

# Лекція № 9

## Сервер InterBase

### Навчальні питання

1. Загальні відомості. Склад сервера InterBase
2. Багатоверсійна архітектура сервера InterBase
3. Типи даних InterBase. Склад БД InterBase
4. Інсталяція, створення об'єктів БД

### ЛІТЕРАТУРА:

1. Ярцев В.П. Створення та обробка баз даних на ПЕОМ. Учебний посібник. ДУІКТ, 2003.- с.130-139.
2. Ковягин А., Востриков, Мир Inter Base/FireBird/Yaffil-М.: КУДИЦ-ОБРАЗ, 2002.-344-358.
3. Бондарь А.Г. Inter Base и FireBird. Практическое руководство для умных пользователей и начинающих разработчиков. СПб. БХВ – Петербург. 2012.- 592с

Сервер ***InterBase*** призначен для збереження та обробки великих обсягів інформації при одночасній роботі з БД багатьох клієнтів.

### **Відмінні риси серверу:**

- кросплатформений продукт**, який сумісний з різними типами операційних систем, включаючи *Windows 7*, *Windows XP*, *Linux*;
- низькі системні вимоги**;
- висока продуктивність** і легкість адміністрування;
- версійна архітектура**, що забезпечує роботу великої кількості клієнтів — пишучі користувачі не блокують читаючих користувачів;
- дуже висока надійність** і стійкість роботи з даними.

- *InterBase* розроблена початку 80-х років американською компанією *DEC*.
- Пізніше розробка *InterBase* велася фірмою *InterBase Software*, що згодом увійшла до складу компанії *Borland, Code Gear*.
- В даний час застосовуються версії програмного забезпечення *InterBase 2009 Developer Edition*.
- На основі версії 6.0, яка є відкритим, вільно розповсюджуваним продуктом, розроблено кілька клонів *Firebird 2.x* і *Yaffil*.
- *InterBase* зручний для навчальних цілей, його локальна версія поширюється з пакетом *Delphi*.

# Склад сервера InterBase

складається з двох частин: клієнтської та серверної

Для роботи з *InterBase* на кожній машині повинний бути встановлений клієнт *InterBase*.

Основною його частиною є бібліотека ***Gds32.dll***.

На сервері БД встановлюється і *клієнт*, і *сервер InterBase*.

Основний модуль сервера *InterBase* - файл ***ibserver.exe***.

Всі об'єкти бази даних зберігаються в одному файлі.

Файли баз даних *InterBase* мають розширення ***.GDB***.

Файли резервних копій мають розширення ***.GBK***.

Файл бази даних *InterBase* має **сторінкову** організацію.

Дані у файлі БД фізично розділені на сторінки, розмір яких визначається при створенні БД (1 Кбайт).

Якщо таблиці БД містять більш 200 тис. записів, є сенс збільшити розмір сторінки з метою збільшення продуктивності.

Як правило, розмір сторінки повинний бути дорівнювати розміру кластера жорсткого диска, на якому розміщений файл БД (4 Кбайт).

# Багатоверсійна архітектура *InterBase*

Сутність багатоверсійної архітектури проста і полягає в наступному:

- кожній транзакції дати свою версію запису, читаючий користувач не заважає пишущому.
- Усі зміни даних у записах виробляються не в самих записах, а в їхніх копіях.
- користувач підключився до БД і запустив транзакцію, у якій робляться якісь зміни в даних, для нього створюється своя версія записів, у яких повинні робитися зміни. Ця версія стає власністю даної транзакції, і всі операції в ній будуть вироблятися з цією версією.

- Якщо транзакція підтвердилася, то колишня (вихідна) версія запису буде позначена як видалена. Нова версія запису позначається як основна.
- Саме цю версію запису побачать інші транзакції, що будуть запуснені пізніше.
- Якщо відбудеться відкат транзакції, то нова версія буде відзначена як видалена, і основною версією залишиться колишня версія запису.
- Для всіх інших користувачів нічого не зміниться.

- Якщо, після запуску першої транзакції до її завершення запускається друга транзакція, що намагається прочитати той же запис, що змінює перша транзакція то друга транзакція буде бачити попередню версію запису.
- Якщо дві транзакції будуть намагатися змінити той самий запис, то виникає конфлікт, розв'язування якого виробляється з використанням механізму блокувань.



- При багатOVERсiйнiй архiтектурi постійно накопичуються застарілі версії записів, що називають **“сміттям”** та підлягають видаленню. Процес видалення застарілих версій записів називають **збиранням сміття**.
- Збирання сміття виробляється щораз, коли яка-небудь транзакція побажає прочитати даний запис.
- Ця транзакція зчитує всі існуючі версії даного запису і застарілі версії, що не використовуються в даний момент іншими транзакціями та видаляє.

# Типи даних у InterBase

- для цілих чисел: **INTEGER, SMALLINT**;
- для дійсних чисел: **FLOAT, DOUBLE PRECISION**;
- для чисел з фіксованим записом: **NUMERIC, DECIMAL**;
- для збереження дати, часу, дати/часу: **DATE, TIME, TIMESTAMP**;
- для збереження символів: **CHARACTER (CHAR), VARYING CHARACTER (VARCHAR)**;
- для збереження довільних динамічно розширюваних даних: **BLOB**.

- **SMALLINT** (2 байти) – цілі числа в діапазоні від  $-32768$  до  $+32767$ ;
- **INTEGER** (4 байти) – цілі числа в діапазоні від  $-2\,147\,483\,648$  до  $+2\,147\,483\,647$ ;
- **FLOAT** (4 байти) – дійсні числа до 7 значущих цифр у діапазоні від  $3.4 \cdot 10^{-38}$  до  $3.4 \cdot 10^{+38}$ ;
- **DOUBLE PRECISION** (8 байт) – дійсні числа до 15 значущих цифр у діапазоні від  $1.7 \cdot 10^{-308}$  до  $1.7 \cdot 10^{+308}$ ;
- **NUMERIC** чи **DECIMAL** (перем.) – дійсні числа з фіксованою крапкою.

Кількість значущих цифр і кількість цифр після крапки вказується при визначенні типу даних;

- **CHAR[ASTER](n)** (0-32767 байт) – текстовий стовпець довжиною до n символів;
- **CHAR[ASTER](n) VARYING** (0-32767 байт) – текстовий стовпець перемінної довжини, що містить до n символів;
- **DATE** (8 байт) – дата в межах від 01.01.0100 до 11.12.5941. Можуть також зберігатися дані про час;
- **BLOB** (перем.) – будь-які двійкові дані, наприклад, малюнки, музичні файли й ін.

- У *InterBase* не визначені типи даних “логічний” і “автоінкрементний” (лічильник).
- Логічний тип замінюється типом CHAR(1),
- Для автоінкрементного типу спільно використовуються *генератори* і *тригери*, що забезпечують одержання унікальних значень при введенні даних.
- При вставці запису тригер перевіряє значення цього поля, і якщо воно виявиться порожнім, вставляє в нього значення генератора, при цьому значення генератора збільшується на одиницю

## Склад БД *InterBase*

- таблиці (Tables);
- індекси (Indexes);
- обмеження (Constraints);
- домени (Domains);
- перегляди (Views);
- генератори (Generators);
- тригери (Triggers);
- збережені процедури (Stored Procedures);
- привілеї (Roles).

**Таблиці** – це, як і раніше, основні елементи структури БД, призначені для збереження даних. Зв'язки між таблицями, засновані на використанні первинних і зовнішніх ключів, забезпечують високу безпеку і цілісність даних.

**Індекси** – як і в локальних БД, це додаткові структури даних, що забезпечують високу швидкість обробки даних.

**Обмеження** – це спеціальні засоби, що дозволяють автоматизувати процес уведення даних, запобігати помилок введення, керувати порядком сортування записів.

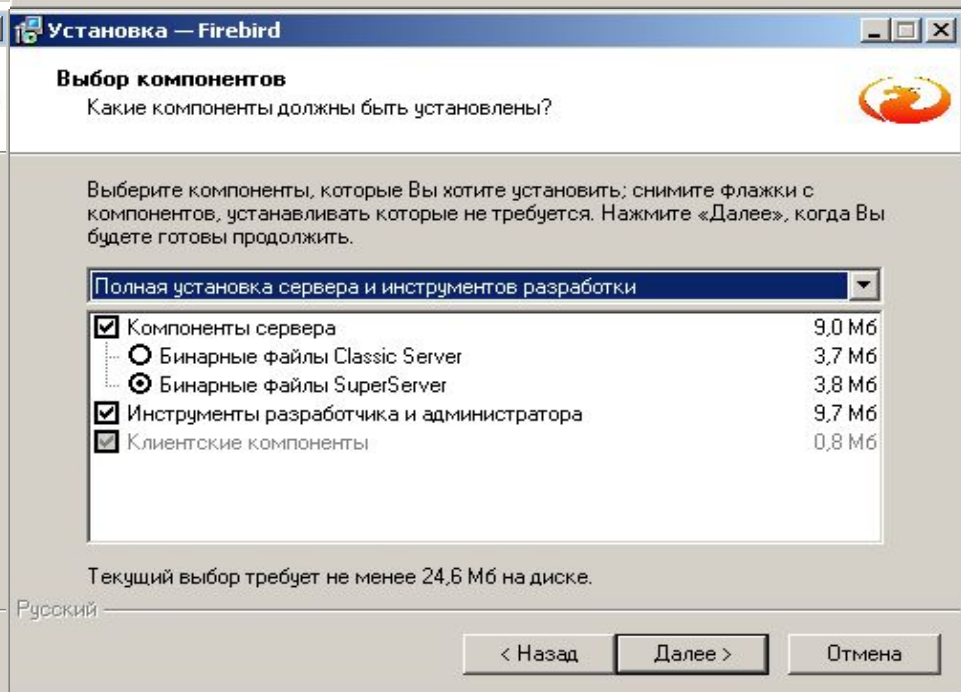
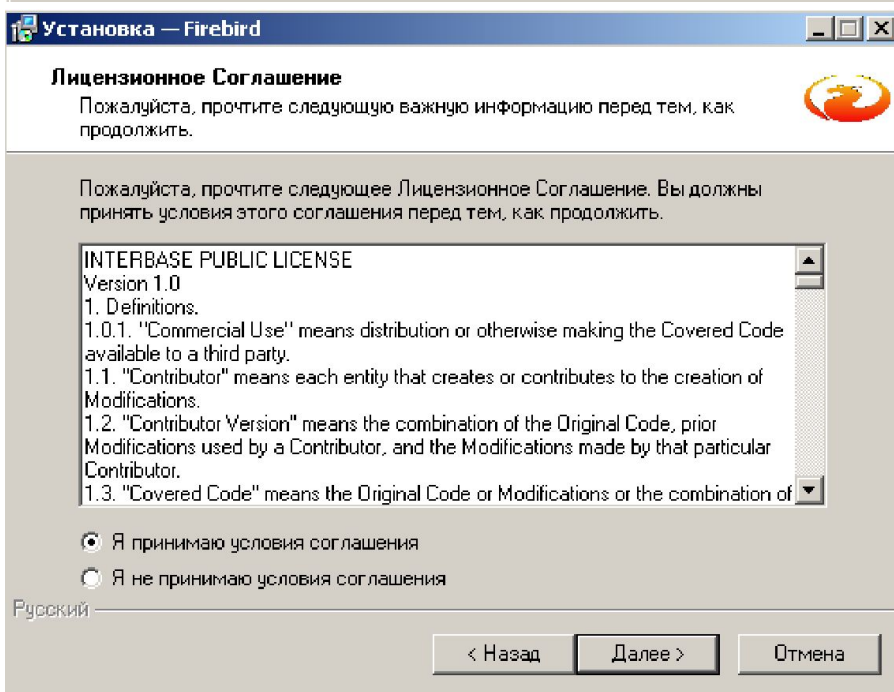
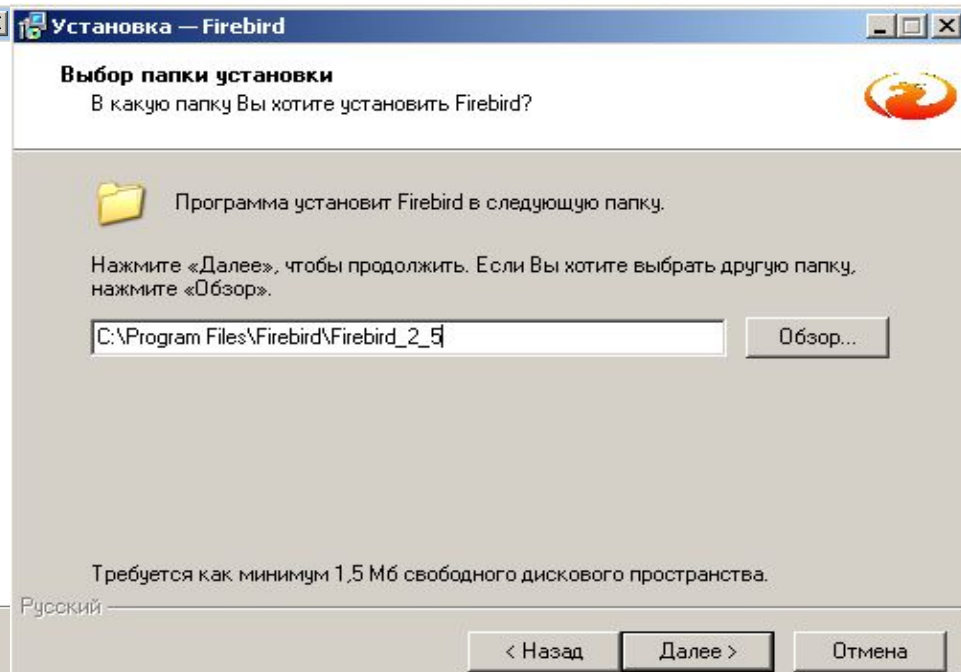
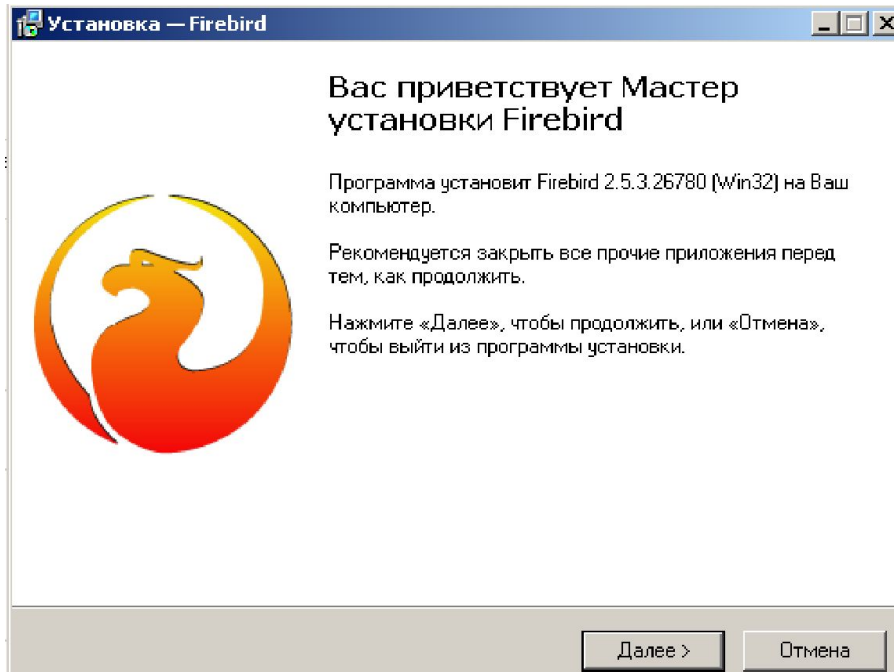
- **Домен** – це іменований опис стовпця. Його можна розглядати як свого роду шаблон опису стовпця. Один раз визначивши домен, його ім'я можна потім можна багаторазове використовувати для створення інших стовпців (у тому числі і для різних таблиць).
- **Перегляд (Представлення)** – це віртуальна таблиця, записи в який відібрані за допомогою оператора SELECT. Перевага перегляду полягає в тому, що створивши його, у наступному можна використовувати його безпосередньо без повторних звертань до оператора SELECT.
- **Генератор** – це засіб для одержання унікальних цілих значень. Призначений для одержання унікальних значень у ключових полях автоінкрементного типу даних.



- **Збережена процедура** – це підпрограма, розташована на сервері і викликувана з програми клієнта. Використання збережених процедур збільшує швидкість обробки даних. Їхня перевага також полягає в тому, що вони є загальними для всіх клієнтських додатків, завдяки чому різко скорочується число помилок, внесених у дані.
- **Тригер** – це збережена процедура, що викликається автоматично при модифікації записів у БД, тобто при зміні, видаленні і додаванні записів. Тригери від звичайних збережених процедур відрізняються тим, що їх не можна викликати з додатка клієнта, передавати їм параметри й одержувати від них

- **Привілегії** – це спеціальні структури даних, у яких зберігається інформація про права доступу для кожного з зареєстрованих користувачів. Після створення об'єкта (наприклад, таблиці) доступ до нього має тільки його творець. Для надання можливості доступу до даних інших користувачів для них потрібно призначити відповідні привілеї. Передбачено такі рівні привілеїв:

- ALL – усі права доступу;
- SELECT – тільки читання;
- DELETE – видалення;
- INSERT – вставка;
- UPDATE – модифікація.



## Выберите дополнительные задачи

Какие дополнительные задачи необходимо выполнить?



Выберите дополнительные задачи, которые должны выполняться при установке Firebird, после этого нажмите «Далее»:

☒ Использовать Guardian для управления сервером

Способ запуска сервера Firebird:

☐ Запускать в качестве Приложения

☒ Запускать в качестве Службы

☒ Автоматически запускать Firebird при к

☒ Установить Апплет Панели Управлени

☐ Копировать клиентскую библиотеку Fir

☒ Создать GDS32.DLL для поддержки ун

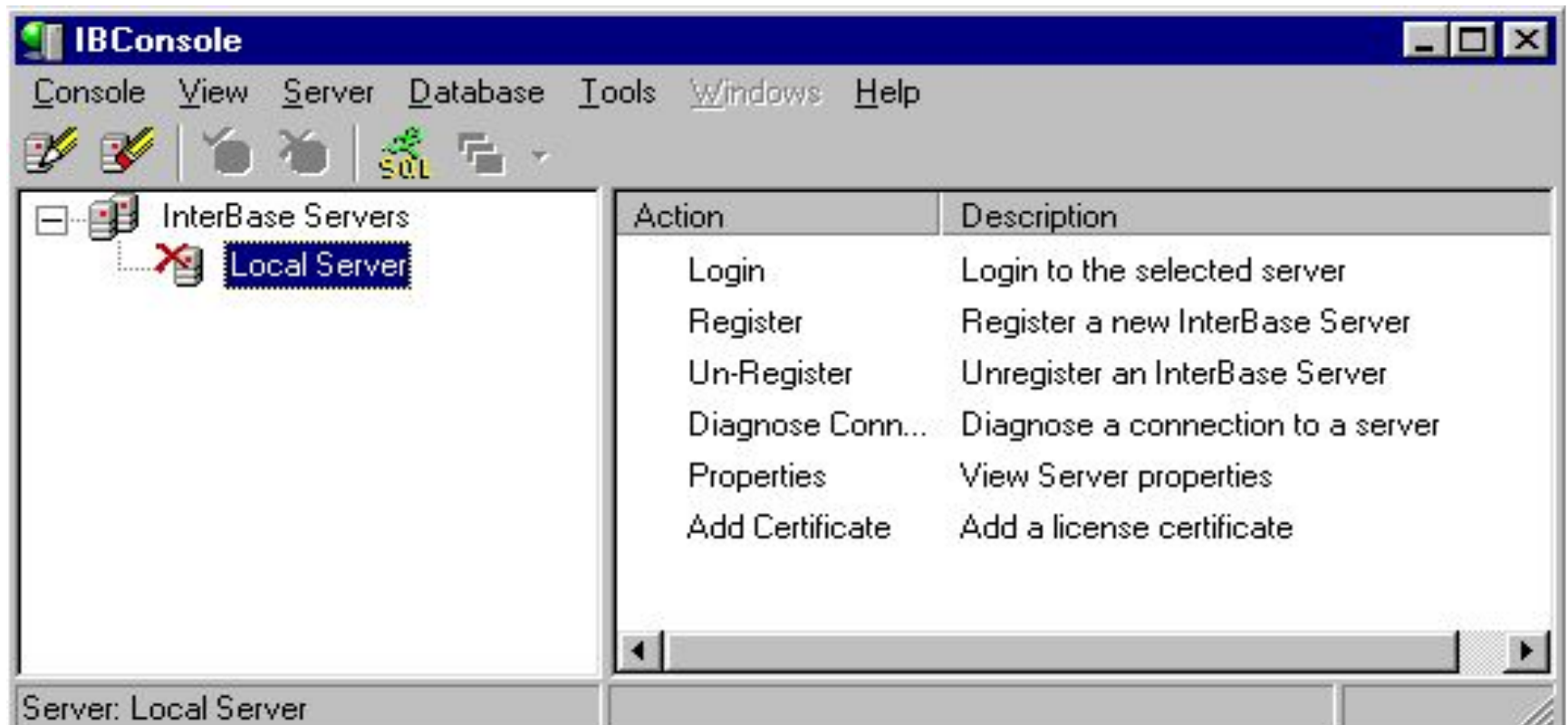
Русский



# Програма IBConsole

Забезпечує:

- Керування локальним і віддаленим сервером;
- Керування БД;
- Інтерактивне виконання *SQL*-запитів.



Панель управления

ФайлПравкаВидИзбранноеСервисСправка

Назад

Мой компьютерПанель управления

См. также

Windows UpdateСправка и поддержка

BDE Administrator

DirectX

Firebird Server Control

Flash Player

Java

Nero BurnR

Nokia Conn

NVIDIA nVie

Realtek HD

TweakUI дл

Windows Co

Автозагру

Автоматич

Администр

Анализ вид

Анализ и п

Анализ про

АнтиAutorun

Брандмауэр Windows

Дата и время

Диспетчер плоттеров Autodesk

Диспетчер стилей печати Autodesk

Звуки и аудиоустройства

Игровые устройства

Инструмент BootSave

Инструмент Hosts Editor

Инструмент Microsoft TimeZone

Инструмент MS AutoPlay

Инструмент Nero Info Tool

Клавиатура

Учётные записи W

Учетные записи п

Центр обеспечени

Центр установлен

Шрифты

Экран

Электропитание

Язык и региональ

Firebird Server Control

The Firebird service is running.

Stop

Version 2.5.3.26780 Firebird 2.5

☒ Use the Guardian

Run

☒ as a Service

☐ as an application

Start

☒ Automatically

☐ Manually

OK

Cancel

Apply

Сервер БД

Свойства обозревателя

Свойства папки

Сетевые подключения

Система

Сканеры и камеры

Службы и устройства

Специальные возможности

Телефон и модем

Тестирование оборудования

Установка и удаление программ

Установка оборудования

Объектов: 64



C:\Program Files\Firebird\Firebird\_2\_5\bin\isql.exe

Use CONNECT or CREATE DATABASE to specify a database

SQL> set sql dialect 3;create database 'C:\pr\_FB2\_5.fdb' user 'sysdba' password  
'masterkey' default character set WIN1251; exit;\_

# Керування сервером

- реєстрація сервера;
- підключення сервера;
- перегляд протоколу роботи;
- керування сертифікатами;
- визначення користувачів.

**SYSDBA**- ім'я адміністратора БД

**masterkey** – пароль





InterBase Servers

- Local Server
  - Databases
  - Backup
  - Server Log
  - Users

Action	Description
Logout	Logout from the current server
User Security	Manage InterBase Users
Properties	View Server properties
Add Certificate	Add a license certificate
Remove Certificate	Remove a license certificate

# Керування базами даних

- реєстрація БД;
- підключення до БД;
- створення і видалення БД;
- перегляд метаданих;
- збір сміття;
- перевірка стану БД;
- аналіз статистики;
- збереження і відновлення БД.

Для створення БД необхідно виконати команду меню **Database | Create Database...**

**Create Database**

Server: Local Server

File(s):

Filename(s)	Size (Pages)
c:\COБД\AD21\Lab9\UspStud_AD21.GDB	100

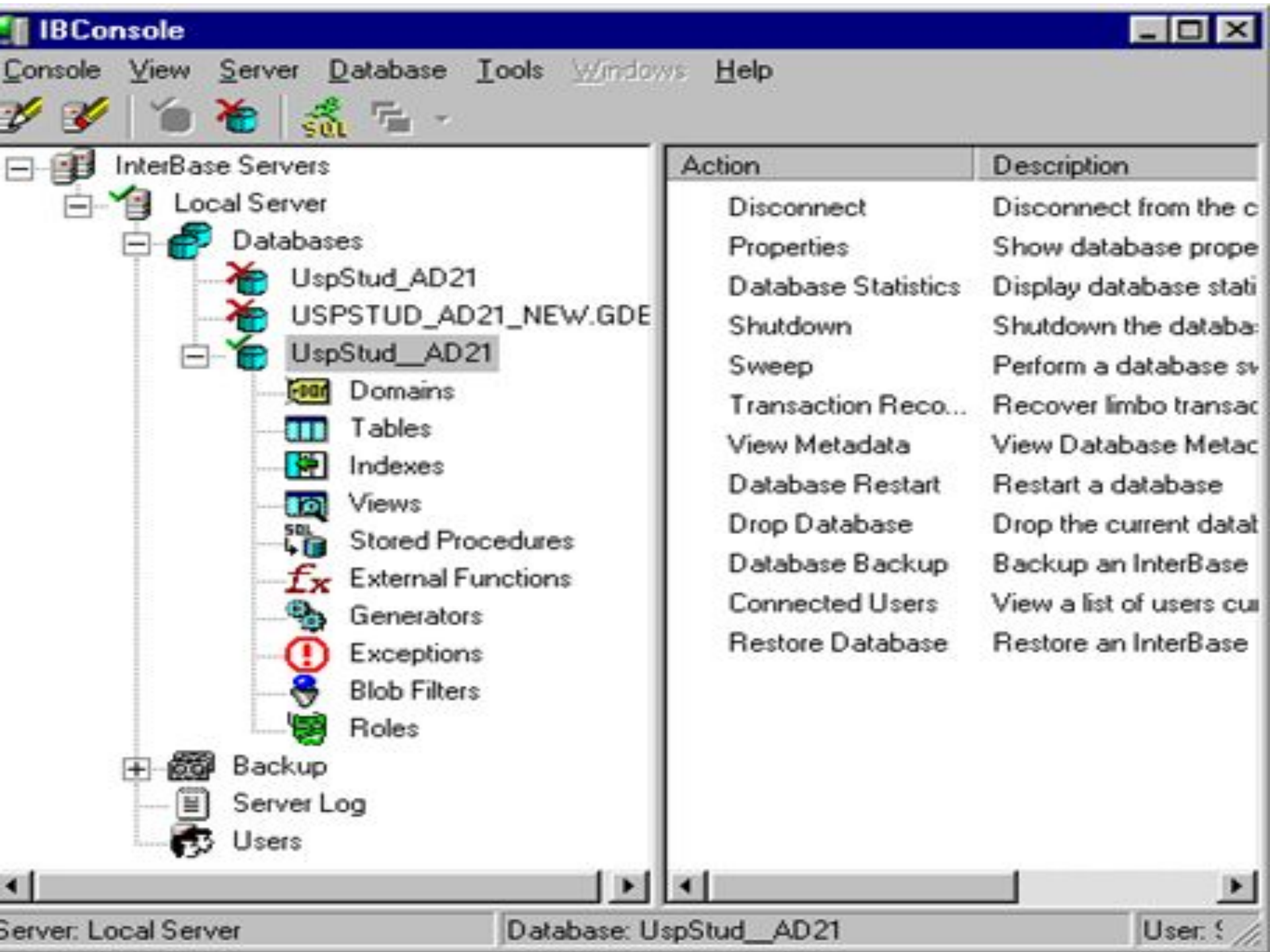
Options:

Page Size	4096
Default Character Set	WIN1251
SQL Dialect	3

☒ Register database

Alias: UspStud\_AD21

OK Cancel



IBConsole																											
Console View Server Database Tools Windows Help																											
<div><div>InterBase Servers</div><div><div>Local Server</div><div><div>Databases</div><div><div>UspStud_AD21</div><div>USPSTUD_AD21_NEW.GDE</div><div>UspStud_AD21</div><div>Domains</div><div>Tables</div><div>Indexes</div><div>Views</div><div>Stored Procedures</div><div>External Functions</div><div>Generators</div><div>Exceptions</div><div>Blob Filters</div><div>Roles</div></div><div>Backup</div><div>Server Log</div><div>Users</div></div></div></div> <div><table><thead><tr><th>Action</th><th>Description</th></tr></thead><tbody><tr><td>Disconnect</td><td>Disconnect from the c</td></tr><tr><td>Properties</td><td>Show database prope</td></tr><tr><td>Database Statistics</td><td>Display database stati</td></tr><tr><td>Shutdown</td><td>Shutdown the databa:</td></tr><tr><td>Sweep</td><td>Perform a database sv</td></tr><tr><td>Transaction Reco...</td><td>Recover limbo transac</td></tr><tr><td>View Metadata</td><td>View Database Metac</td></tr><tr><td>Database Restart</td><td>Restart a database</td></tr><tr><td>Drop Database</td><td>Drop the current datat</td></tr><tr><td>Database Backup</td><td>Backup an InterBase</td></tr><tr><td>Connected Users</td><td>View a list of users cur</td></tr><tr><td>Restore Database</td><td>Restore an InterBase</td></tr></tbody></table></div>		Action	Description	Disconnect	Disconnect from the c	Properties	Show database prope	Database Statistics	Display database stati	Shutdown	Shutdown the databa:	Sweep	Perform a database sv	Transaction Reco...	Recover limbo transac	View Metadata	View Database Metac	Database Restart	Restart a database	Drop Database	Drop the current datat	Database Backup	Backup an InterBase	Connected Users	View a list of users cur	Restore Database	Restore an InterBase
Action	Description																										
Disconnect	Disconnect from the c																										
Properties	Show database prope																										
Database Statistics	Display database stati																										
Shutdown	Shutdown the databa:																										
Sweep	Perform a database sv																										
Transaction Reco...	Recover limbo transac																										
View Metadata	View Database Metac																										
Database Restart	Restart a database																										
Drop Database	Drop the current datat																										
Database Backup	Backup an InterBase																										
Connected Users	View a list of users cur																										
Restore Database	Restore an InterBase																										
Server: Local Server Database: UspStud_AD21 User: ?																											

# Інтерактивне виконання SQL-інструкцій

***create table STUDENT***

```
(    KodStud smallint not Null Primary Key,  
    Fam char(15) not Null,  
    Nam1 char(15),    Nam2 char(15),  
    God_r smallint,  
    NumStudBil char(10));
```

***create table USP***

```
(    KodStud smallint not Null,  
    KodPred smallint not Null,  
    Dat timestamp not Null,  
    Ocenka smallint not Null,  
    Primary Key(KodStud, KodPred, Dat),  
    ForeIGN Key(KodStud) references STUDENT,  
    ForeIGN Key(KodPred) references PREDMET);
```



```
create table STUDENT
(
  KodS char(15) not Null Primary Key,
  Fam char(15) not Null,
  Nam1 char(15),
  Nam2 char(15),
  God_r smallint,
  NumStudBil char(10)
);
```

8: 1

Modified

Client dialect 3

No active transaction.

AutoDDL: ON

Data Plan Statistics


- Додаємо в таблицю "Grupa" поле для визначення факультету :

```
ALTER TABLE Grupa  
ADD COLUMN Fakultet char(20);
```

- Видаляємо поле для визначення факультету:

```
ALTER TABLE Grupa DROP COLUMN Fakultet;
```

- Віддаляється таблиця Grupa.

```
DROP TABLE Grupa;
```

Створення первинного індексу по номеру замовника:

```
CREATE INDEX NameGr ON Grupa (NameGr) WITH  
PRIMARY;
```

```
DROP INDEX NameGr ON Grupa ;
```

Для редагування, додавання нових і видалення записів призначені оператори

UPDATE, INSERT і DELETE.

Оператор редагування даних UPDATE має формат:

UPDATE ім'я таблиці  
SET ім'я полю = вираження,  
[WHERE умова відбору];

Приклад:

```
UPDATE Student  
SET GodRog = 1996  
WHERE Fam = "Іванов"
```