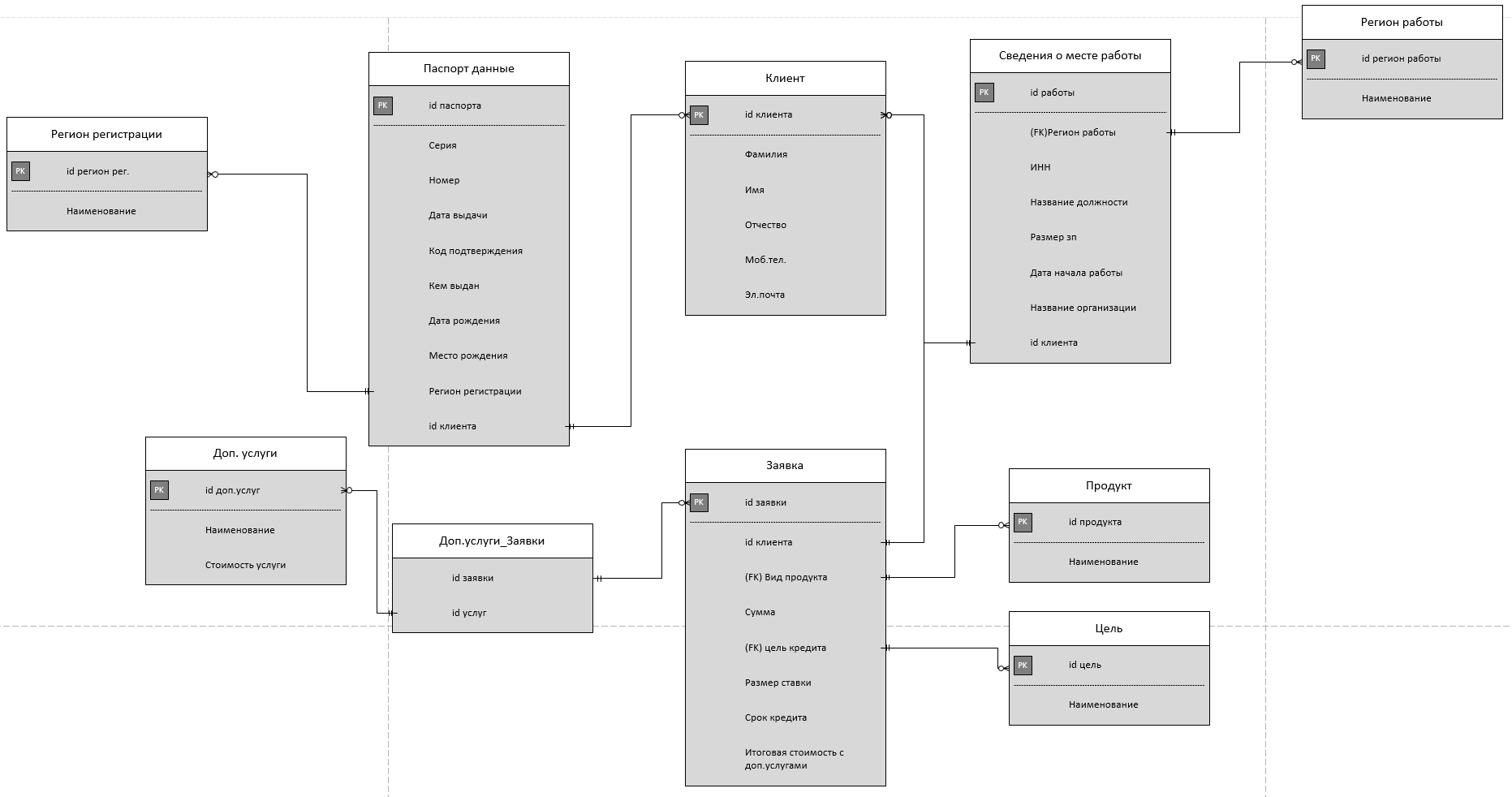
1. 

Рисунок - ER - диаграмма

По заданным скриншотам была разработана следующая ER-диаграмма(рис.1), содержащая сущности:

**Client** – содержит информацию о физических лицах, которые вступают во взаимодействие с банком. Информация о Client помогает банку при планировании, разработке продуктов и рекламной деятельности.  
**Pasport** – содержит информацию о паспортных данных физических лиц.  
**Region of registration** – содержит информацию о регионах регистрации физических лиц.  
**Work** – содержит информацию о работах физических лиц.  
**Region of work** – содержит информацию о регионах работы физических лиц.  
Request – содержит информацию о заявках кредитования физических лиц. Информация о Request помогает банку при планировании, разработке продуктов и рекламной деятельности.  
**Target**- содержит информацию о целях кредитования физических лиц.  
**Product** – содержит информацию о видах продукта кредитования.  
**Services** – содержит информацию о дополнительных услугах кредитования.  
**request\_services** – является связующей таблицей для связи между дополнительными услугами и заявками физических лиц.

Для полей приложения в которых предоставлен выбор (выпадающей список), были созданы отдельные сущности.

Для числовых полей выбран тип данных int(для целочисленных значений) и float(для дробных значений с плавающей запятой)  
Для строковых переменных выбран тип данных varchar(строковый тип переменной длины)  
Для полей с датами выбран формат данных date(хранение дат)

В итоге была создана база данных в СУБД – MySql версия 5.37 (рис.2)

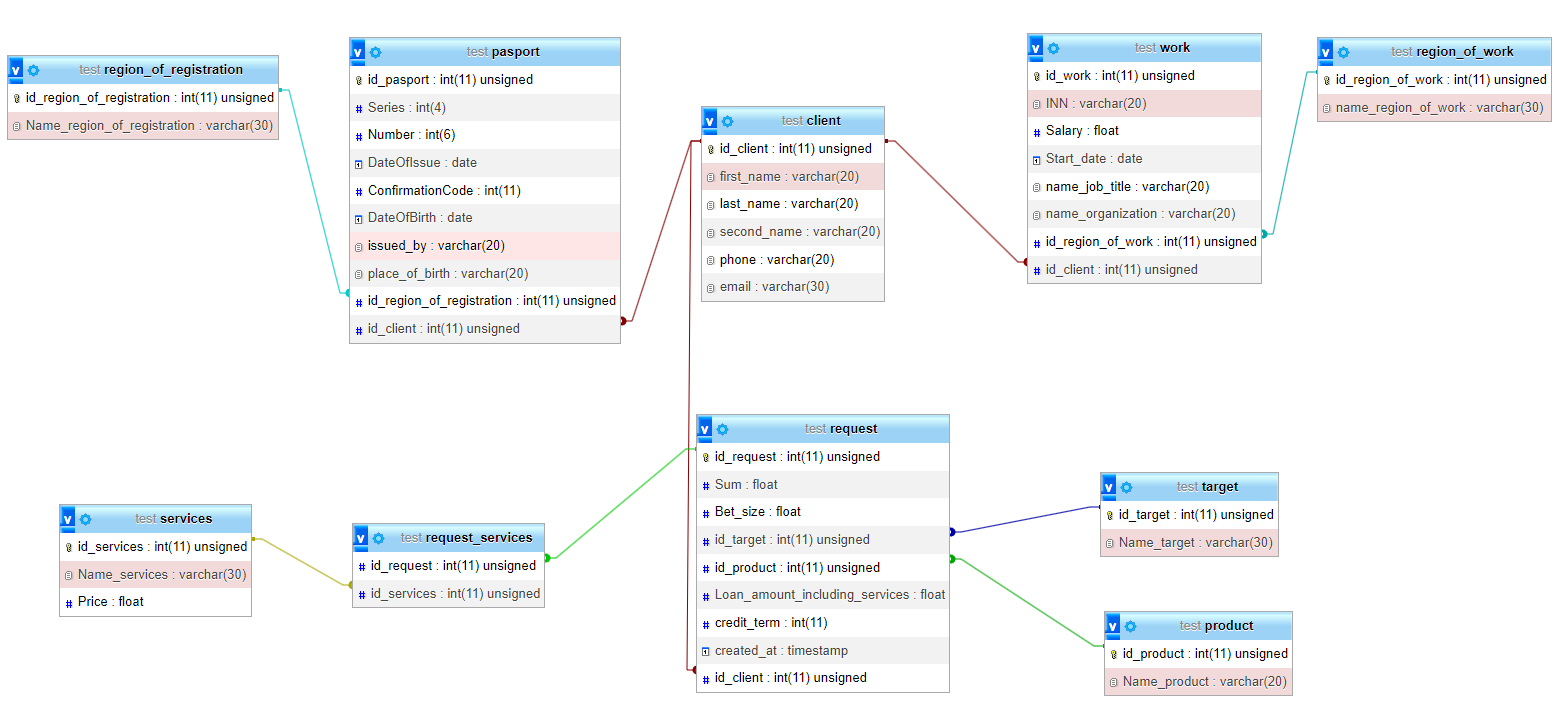
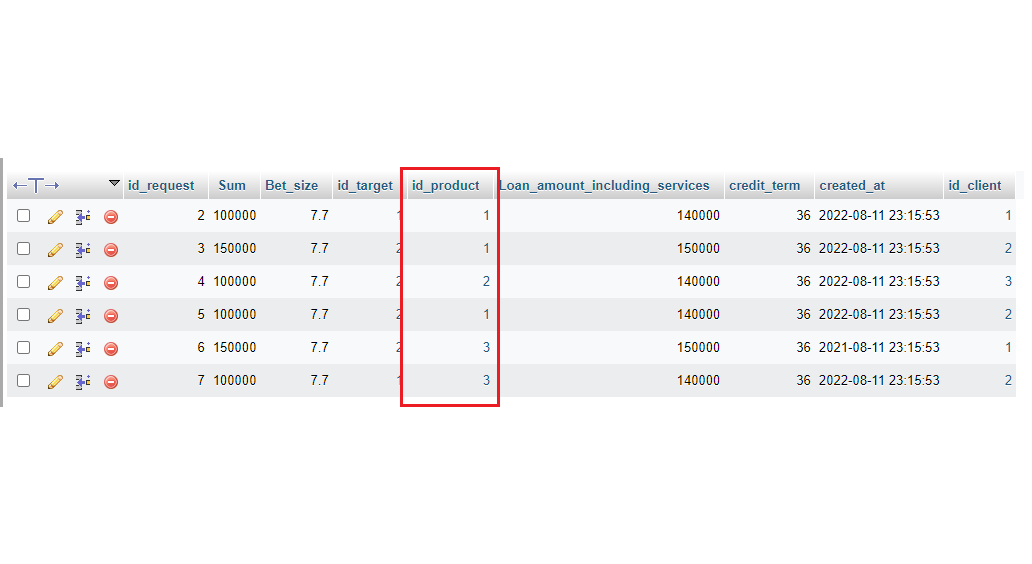


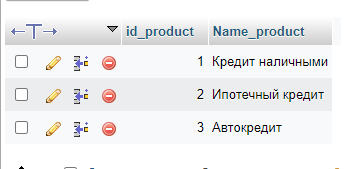
Рисунок - Физическая модель БД

2. Sql – запрос который возвращает самый популярный вид продукта за текущий год

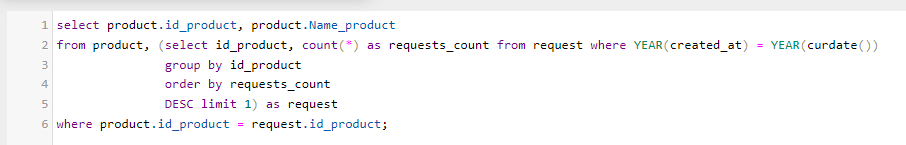
Данные таблицы «Заявки»



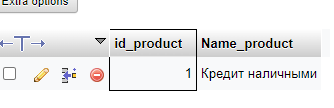
Данные таблицы «Вид продукта»



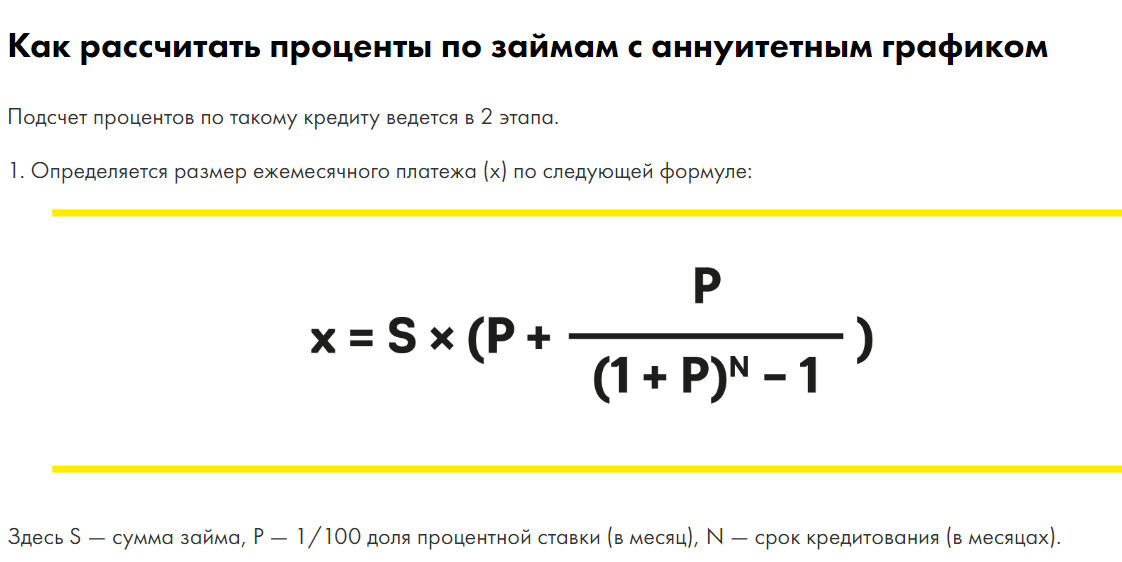
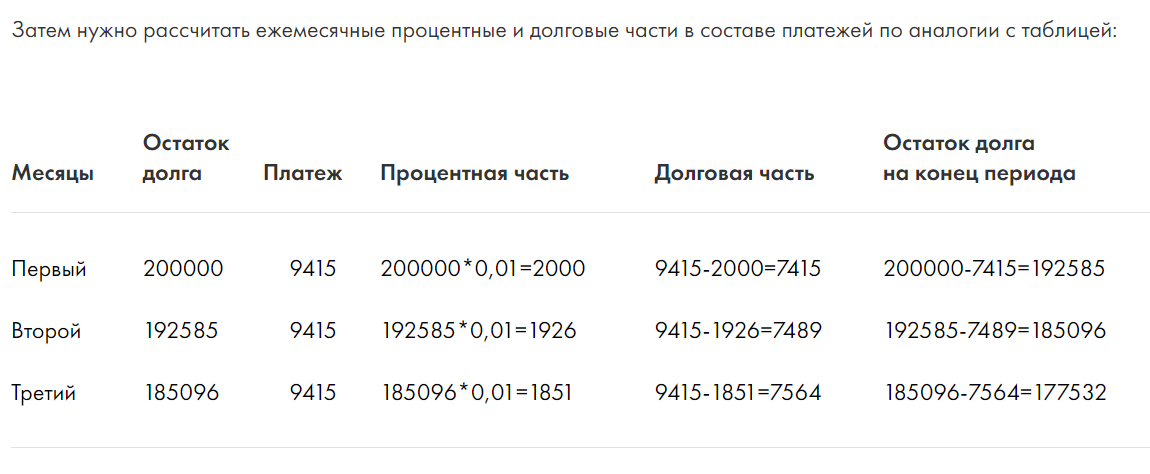
Sql-запрос



Результат выполненого запроса:



3. Формулы расчета графика аннуитетных платежей найдены в интернете

1. Предоставляем выбор для ввода данных:
   1. Из консоли
   2. Из БД
2. По формулам рассчитываем:
   1. Размер ежемесячного платежа
   2. Основной долг
   3. Долг по процентам
   4. Остаток основного долга
3. Вывод в табличном формате

Листинг кода

using System;

using MySql.Data.MySqlClient;

namespace Test

{

class Program

{

static void Main(string[] args)

{

double[] data = new double[3];

Console.Write("Выберите тип ввода: (1 - ввод с консоли, 2 из БД): ");

int type = Convert.ToInt32(Console.ReadLine());

if (type == 1) {

data = GetFromString();

} else {

data = GetFromDataBase();

}

double amount = data[0];

double rate = data[1];

double term = data[2];

double pay = amount \* (rate / (1 - Math.Pow((1 + rate), -term))); // размер ежемесячного платежа

Console.WriteLine("---------------------------------------------------------------------------------------------------");

Console.WriteLine("Месяц Ежемесячный платеж Основной долг Долг по процентам Остаток основного долга");

Console.WriteLine("---------------------------------------------------------------------------------------------------");

for (int month = 1; month <= term; month++)

{

double prosentPart = amount \* rate; // процентная часть

double debtPart = pay - prosentPart; // долговая часть

amount -= debtPart; //остаток долга

Console.WriteLine("{0, 5} {1, 15} {2, 15} {3, 20} {4, 27}", month, FormatNumber(pay), FormatNumber(debtPart), FormatNumber(prosentPart), FormatNumber(Math.Abs(amount)));

}

}

public static string FormatNumber(double number)

{

return number.ToString("0.##");

}

public static double[] GetFromString()

{

double[] data = new double[3];

Console.Write("Cумма кредита: ");

data[0] = Convert.ToDouble(Console.ReadLine());

Console.Write("Ставка: ");

data[1] = Convert.ToDouble(Console.ReadLine()) / 100 / 12;

Console.Write("Срок (мес.): ");

data[2] = Convert.ToInt32(Console.ReadLine());

return data;

}

public static double[] GetFromDataBase()

{

double[] data = new double[3];

// Подключение к базе данных

string cs = @"server=localhost;userid=root;password=;database=test";

var con = new MySqlConnection(cs);

con.Open();

Console.Write("Введите ID заявки из базы данных: ");

double requestId = Convert.ToDouble(Console.ReadLine());

var requestCommand = new MySqlCommand(

"SELECT `Loan\_amount\_including\_services`, `Bet\_size`, `credit\_term` from `request` " +

"WHERE `id\_request` = " + requestId + " LIMIT 1",

con);

MySqlDataReader rdr = requestCommand.ExecuteReader();

while (rdr.Read())

{

data[0] = rdr.GetDouble(0);

data[1] = Math.Round(rdr.GetDouble(1), 2);

data[2] = rdr.GetInt32(2);

Console.WriteLine("Cумма кредита: " + data[0]);

Console.WriteLine("Ставка: " + data[1]);

Console.WriteLine("Срок (мес.): " + data[2]);

data[1] = data[1] / 100 / 12;

}

return data;

}

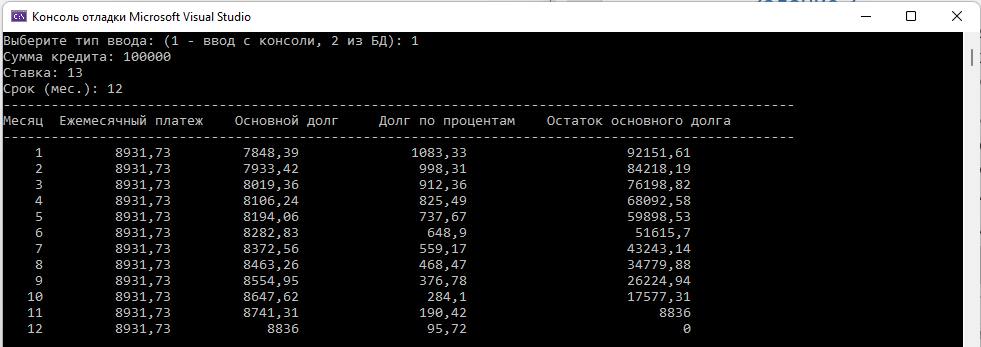
}

}

1. Результат откладки ввод с консоли:

Данные, на основе которых проводилось тестирование работы функции:

Сумма кредита: 100000  
Ставка: 13   
Срок (мес.): 12



1. Результат откладки ввод из БД:

Данные, на основе которых проводилось тестирование работы функции:

Сумма кредита: 140000  
Ставка: 7.7  
Срок (мес.): 36

