***ОМЕГА НЕРЕАЛЬНО КРУТОЙ ГАЙД НА ТРОЕЧКУ ОТ ВАЛЕРЫ МИРОНОВА***

1. Все таблицы и файлы отредачить и подогнать под адекватный вид в блокноте:

-поменять кодировку на utf-8;

-переименовать первую строку на английский;

-три разделителя заменить на пробел;

-разделитель с пробелом заменить на пробел;

-двойные пробелы заменить на пробел;

-в конце пробел надо заменить на разделитель;

-обратные слеши (\) поменять на нормальные (/);

-сохранять в формате scv.

1. В Воркбенче подключаемся к базе данных и создаем новую схему базы данных с кодировкой utf-8 (или выбираем ее, если она будет на демке).
2. В файле my.sql ON DELETE NO ACTION заменить на ON DELETE CASCADE.
3. Импорт структуры базы данных через Server🡪Data Import, ставим галочку на Import from Self-Contained File и выбираем файл my.sql.
4. Обновляем схемы и видим появившиеся таблицы. Нужно импортировать отредаченную в первом пункте таблицу. Клацаем дважды по базе данных, выбираем нужную схему, открываем ее, нажимаем вторую кнопку рядом с Export/Import, выбираем нужный файл, затем use existing table, находим подходящую существующую схему, затем сопоставляем колонки с названиями в файле, затем при импорте нажимаем шоу логс, чтобы смотреть наличие ошибок.

P.S. Если появится ошибка, связанная с таблицей ключей, сначала заполняем ее, потом работаем со схемой, из которой данные ссылаются на ключи.

1. В схемах с неправильной датой и временем нужно в exсel выделить нужный столбик, нажать ПКМ🡪формат ячеек🡪все форматы🡪ДД.ММ.ГГГГ ч:мм с заменой на ГГГГ-ММ-ДД чч:мм:сс
2. Если excel таблица не отображает русский, надо поменять кодировку на utf-8 через блокнот и в нем же изменить первую строчку на английский язык.
3. В Воркбенче Создать новую соединяющую таблицу, копируя строки из my.sql

CREATE TABLE IF NOT EXISTS tmp

(

LastName VARCHAR(100) CHARACTER SER ‘utf8mb4’ NOTNULL,

Title VARCHAR(100) CHARACTER SER ‘utf8mb4’ NOTNULL,

StartTime DATETIME NOT NULL

);

1. В созданную таблицу загрузить отредактированный файл из пунктов 5 и 6.
2. Выполнить запрос по соединению таблиц

INSERT INTO ClientService (StartTime, ClientID, ServiceID)  
SELECT tmp.StartTime as StartTime, Client.ID as ClientID, Service.ID as ServiceID  
FROM tmp  
INNER JOIN Client ON tmp.LastName = Client.LastName  
INNER JOIN Service ON tmp.Title = service.Title

1. В IDEA надо создать новый Java проект, в src создать пакет с названием приложения (com.polushka.app), а в нём создаём пакеты ui и utils. В папке проекта создаём папку resources и в IDEA клацаем по ней ПКМ, выбираем Mark Directory as resources root. В неё закидываем иконку и папку с картинками.
2. Затем нужно добавить в проект библиотеку mysql connect через File🡪Project Structure🡪Libraries🡪+from maven, в ней найти mysql.
3. Создаем в пакете ui новый GUI Form ClientTableForm. В JPanel переименовываем field name и кидаем в него JScrollPane и JTable, последний также переименовываем. В классе дописываем extends BaseForm после ClientTableForm. В нём пишем

setContentPane(mainPanel); //выбор панели

table.setRowHeight(25); //увеличение строк

// Создание модели данных для таблицы

DefaultTableModel client\_mdl = new DefaultTableModel();

// Указание названий столбцов

client\_mdl.setColumnIdentifiers(new String[]{"ID","Имя","Фамилия","Отчество","Дата рождения","Дата регистрации","Почта","Номер телефона","Пол","Фото"});

// Ловятся исключения, связанные с БД

try {

// Подключение к БД demoexam по адресу 127.0.0.1:3306, пользователь - root

Connection connection = DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/demoexam?serverTimeZone=UTC","root","");

// Подготовка запроса

PreparedStatement preparedStatement = connection.prepareStatement("SELECT \* FROM client");

// Выолнение запроса, в ResultSet помещаются строки из БД

ResultSet resultSet = preparedStatement.executeQuery();

// Прогон по каждой cтроке БД по порядку, пока строки существуют

while (resultSet.next()){

// Добавление в модель строки с данными из БД

client\_mdl.addRow(new String[]{

// Взятие из строки ResultSet'а значения из первого столбца

resultSet.getString(1),

resultSet.getString(2),

resultSet.getString(3),

resultSet.getString(4),

resultSet.getString(5),

resultSet.getString(6),

resultSet.getString(7),

resultSet.getString(8),

resultSet.getString(9).equals("м") ? "Мужской" : "Женский",

resultSet.getString(10)

});

}

connection.close();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

// Установка созданной модели на объект таблицы

table.setModel(client\_mdl);

1. В utils создаём класс BaseForm, после него пишем extends JFrame.

В нем пишем параметры окна

setTitle("ООО Фирма");//название приложения

setMinimumSize(new Dimension(1400,800)); //установка разрешения окна

setLocationRelativeTo(null); //центрирование окна  
setIconImage(new ImageIcon(getClass().getResource("imgs/school\_logo.png")).getImage());//добавление иконки

1. Создаём класс Main и в классе пишем psvm для шаблона. В нём пишем

ClientTableForm form = new ClientTableForm();

form.setVisible(true);

form.pack();

1. Затем нужно собрать проект через File🡪 Project Structure🡪Artifacts🡪+Jar From modules… В диалоговом окне выбираем главный класс и соглашаемся со всем. Теперь в разделе Build можно вызывать Build Artifacts, в которой выбрать Build для создания приложения.
2. В начале класса через конструкцию

/\*\*

\*/

надо описать классы: что делают и зачем нужны.

1. В папке с заданием будет шаблон README файла. Его надо скопировать в папку с проектом и переименовать в README.md  
   Открыв файл через IDEA, можно сразу увидеть то, как он будет выглядеть. Описываем по инструкции.
2. Создаём репозиторий на гите (или где мы там будем работать). Копируем ссылку на репозиторий. Затем открываем папку с проектом, в свободном месте нажимаем ПКМ🡪Git Bush Here. В командной строке пишем последовательно
3. git init
4. git remote add origin \*ссылка на репозиторий\*
5. git add -A
6. git commit -m “описание коммита”
7. git branch session1
8. git checkout session1
9. git push

и копируем ту команду, которую предложит командная строка