|  |
| --- |
| Министерство науки и высшего образования Российской Федерации |
| Федеральное государственное бюджетное учреждение  высшего образования |
| **МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ** |
| **(МОСКОВСКИЙ ПОЛИТЕХ)** |
| Факультет информационных технологий |

Кафедра «Прикладная информатика»

Форма обучения: очная

**Лабораторная работа №1**

«Тема: **Data Vault**»

**по дисциплине**

**«Хранилище данных»**

**Вариант №6**

|  |  |  |
| --- | --- | --- |
| Студент | (личная подпись) | Жабина П.А.  (И.О. Фамилия) |

Москва 2021

СОДЕРЖАНИЕ

[1 Тема. Актуальность темы. 3](#_Toc88210024)

[2 База данных. Модель 3NF 3](#_Toc88210025)

[3 База данных. Data Vault 5](#_Toc88210026)

# Тема. Актуальность темы.

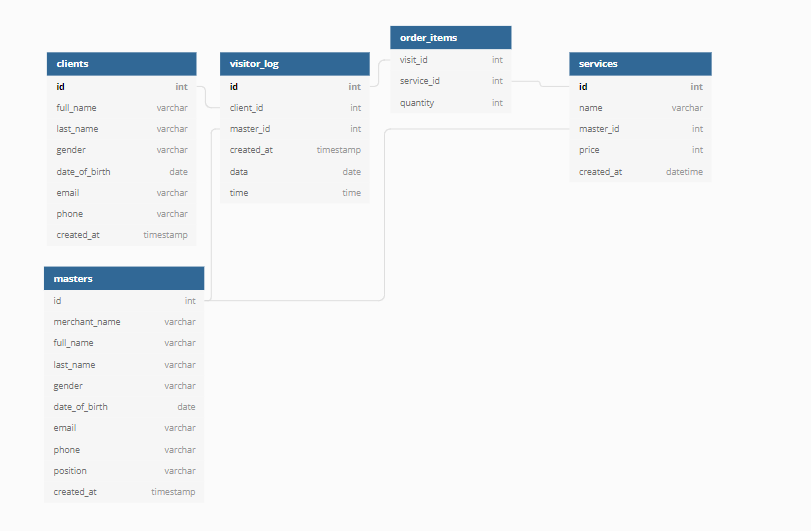
Тема лабораторной работы. Парикмахерская.

Автоматизация деятельности небольших предприятий представляет собой задачу сложного выбора между функциональностью системы и ее стоимостью на рынке программного обеспечения. Существующие решения для небольших компаний со специфичными видами деятельности, например, предоставление парикмахерских и косметических услуг населению, по стоимости и функционалу не всегда устраивают владельцев.

Актуальность исследования состоит в необходимости автоматизации деятельности салона красоты, связанной с обслуживанием посетителей, расходами препаратов и учетом деятельности мастеров, оплата которых зависит от объемов проделанной работы. Мастера выполняют работу по сменно, и администратор, который курирует работу мастеров должен выходить в несколько смен, или управлять работой удаленно. Автоматизация процессов расчета с посетителями, определения объемов проведенных работ позволит администратору исполнять свои обязанности в рамках рабочего времени.

# База данных. Модель 3NF

Ниже представлена er-диаграмма и sql код



CREATE TYPE "products\_status" AS ENUM (

  'out\_of\_stock',

  'in\_stock',

  'running\_low'

);

CREATE TABLE "clients" (

  "id" SERIAL PRIMARY KEY,

  "full\_name" varchar,

  "last\_name" varchar,

  "gender" varchar,

  "date\_of\_birth" date,

  "email" varchar,

  "phone" varchar,

  "created\_at" timestamp

);

CREATE TABLE "visitor\_log" (

  "id" int PRIMARY KEY,

  "client\_id" int,

  "master\_id" int,

  "created\_at" timestamp,

  "data" date,

  "time" time

);

CREATE TABLE "order\_items" (

  "visit\_id" int,

  "service\_id" int,

  "quantity" int DEFAULT 1

);

CREATE TABLE "services" (

  "id" int PRIMARY KEY,

  "name" varchar,

  "master\_id" int,

  "price" int,

  "created\_at" datetime DEFAULT (now())

);

CREATE TABLE "masters" (

  "id" int,

  "merchant\_name" varchar,

  "full\_name" varchar,

  "last\_name" varchar,

  "gender" varchar,

  "date\_of\_birth" date,

  "email" varchar,

  "phone" varchar,

  "position" varchar,

  "created\_at" timestamp

);

ALTER TABLE "visitor\_log" ADD FOREIGN KEY ("client\_id") REFERENCES "clients" ("id");

ALTER TABLE "visitor\_log" ADD FOREIGN KEY ("master\_id") REFERENCES "masters" ("id");

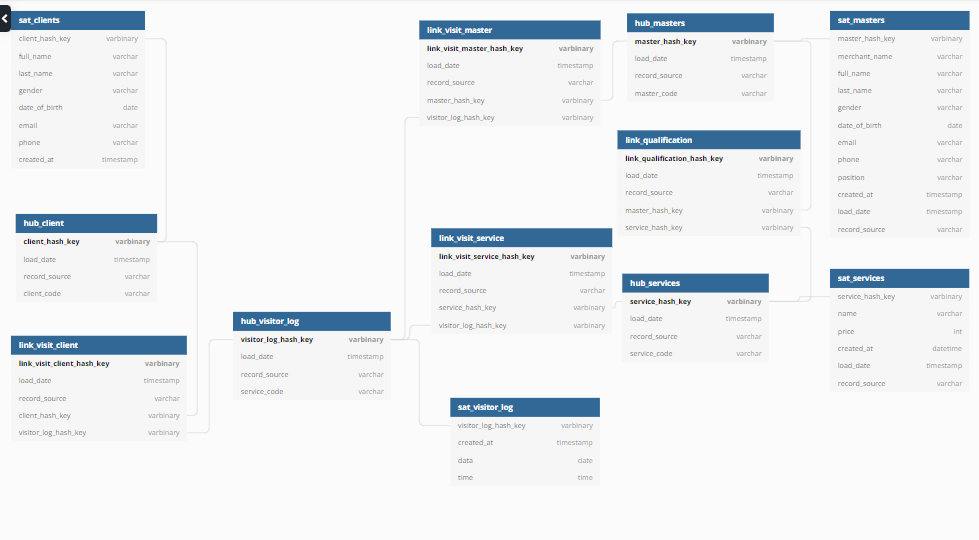
ALTER TABLE "order\_items" ADD FOREIGN KEY ("visit\_id") REFERENCES "visitor\_log" ("id");

ALTER TABLE "order\_items" ADD FOREIGN KEY ("service\_id") REFERENCES "services" ("id");

ALTER TABLE "services" ADD FOREIGN KEY ("master\_id") REFERENCES "masters" ("id");

# База данных. Data Vault

Далее исходную 3ю нормальную форму необходимо было доработать до Data vault.



CREATE TABLE "sat\_clients" (

  "client\_hash\_key" varbinary,

  "full\_name" varchar,

  "last\_name" varchar,

  "gender" varchar,

  "date\_of\_birth" date,

  "email" varchar,

  "phone" varchar,

  "created\_at" timestamp

);

CREATE TABLE "hub\_client" (

  "client\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "client\_code" varchar

);

CREATE TABLE "link\_visit\_client" (

  "link\_visit\_client\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "client\_hash\_key" varbinary,

  "visitor\_log\_hash\_key" varbinary

);

CREATE TABLE "sat\_visitor\_log" (

  "visitor\_log\_hash\_key" varbinary,

  "created\_at" timestamp,

  "data" date,

  "time" time

);

CREATE TABLE "hub\_visitor\_log" (

  "visitor\_log\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "service\_code" varchar

);

CREATE TABLE "sat\_services" (

  "service\_hash\_key" varbinary,

  "name" varchar,

  "price" int,

  "created\_at" datetime DEFAULT (now()),

  "load\_date" timestamp,

  "record\_source" varchar

);

CREATE TABLE "hub\_services" (

  "service\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "service\_code" varchar

);

CREATE TABLE "link\_visit\_service" (

  "link\_visit\_service\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "service\_hash\_key" varbinary,

  "visitor\_log\_hash\_key" varbinary

);

CREATE TABLE "sat\_masters" (

  "master\_hash\_key" varbinary,

  "merchant\_name" varchar,

  "full\_name" varchar,

  "last\_name" varchar,

  "gender" varchar,

  "date\_of\_birth" date,

  "email" varchar,

  "phone" varchar,

  "position" varchar,

  "created\_at" timestamp,

  "load\_date" timestamp,

  "record\_source" varchar

);

CREATE TABLE "hub\_masters" (

  "master\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "master\_code" varchar

);

CREATE TABLE "link\_visit\_master" (

  "link\_visit\_master\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "master\_hash\_key" varbinary,

  "visitor\_log\_hash\_key" varbinary

);

CREATE TABLE "link\_qualification" (

  "link\_qualification\_hash\_key" varbinary PRIMARY KEY,

  "load\_date" timestamp,

  "record\_source" varchar,

  "master\_hash\_key" varbinary,

  "service\_hash\_key" varbinary

);

ALTER TABLE "sat\_clients" ADD FOREIGN KEY ("client\_hash\_key") REFERENCES "hub\_client" ("client\_hash\_key");

ALTER TABLE "hub\_client" ADD FOREIGN KEY ("client\_hash\_key") REFERENCES "link\_visit\_client" ("client\_hash\_key");

ALTER TABLE "hub\_visitor\_log" ADD FOREIGN KEY ("visitor\_log\_hash\_key") REFERENCES "link\_visit\_client" ("visitor\_log\_hash\_key");

ALTER TABLE "sat\_visitor\_log" ADD FOREIGN KEY ("visitor\_log\_hash\_key") REFERENCES "hub\_visitor\_log" ("visitor\_log\_hash\_key");

ALTER TABLE "sat\_services" ADD FOREIGN KEY ("service\_hash\_key") REFERENCES "hub\_services" ("service\_hash\_key");

ALTER TABLE "hub\_services" ADD FOREIGN KEY ("service\_hash\_key") REFERENCES "link\_visit\_service" ("service\_hash\_key");

ALTER TABLE "hub\_visitor\_log" ADD FOREIGN KEY ("visitor\_log\_hash\_key") REFERENCES "link\_visit\_service" ("visitor\_log\_hash\_key");

ALTER TABLE "sat\_masters" ADD FOREIGN KEY ("master\_hash\_key") REFERENCES "hub\_masters" ("master\_hash\_key");

ALTER TABLE "hub\_masters" ADD FOREIGN KEY ("master\_hash\_key") REFERENCES "link\_visit\_master" ("master\_hash\_key");

ALTER TABLE "hub\_visitor\_log" ADD FOREIGN KEY ("visitor\_log\_hash\_key") REFERENCES "link\_visit\_master" ("visitor\_log\_hash\_key");

ALTER TABLE "hub\_masters" ADD FOREIGN KEY ("master\_hash\_key") REFERENCES "link\_qualification" ("master\_hash\_key");

ALTER TABLE "hub\_services" ADD FOREIGN KEY ("service\_hash\_key") REFERENCES "link\_qualification" ("service\_hash\_key");