# Московский авиационный институт

**(Национальный исследовательский университет)**

Факультет прикладной математики и информационных технологий Кафедра вычислительной математики и программирования

Курсовой проект

по курсу «Базы данных»

Тема: Организация работы цеха сборки PC.

Студенты: Черненко И. Д.

Орозбакиев Э. Д.

Косогоров В.В.

Симонов С.Я.

Группа: М8О-306Б Преподаватель: Кузнецова Е. В.

Оценка:

Москва, 2020

# Техническое задание.

# 1.Технические требования

# Постановка задачи: требуется разработать прикладное программное обеспечение для хранения данных о заказах компьютеров сборочного цеха коммерческой компании. Клиенты собирают виртуальную модель компьютера из различных комплектующих, представленных в конструкторе.

Рабочий может посмотреть всю базу имеющихся комплектующих, а также сведения о назначенных ему заказах. Администрация, в отличие от клиентов и рабочих, может посмотреть всю базу имеющихся комплектующих и всю базу заказов, а также составить товарный чек для клиента.

Клиент может собирать компьютеры и смотреть данные о своих заказах. Каждая модель компьютера должна содержать только один вид каждого комплектующего. Каждый заказ может содержать только одного клиента и одного работника, ответственного за заказ. Каждый заказ может содержать только одну модель компьютера.

**2. Требования по функциональности**

Сборочному цеху требуется держать информацию о сотрудниках, клиентах и деталях для заказов.

1. Возможность обновления БД: добавление заказов клиентом.
2. Поиск информации в БД.
3. Возможность создания отчетности.

**3. Требования по интерфейсу**

Важной частью вышеупомянутой системы, является возможность получения готовых отчетов.

Перечень доступных документов.

1. Расчетных лист заказа.
2. Справочная информация о количестве комплектующих на складе.

**4. Требования по безопасности**

1. Разделение пользователей на доверенных и частично доверенных средствами СУБД.
2. Проверка правильности ввода исходных данных пользователем.

**Инструменты**

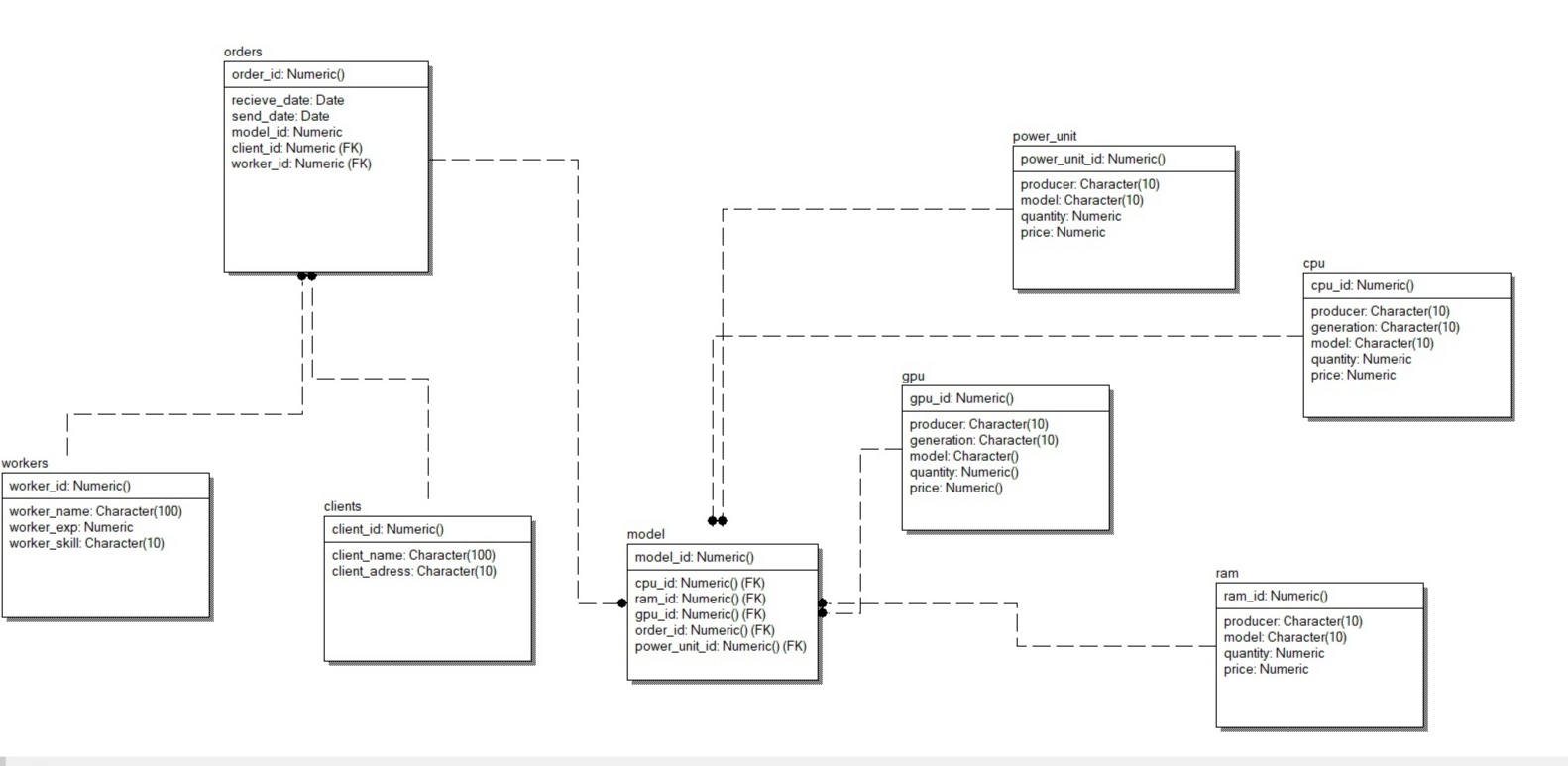
CASE-средство –Erwin Data Modeler;

СУБД –Microsoft Visual FoxPro 9.0.

**Разделение деятельности**

|  |  |
| --- | --- |
| Черненко И.Д. | Соединение разрозненных частей проекта, создание ветки форм администратора, разработка базы данных. |
| Орозбакиев Э.Д. | Разработка внешнего вида, создание ветки форм клиента, разработка базы данных. |
| Косогоров В.В. | Написание запросов |
| Симонов С.Я. | Создание отчетов, создание ветки форм сотрудника |
| Командная деятельность | Разделение пользователей на доверенных и частично доверенных. |

**Логическая модель в ERWIN:**

**

**Физическая модель в ERWIN:**

# 

**Формы**

1. **Формы администратора.**

|  |  |
| --- | --- |
| **Название** | **Описание** |
| admin\_orders | Просмотр заказов по заданной дате, клиенту и рабочему |
| admin\_items | Поиск моделей по заданному названию и просмотр всех комплектующих на складе |
| report | Печать отчета по клиенту и id заказа |

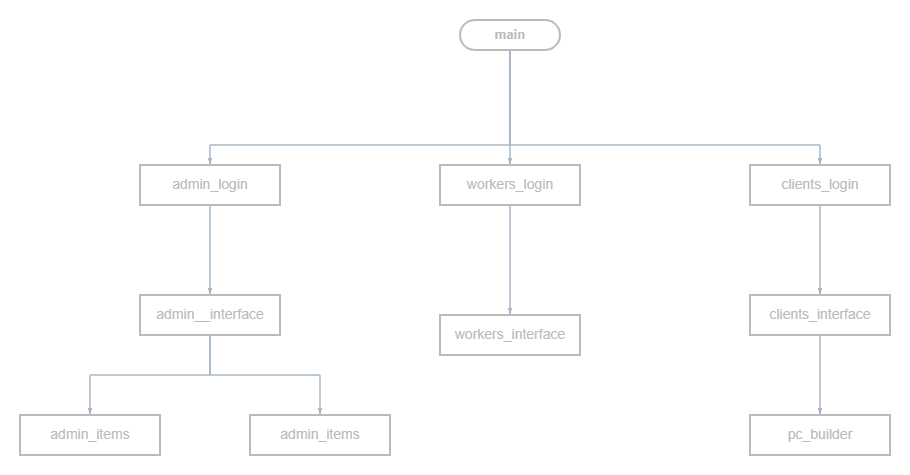
2. **Формы рабочего.**

|  |  |
| --- | --- |
| **Название** | **Описание** |
| workers\_interface | Вывод всех заказов определенного рабочего, просмотр всех комплектующих на складе |

3. **Формы клиента.**

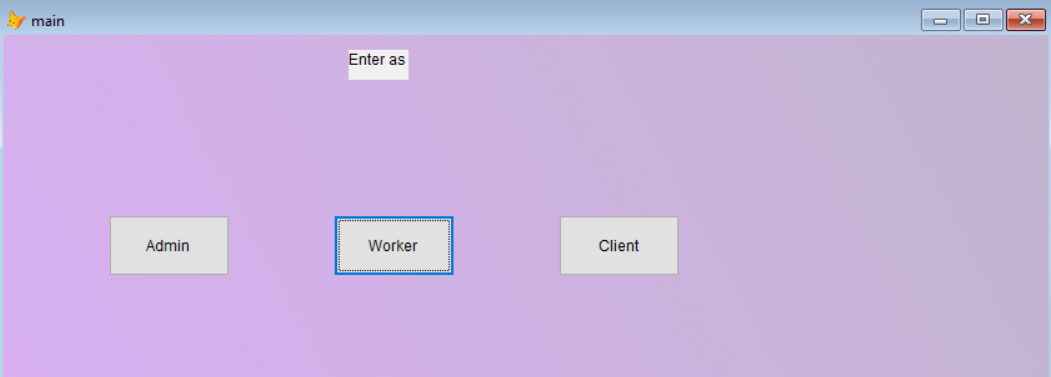
|  |  |
| --- | --- |
| **Название** | **Описание** |
| client\_interface | Просмотр текущих заказов клиента данной учетной записи |
| pc\_builder | Формирование заказа |

**Схема форм**

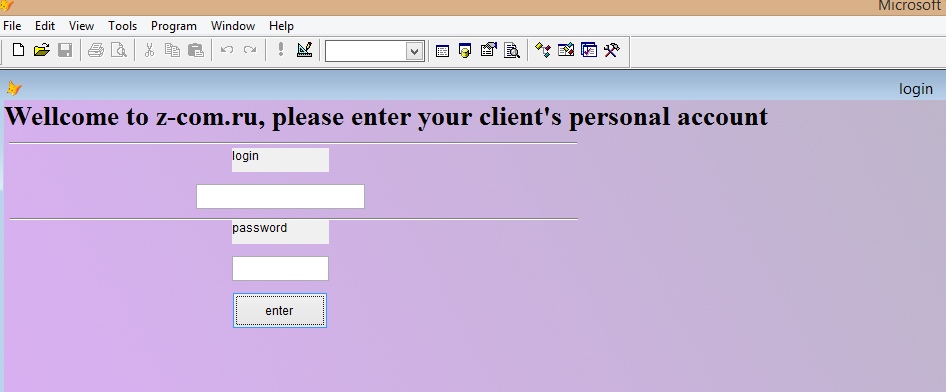
****

**Скриншоты форм**

1. ***Главное меню***

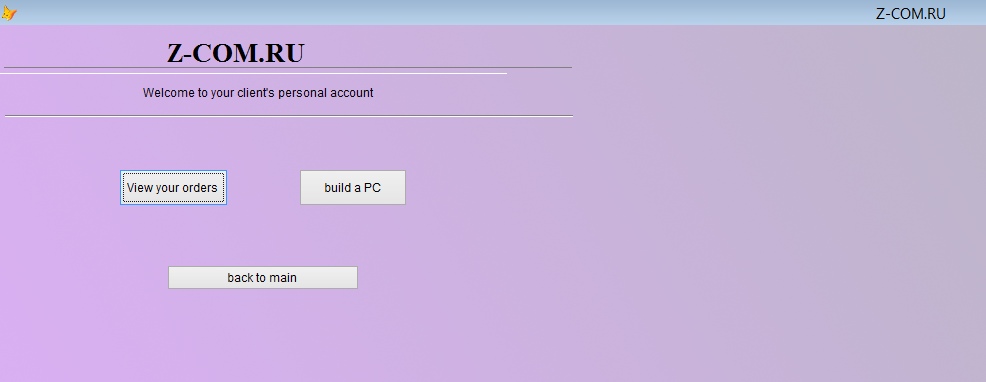
******

*2.* ***Страница авторизации клиента***

******

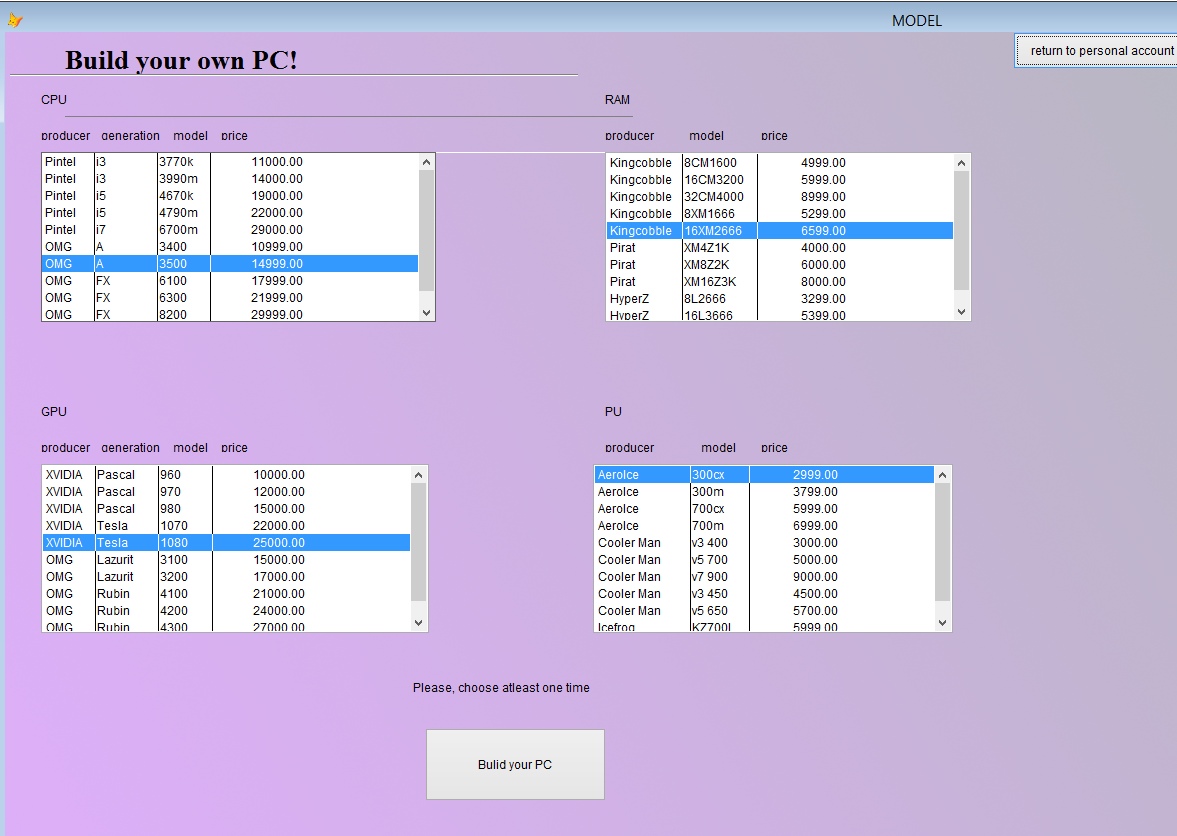
|  |  |
| --- | --- |
| enter | PUBLIC nameREZ  nameREZ = thisform.text1.Value  DECLARE REZ\_NAME(1)  SELECT c.name FROM clients c WHERE c.name IN (nameREZ) INTO ARRAY REZ\_NAME  FOR i =1 TO ALEN(Rez\_Name)  IF TYPE('REZ\_NAME')=TYPE('nameREZ') AND REZ\_NAME = nameREZ  SELECT c\_id FROM clients WHERE name IN (nameREZ) INTO ARRAY REZ  SELECT passw FROM passwords WHERE c\_id = REZ INTO ARRAY REZ1  IF (thisform.password.text= REZ1(1))  DO FORM "C:\Users\ilya\Documents\Visual FoxPro Projects\cp\_bd\_project\clients\_interface"  RELEASE thisform  else  =MESSAGEBOX('Неправильный пароль')  THISFORM.password.SetFocus  RETURN  ENDIF  else  =MESSAGEBOX('Неправильный логин')  THISFORM.password.SetFocus  RETURN  ENDIF  ENDFOR |

*3.* ***Главное меню клиента***

******

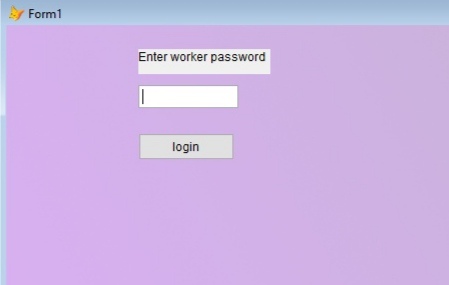
|  |  |
| --- | --- |
| Viuw your orders | SELECT o.o\_id AS order\_id, o.m\_id AS model\_id, o.sd AS shipping\_date;  FROM orders o, clients c;  WHERE (c\_idREP = c.c\_id) AND (c.c\_id = o.c\_id) |

4. ***Создание заказа комплектующих(build a PC)***

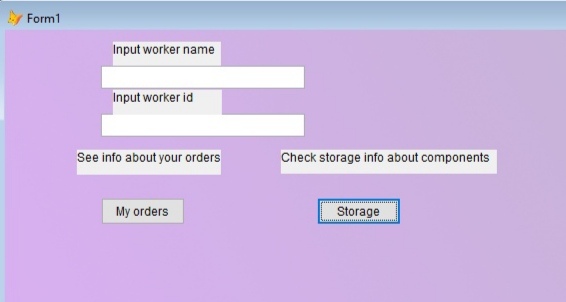
******

|  |  |
| --- | --- |
| Build your PC | MESSAGEBOX("Ваш заказ в обработке!",0)  PUBLIC w\_idREP  SELECT w.w\_id AS id, w.exp, COUNT(\*) AS counts;  FROM orders o, workers w;  WHERE W.w\_id = O.w\_id;  GROUP BY W.w\_id, W.exp;  INTO CURSOR tmp;  SELECT w.w\_id, DATE(), DATE() + IIF(w.skill = "Low", 10, IIF(w.skill = "Medium", 5, 3));  FROM tmp, workers w INTO ARRAY REZ;  WHERE tmp.counts = (SELECT MIN(counts) FROM tmp);  AND tmp.exp = (SELECT MAX(exp) FROM tmp) AND w.w\_id = tmp.id;  INSERT INTO model (cpu\_id, ram\_id, gpu\_id, pu\_id) VALUES (thisform.list1.value,thisform.list2.value,thisform.list3.value,thisform.list4.value)  PUBLIC m\_idREP  m\_idREP = model.m\_id  INSERT INTO ORDERS(rd, sd, m\_id, c\_id, w\_id) VALUES (REZ(2), REZ(3), m\_idREP, c\_idREP, REZ(1))  RELEASE thisform |

5. ***Страница авторизации работника***

******

6. ***Главное меню работника***

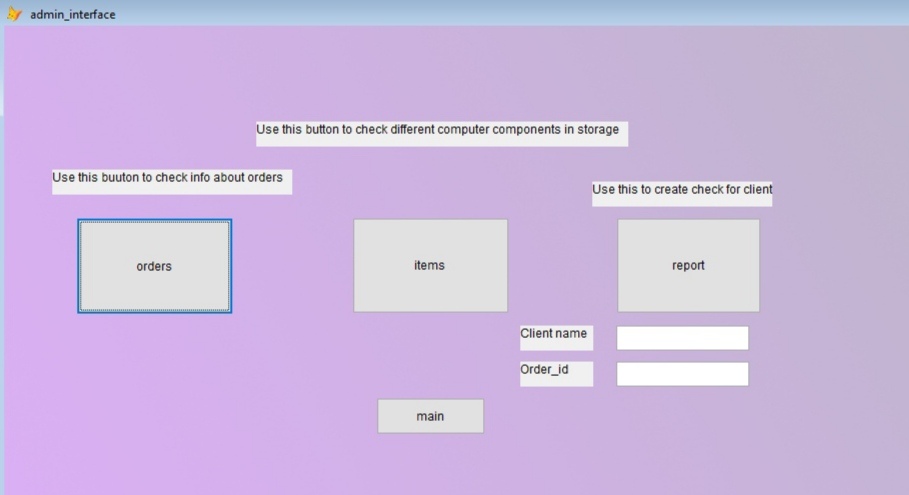


|  |  |
| --- | --- |
| My orders | SELECT o.o\_id AS order\_id, pu.model AS pu\_model, gpu.model AS gpu\_model, cpu.model AS cpu\_model, ram.model AS ram\_model, c.name AS client\_name, o.sd AS shipping\_date;  FROM orders o, model m, clients c, workers w, pu, gpu, cpu, ram;  WHERE w.name = thisform.input\_worker\_name.Text AND w.w\_id = thisform.input\_worker\_id.Text;  AND c.c\_id = o.c\_id AND o.w\_id = w.w\_id AND o.m\_id = m.m\_id;  AND m.pu\_id = pu.pu\_id AND m.gpu\_id = gpu.gpu\_id AND m.cpu\_id = cpu.cpu\_id AND m.ram\_id = ram.ram\_id;  ORDER BY o.o\_id ASC |
| Storage | SELECT 'PU' AS item\_type, producer, model, quantity FROM pu;  UNION;  SELECT 'GPU' AS item\_type, producer, model, quantity FROM gpu;  UNION;  SELECT 'CPU' AS item\_type, producer, model, quantity FROM cpu;  UNION;  SELECT 'RAM' AS item\_type, producer, model, quantity FROM ram |

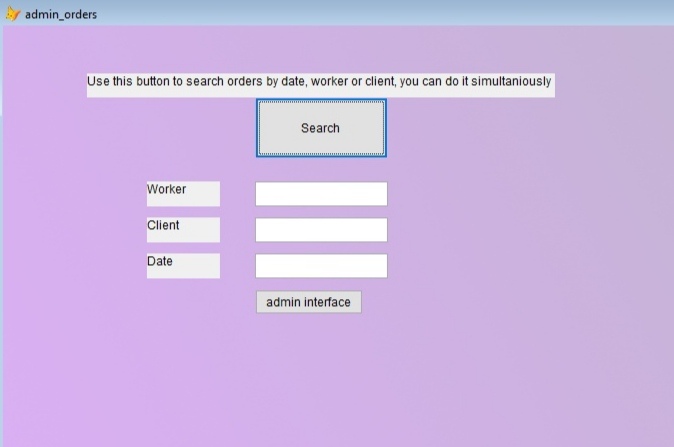
7. ***Страница авторизации администратора***

******

8. ***Главное меню администратора***

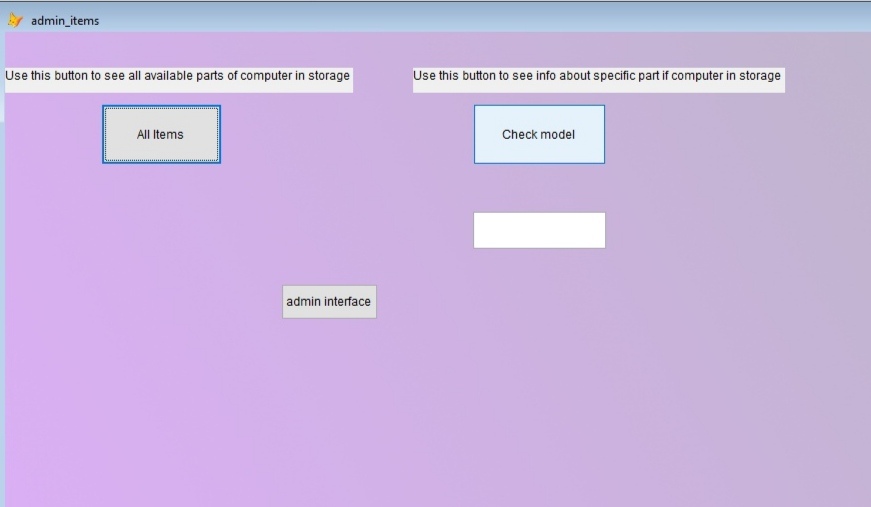
******

9. ***Просмотр заказов (по дате, клиенту и рабочему)(orders)***

******

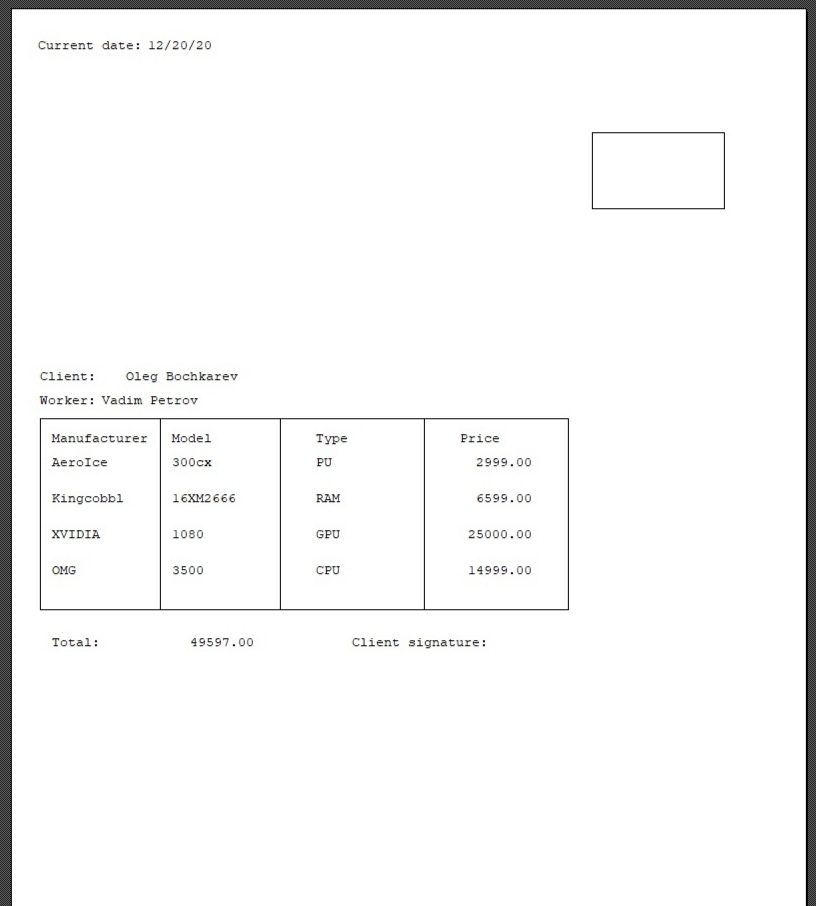
|  |  |
| --- | --- |
| Serch | INPUT\_WORKER\_NAME = thisform.text1.value  INPUT\_CLIENT\_NAME = thisform.text2.value  INPUT\_CREATION\_DATE = thisform.text3.value  SELECT c.name AS client\_name, pu.model AS pu, gpu.model AS gpu, cpu.model AS cpu, ram.model AS ram, o.sd AS shipping\_date;  FROM clients c, pu, gpu, cpu, ram, orders o, workers w, model m;  WHERE o.w\_id = w.w\_id AND o.m\_id = m.m\_id AND o.c\_id = c.c\_id;  AND m.pu\_id = pu.pu\_id AND m.gpu\_id = gpu.gpu\_id AND m.cpu\_id = cpu.cpu\_id AND m.ram\_id = ram.ram\_id;  and w.name = IIF(INPUT\_WORKER\_NAME LIKE "\_%", INPUT\_WORKER\_NAME, w.name);  and c.name = IIF(INPUT\_CLIENT\_NAME LIKE "\_%", INPUT\_CLIENT\_NAME, c.name);  and o.rd = IIF("" LIKE "\_%", "", o.rd); |

10. ***Поиск моделей по заданному названию и просмотр всех комплектующих на складе***

****

|  |  |
| --- | --- |
| all\_items | SELECT 'PU' AS item\_type, producer, model, quantity FROM pu;  UNION;  SELECT 'GPU' AS item\_type, producer, model, quantity FROM gpu;  UNION;  SELECT 'CPU' AS item\_type, producer, model, quantity FROM cpu;  UNION;  SELECT 'RAM' AS item\_type, producer, model, quantity FROM ram |
| Model\_check | NPUT\_MODEL = thisform.Text1.Value  SELECT 'PU' AS item\_type, producer, model, quantity FROM pu WHERE model = INPUT\_MODEL;  UNION;  SELECT 'GPU' AS item\_type, producer, model, quantity FROM gpu WHERE model = INPUT\_MODEL;  UNION;  SELECT 'CPU' AS item\_type, producer, model, quantity FROM cpu WHERE model = INPUT\_MODEL;  UNION;  SELECT 'RAM' AS item\_type, producer, model, quantity FROM ram WHERE model = INPUT\_MODEL |

11. ***Распечатанный лист заказа***

****

|  |  |
| --- | --- |
| report | INPUT\_NAME = thisform.text1.value  INPUT\_ORDER = thisform.text2.text  SELECT o.o\_id AS order\_id, c.name AS client\_name, w.name AS worker\_name, ;  "PU" AS type\_pu, pu.producer AS pu\_producer, pu.model AS pu\_model, pu.price AS pu\_price, ;  "GPU" AS type\_gpu, gpu.producer AS gpu\_producer, gpu.model AS gpu\_model, gpu.price AS gpu\_price, ;  "CPU" AS type\_cpu, cpu.producer AS cpu\_producer,cpu.model AS cpu\_model, cpu.price AS cpu\_price, ;  "RAM" AS type\_ram, ram.producer AS ram\_producer,ram.model AS ram\_model, ram.price AS ram\_price, ;  "TOTAL\_PRICE" AS total\_price\_col, ;  pu.price + gpu.price + cpu.price + ram.price AS total\_price;  FROM clients c, orders o, workers w, model m, pu, gpu, cpu, ram INTO CURSOR aaa;  WHERE o.c\_id = c.c\_id AND o.w\_id = w.w\_id AND o.m\_id = m.m\_id;  AND m.pu\_id = pu.pu\_id AND m.gpu\_id = gpu.gpu\_id AND m.cpu\_id = cpu.cpu\_id AND m.ram\_id = ram.ram\_id;  AND c.name = INPUT\_NAME AND o.o\_id = CAST(INPUT\_ORDER AS INT);  REPORT FORM "report1.frx" preview  release thisform |