$%^&\*()]'); |  |
| Surname\_Client | Varchar(50) | Да |  |
| Name\_Client | Varchar(50) | Да |  |
| Patronymic\_Client | Varchar(50) | Нет | Default “-“ |
| Document\_Series\_Client | Varchar(5) | Да | [0-9]{2} [0-9]{2} |
| Document\_Number\_Client | Varchar(6) | Да | [0-9]{6} |
| Date\_of\_Birth\_Client | Date | Да | >= 18 лет |
| Gender\_Client | Varchar(3) | Да |  |
| Issued\_By\_Client | Varchar(70) | Да |  |
| Issued\_Date\_Client | Date | Да | >= 14 лет |
| Code\_Department\_Client | Varchar(7) | Да | [0-9]{3}-[0-9]{3} |
| Phone\_Client | Varchar(16) | Да | +7([0-9]{3})[0-9]{3}-  [0-9]{2}-[0-9]{2},  Уникальный |
| Email\_Address\_Client | Varchar(70) | Да | %@%.%  Уникальный |
| Login\_Client | Varchar(50) | Да | уникальные,  Length >= 4, [a-Z] |
| Password\_Client | Varchar(50) | Да | Length >= 4, [a-Z][0-9][!@#$%^&\*()] |
| Бронирование клиента (Client\_Reservation) | ID\_Client\_Reservation | Int | Да | Суррогатный ключ | -- Таблица Бронирование клиента (Client\_Reservation)  CREATE TABLE Client\_Reservation (  ID\_Client\_Reservation SERIAL NOT NULL CONSTRAINT PK\_Client\_Reservation PRIMARY KEY,  Client\_ID INT NOT NULL REFERENCES Client(ID\_Client),  Reservation\_ID INT NOT NULL REFERENCES Reservation(ID\_Reservation)  ); |  |  |
| Client\_ID | Int | Да | Внешний ключ |
| Reservation\_ID | Int | Да | Внешний ключ |
| Дополнительные услуги (Add\_Services) | ID\_Add\_Services | Int | Да | Суррогатный ключ | -- Таблица Дополнительные услуги (Add\_Services)  CREATE TABLE Add\_Services (  ID\_Add\_Services SERIAL NOT NULL CONSTRAINT PK\_Add\_Services PRIMARY KEY,  Name\_Add\_Services VARCHAR(8) NOT NULL,  Cost\_Add\_Services INT NOT NULL  ); |  |  |
| Name\_Add\_Services | Varchar(8) | Да |  |
| Cost\_Add\_Services | Int | Да |  |
| Дополнительные услуги бронирования (Add\_Services\_Reservation) | ID\_Add\_Services\_Reservation | Int | Да | Суррогатный ключ | -- Таблица Дополнительные услуги бронирования (Add\_Services\_Reservation)  CREATE TABLE Add\_Services\_Reservation (  ID\_Add\_Services\_Reservation SERIAL NOT NULL CONSTRAINT PK\_Add\_Services\_Reservation PRIMARY KEY,  Add\_Services\_ID INT NOT NULL REFERENCES Add\_Services(ID\_Add\_Services),  Reservation\_ID INT NOT NULL REFERENCES Reservation(ID\_Reservation)  ); |  |  |
| Add\_Services\_ID | Int | Да | Внешний ключ |
| Reservation\_ID | Int | Да | Внешний ключ |
| Сотрудник (Employee) | ID\_Employee | Int | Да | Суррогатный ключ | -- Таблица Сотрудник (Employee)  CREATE TABLE Employee (  ID\_Employee SERIAL NOT NULL CONSTRAINT PK\_Employee PRIMARY KEY,  Surname\_Employee VARCHAR(50) NOT NULL,  Name\_Employee VARCHAR(50) NOT NULL,  Patronymic\_Employee VARCHAR(50),  Login\_Employee VARCHAR(50) NOT NULL,  Password\_Employee VARCHAR(50) NOT NULL  ); | ALTER TABLE Employee  ADD CONSTRAINT CK\_Login\_Employee CHECK (LENGTH(Login\_Employee) >= 4 AND Login\_Employee ~ '^[a-zA-Z\_]+$'),  ADD CONSTRAINT CK\_Password\_Employee CHECK (LENGTH(Password\_Employee) >= 4 AND Password\_Employee ~ '[a-zA-Z]' AND Password\_Employee ~ '[0-9]' AND Password\_Employee ~ '[!@#$%^&\*()]'); |  |
| Surname\_Employee | Varchar(50) | Да |  |
| Name\_Employee | Varchar(50) | Да |  |
| Patronymic\_Employee | Varchar(50) | Нет | Default “-“ |
| Login\_Employee | Varchar(50) | Да | уникальные,  Length >= 4, [a-Z] |
| Password\_Employee | Varchar(50) | Да | Length >= 4, [a-Z][0-9][!@#$%^&\*()] |
| Договор (Contract) | ID\_Contract | Int | Да | Суррогатный ключ | -- Таблица Договор (Contract)  CREATE TABLE Contract (  ID\_Contract SERIAL NOT NULL CONSTRAINT PK\_Contract PRIMARY KEY,  Number\_Contract VARCHAR(12) NOT NULL,  Creation\_Date\_Contract DATE NOT NULL,  Creation\_Time\_Contract TIME NOT NULL,  Reservation\_ID INT NOT NULL REFERENCES Reservation(ID\_Reservation),  Employee\_ID INT NOT NULL REFERENCES Employee(ID\_Employee)  ); | ALTER TABLE Contract  ADD CONSTRAINT chk\_number\_contract CHECK (Number\_Contract ~ 'ДГН-[0-9]{8}'); |  |
| Number\_Contract | Varchar(12) | Да | ДГН-[0-9] {8} |
| Reservation\_ID | Int | Да |  |
| Employee\_ID | Int | Да |  |
| Creation\_Date\_Contract | Date | Да | > Дата создания брони, < Дата заселения |
| Creation\_Time\_Contract | Time | Да | < Время заселения |
| Статус чека (Status\_Receipt) | ID\_Status\_Receipt | int | Да | Суррогатный ключ |  | CREATE TABLE Status\_Receipt (  ID\_Status\_Receipt INT PRIMARY KEY,  Name\_Status\_Receipt VARCHAR(9) NOT NULL CHECK (Name\_Status\_Receipt IN ('Не оплачен', 'Оплачен'))  );  ALTER TABLE Status\_Receipt  ADD CONSTRAINT unique\_name\_status\_receipt UNIQUE (Name\_Status\_Receipt); |  |
| Name\_Status\_Receipt | varchar(9) | Да | Уникален,  ‘Не оплачен’, ‘Оплачен’ |
| Вид расчета (Type\_Pay) | ID\_Type\_Pay | int | Да | Суррогатный ключ |  | CREATE TABLE Type\_Pay (  ID\_Type\_Pay INT PRIMARY KEY,  Name\_Type\_Pay VARCHAR(11) NOT NULL CHECK (Name\_Type\_Pay IN ('Наличный', 'Безналичный'))  );  ALTER TABLE Type\_Pay  ADD CONSTRAINT unique\_name\_type\_pay UNIQUE (Name\_Type\_Pay); |  |
| Name\_Type\_Pay | varchar(11) | Да | Уникален,  ‘Наличный’, ‘Безналичный’; |
| Чек (Receipt) | ID\_Receipt | int | Да | Суррогатный ключ |  | CREATE TABLE Receipt (  ID\_Receipt INT PRIMARY KEY,  Num\_Receipt VARCHAR(19) NOT NULL CHECK (Num\_Receipt ~ 'Кч/[0-9]{2}/Ус-[0-9]{10}'),  Date\_Create\_Receipt DATE NOT NULL DEFAULT CURRENT\_DATE,  Time\_Create\_Receipt TIME NOT NULL DEFAULT CURRENT\_TIME,  Status\_Receipt\_ID INT NOT NULL,  Total\_Cost\_Receipt INT NOT NULL CHECK (Total\_Cost\_Receipt > 0),  Type\_Pay\_ID INT NOT NULL,  Entered\_Sum\_Receipt INT NOT NULL CHECK (Entered\_Sum\_Receipt >= 0),  Change\_Sum\_Receipt INT NOT NULL CHECK (Change\_Sum\_Receipt >= 0)  );  ALTER TABLE Receipt  ADD CONSTRAINT unique\_num\_receipt UNIQUE (Num\_Receipt); |  |
| Num\_Receipt | varchar(19) | Да | Уникален,  Кч/[0-9]{2}/Ус-[0-9]{10} |
| Date\_Create\_Receipt | date | Да | = Наст. время |
| Time\_Create\_Receipt | time | Да | = Наст. время |
| Status\_Receipt\_ID | int | Да | Внешний ключ |
| Total\_Cost\_Receipt | int | Да | > 0 |
| Type\_Pay\_ID | int | Да | Внешний ключ |
| Entered\_Sum\_Receipt | int | Да | >= 0 |
| Change\_Sum\_Receipt | int | Да | >= 0 |
| Заказы клиентов (Client\_Orders) | ID\_Client\_Orders | int | Да | Суррогатный ключ |  | CREATE TABLE Client\_Orders (  ID\_Client\_Orders INT PRIMARY KEY,  Contract\_ID INT NOT NULL,  Receipt\_ID INT NOT NULL  ); |  |
| Contract\_ID | int | Да | Внешний ключ |
| Receipt\_ID | int | Да | Внешний ключ |
| Виды услуг (Type\_Service) | ID\_Type\_Service | int | Да | Суррогатный ключ |  | CREATE TABLE Type\_Service (  ID\_Type\_Service INT PRIMARY KEY,  Name\_Type\_Service VARCHAR(100) NOT NULL  );  ALTER TABLE Type\_Service  ADD CONSTRAINT unique\_name\_type\_service UNIQUE (Name\_Type\_Service); |  |
| Name\_Type\_Service | varchar(100) | Да | Уникален |
| Услуга (Service) | ID\_Service | int | Да | Суррогатный ключ |  | CREATE TABLE Service (  ID\_Service INT PRIMARY KEY,  Type\_Service\_ID INT NOT NULL,  Quantity\_Service INT NOT NULL CHECK (Quantity\_Service >= 0),  Cost\_Service INT NOT NULL CHECK (Cost\_Service > 0)  ); |  |
| Type\_Service\_ID | int | Да | Внешний ключ |
| Quantity\_Service | int | Да | >= 0 |
| Cost\_Service | int | Да | > 0 |
| Услуга чека (Service\_Receipt) | ID\_Service\_Receipt | int | Да | Суррогатный ключ |  | CREATE TABLE Service\_Receipt (  ID\_Service\_Receipt INT PRIMARY KEY,  Service\_ID INT NOT NULL,  Receipt\_ID INT NOT NULL  ); |  |
| Service\_ID | int | Да | Внешний ключ |
| Receipt\_ID | int | Да | Внешний ключ |