Практическая работа №2.5.

1. Базы данных. Язык запросов SQL. Работа с базой MySQL из JAVA. Создание игры "Предсказание будущего" с использованием базы данных.

1.1. Задача:Изучение теоретического материала. Закрепление полученных знаний путем разработки приложения на языке программирования Java в среде разработки eclipse.

1.2.Тип занятий: исследовательский.

1.3.Содержание: теоретический материал, задания для самостоятельного выполнения.

2. Задания

2.1. первая часть: создание интерфейса

1. Откройте eclipse. Командой File – new – Java Project создайте новый проект, в нем создайте пакет и командой Other создайте Application Window.
2. Добавьте на форму panel, кнопку с надписью "узнать свое будущее", метку для вывода текста и метку для отображения картинки. Не забывайте про absolute layout.

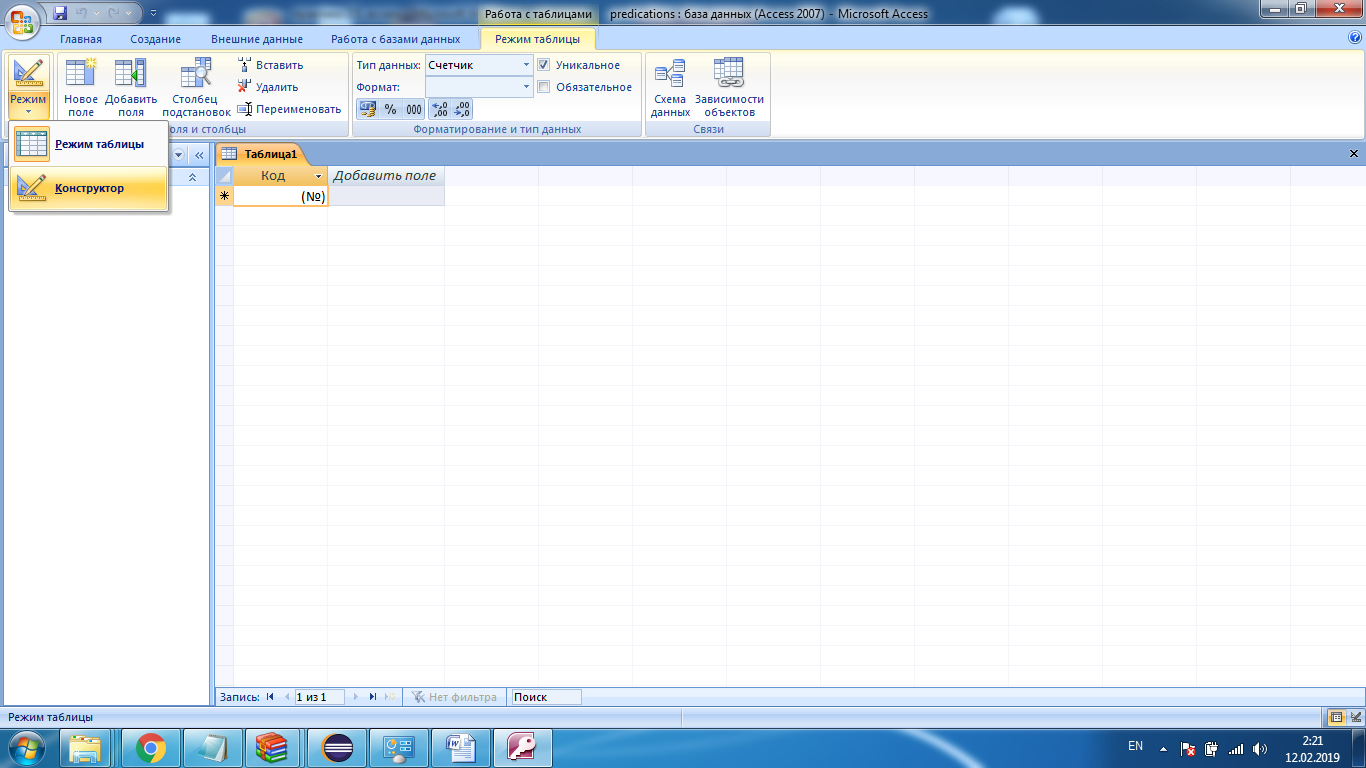
3.2. вторая часть: создание базы данных

1. Откройте программу Microsoft Access.
2. Создайте новую базу данных:

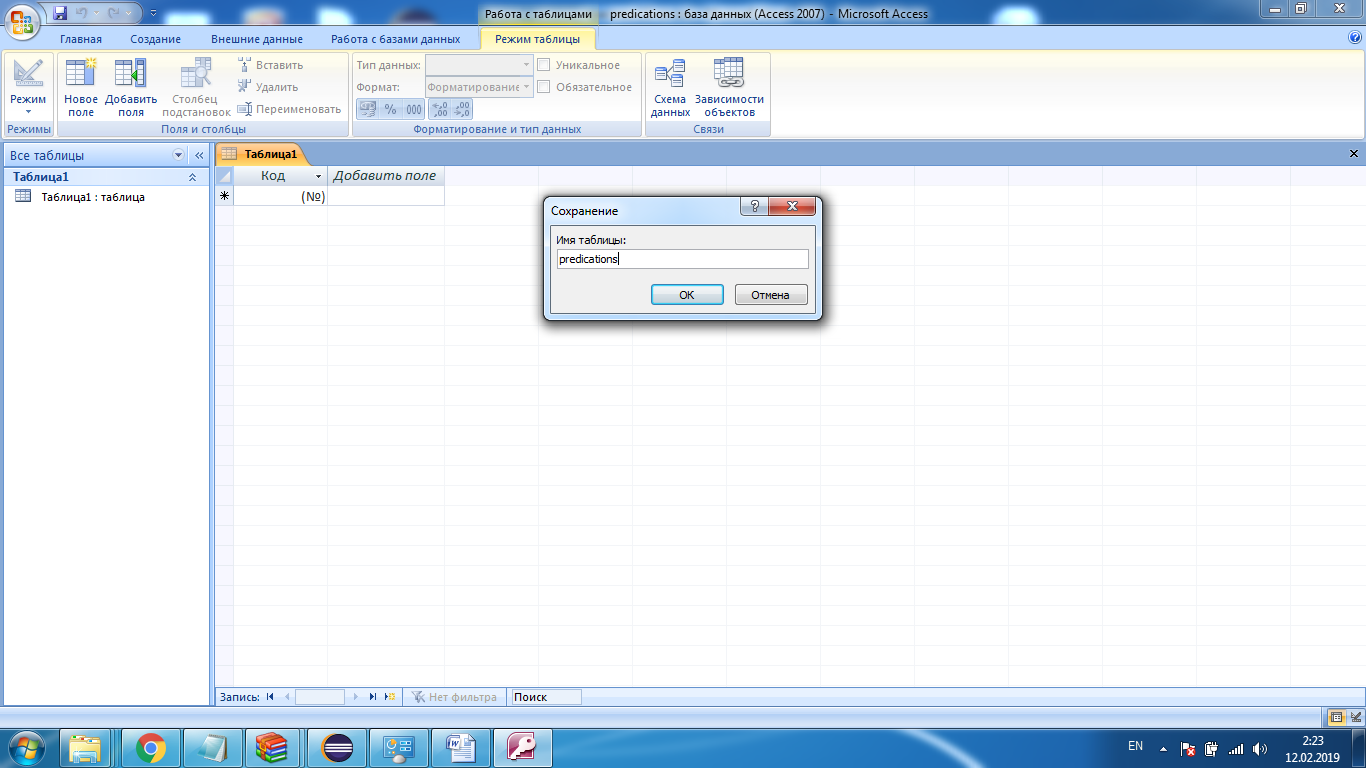
В правой части окна задайте имя и нажмите кнопку Создать:



Далее в следующем окне перейдите в режим конструктора:

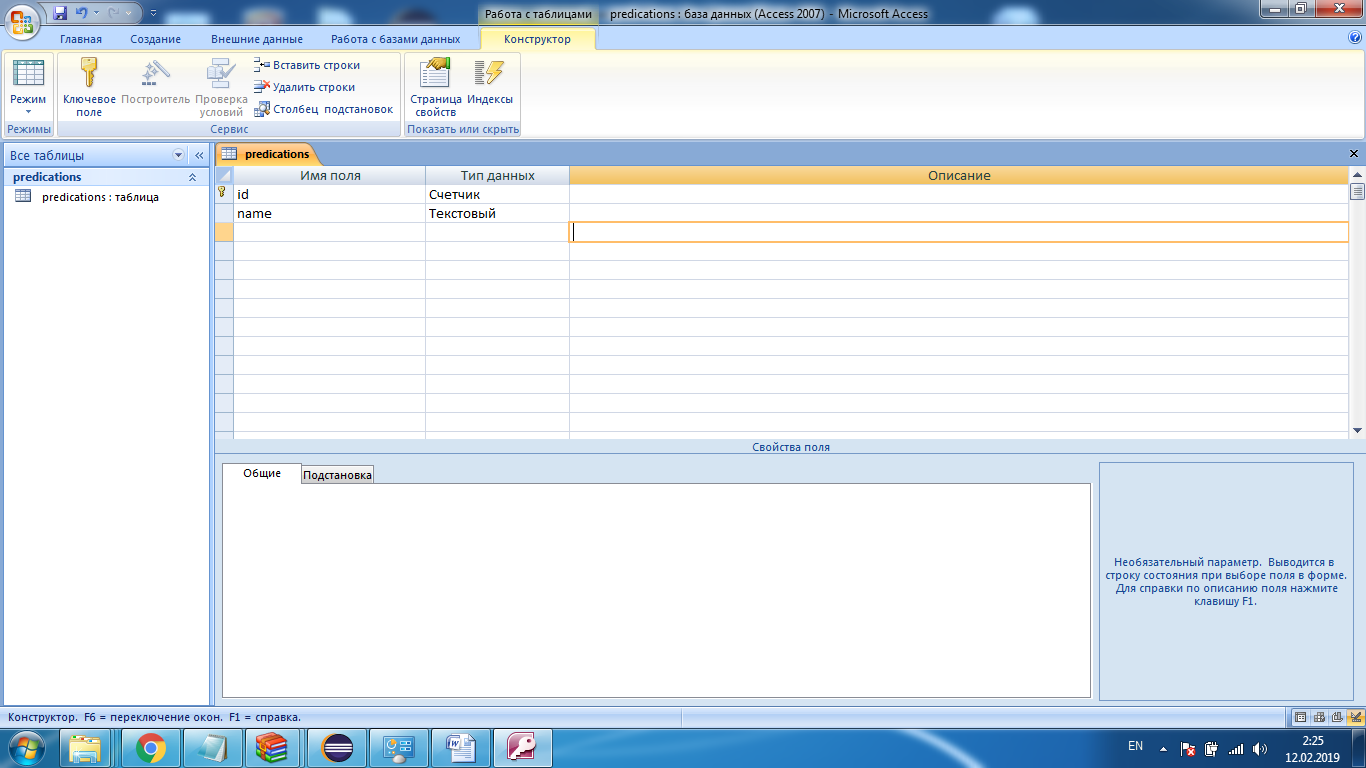


1. Задайте название вашей таблицы с предсказаниями (создадим нашу таблицу с предсказаниями, лучше давать английское название, предсказания - predications):

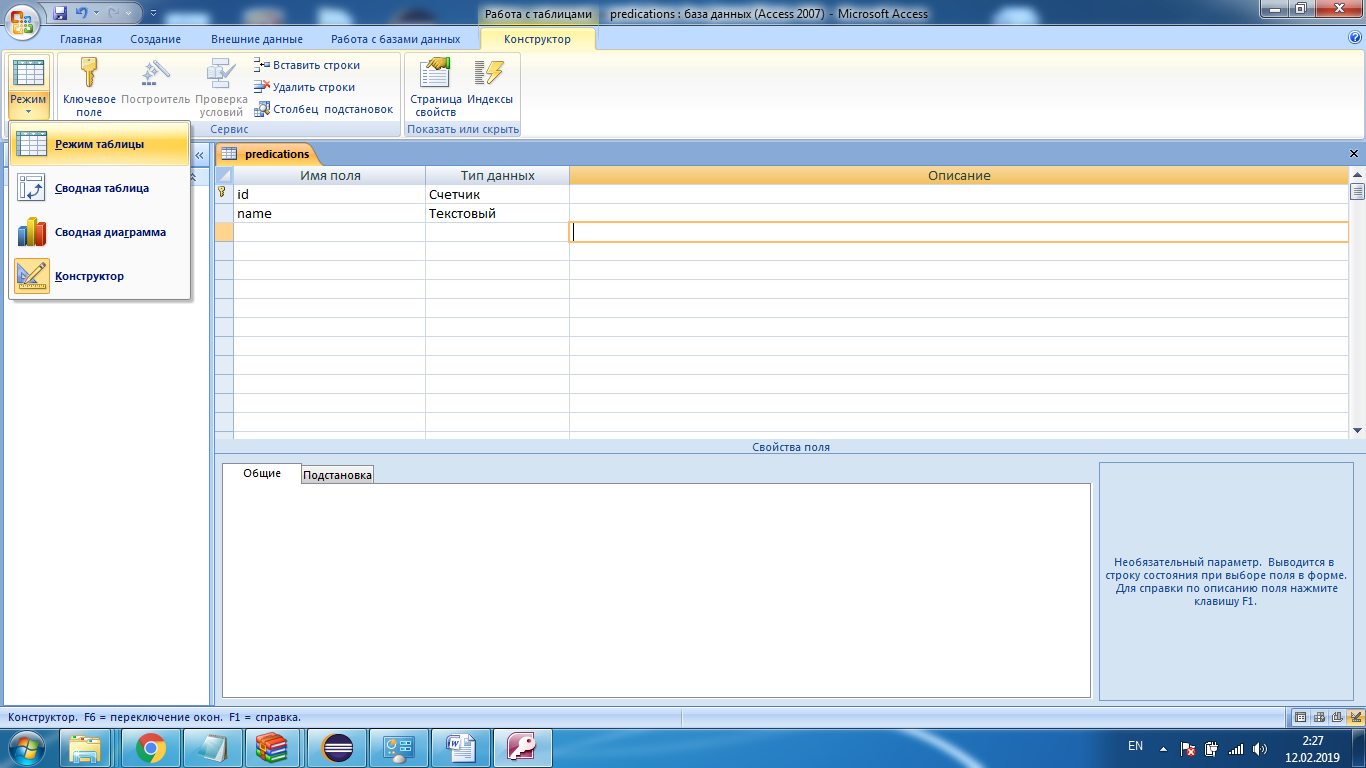


1. В следующем окне задаем столбцы: у вас должно быть два столбца:

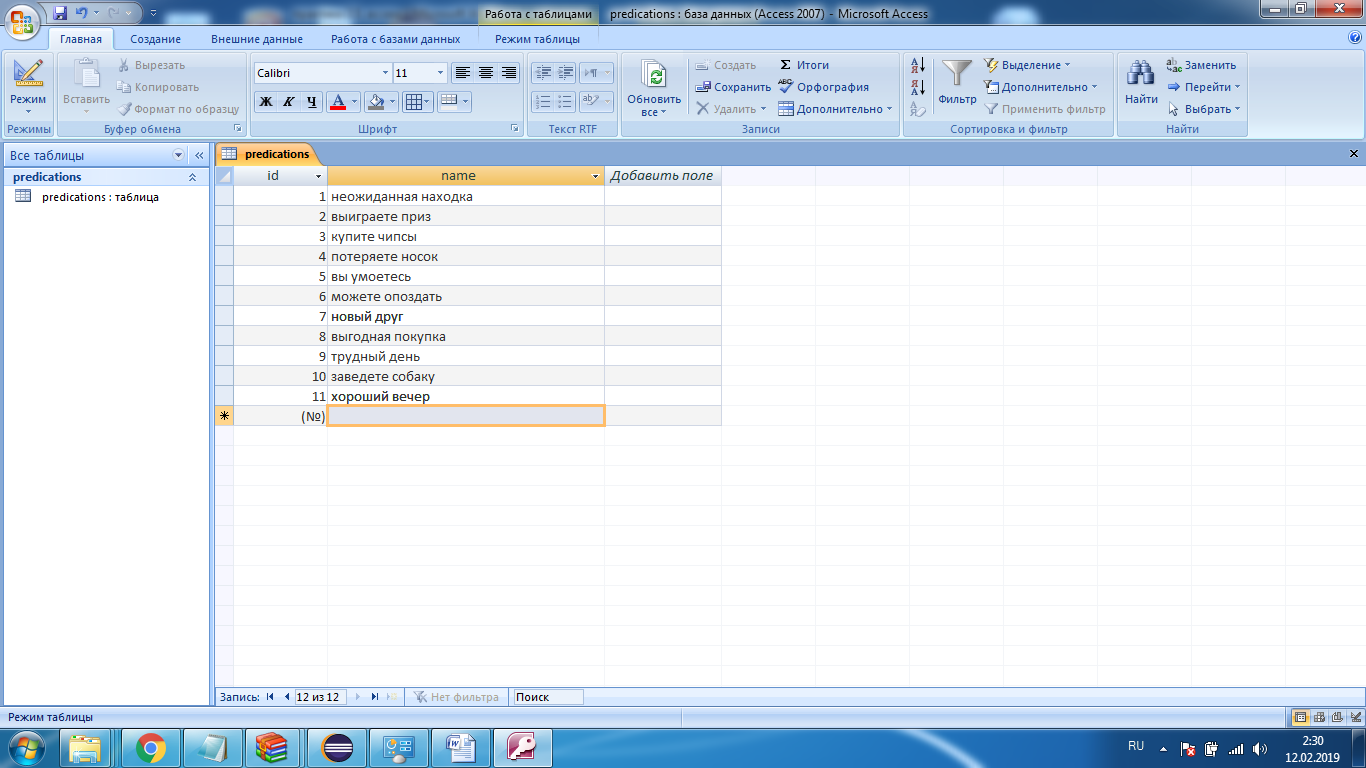
* id счетчик - счетчик записей с параметрами
* name текстовый - текстовое значение предсказания



Потом снова вернитесь в режим таблицы:



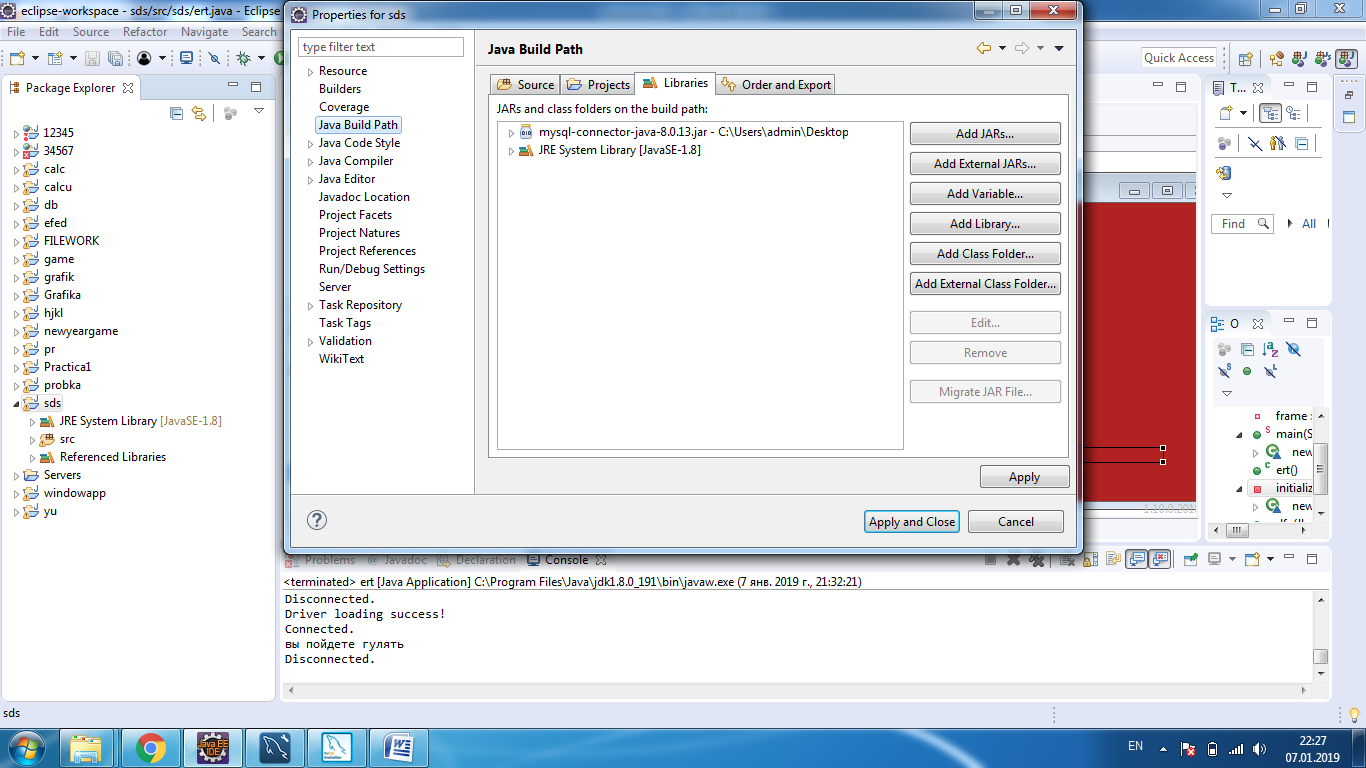
1. Заполните таблицу 11 значениями, придумайте их самостоятельно, при этом не нужно вводить значения в поле id, оно генерируется само:



1. Сохраните вашу базу данных.

3.3. третья часть: подключение к базе данных из JAVA

1. Поставьте курсор на свой проект в eclipse и нажав правой кнопкой мыши, перейдите в properties, в левом меню выберите JAVA Build Path и вкладку Libraries:



Далее нажав на Add External JARs выберите библиотеку для работы с базой данных MySQL из JAVA:

* jackcess
* ucanaccess-4.0.4
* hsqldb
* commons-lang3-3.1
* commons-logging-1.1.3

1. Далее нужно осуществить подключение к вашей базе данных, для этого в методе initialize() введем код:

Connection connection = null;

Statement statement = null;

ResultSet resultSet = null;

try {

Class.forName("net.ucanaccess.jdbc.UcanaccessDriver");

}

catch(ClassNotFoundException cnfex) {

System.out.println("Problem in loading or "

+ "registering MS Access JDBC driver");

cnfex.printStackTrace();

}

try {

Class.forName("net.ucanaccess.jdbc.UcanaccessDriver");

Connection

conn=DriverManager.getConnection("jdbc:ucanaccess://C:\\Users\\IT-cube\\Documents\\predications.accdb");

System.out.println("Succesfully connected");

}

catch(SQLException sqlex){

sqlex.printStackTrace();

}

finally {

try {

if(null != connection) {

resultSet.close();

statement.close();

connection.close();

}

}

catch (SQLException sqlex) {

sqlex.printStackTrace();

}

}

1. Запустите приложение, если подключение/отключение к бд проходит
2. успешно, можно перейти к работе с таблицей.

После задания подключения добавим // Creating JDBC Statement

statement = conn.createStatement();

// retrieve data into ResultSet

resultSet = statement.executeQuery("SELECT \* FROM имя\_вашей\_таблицы");

// вывод значений

**while**(resultSet.next()) {

System.***out***.println(resultSet.getInt(1) + "\t" +

resultSet.getString(2));

}

Теперь у вас в консоли должны отразятся все ваши строки из таблицы, если они появились, перейдем к реализации гадания, для этого нам всего лишь потребуется задать массив String [] mass = **new** String[11]; //массив для хранения предсказаний

**int** i=0; //счетчик для массива

А внутри цикла с выводом предсказаний (вывод можно закомментировать) добавить:

mass[i]= resultSet.getString(2); //присваиваем нашему массиву предсказания из таблицы

i++;

1. после цикла инициализируем переменную, получающую случайное значение от 0 до 10 и нужно выводить на экран значение массива только с индексом, которым будет служить случайное число:

**int** b = (**int**) (Math.*random*()\*10);

System.***out***.println(mass[b]);

13) Почти готово, создадим в классе метод **public** **void** dfe(JLabel label) **throws** ClassNotFoundException{} и перенесем весь наш код в него, параметром в нем служит метка для записи предсказания. Чтобы вызвать метод нужно в методе initialize(), а также внутри события по нажатию кнопки добавить dfe(lblNewLabel\_1); - внутри имя вашей метки, куда будет выводиться текст.

14) Теперь в наш метод перед закрытием подключения добавьте label.setText(mass[b]); Запускайте.