

Підручник з Kexi

Цю документацію було створено на основі сторінки
UserBase KDE Kexi/Handbook 14 вересня 2012 року.
Оновлено до версії 2.4 командою з документування KDE
Переклад українською: Юрій Чорноіван



Підручник з Кехі

Зміст

1	Вступ	8
2	Основи Кехі	9
2.1	Бази даних Кехі	9
2.2	Створення файлу нової бази даних	10
2.3	Головне вікно Кехі	12
2.3.1	Основні елементи вікна програми	12
2.3.2	Панель вкладок	13
2.3.3	Панель навігації проектом	13
2.3.4	Панель відкритих об'єктів бази даних або панель з вкладками вікон	14
2.3.5	Панель редактора властивостей	14
2.4	Відкриття існуючого файлу бази даних Кехі	15
2.4.1	Відкриття файлу бази даних за допомогою діалогового вікна Відкрити проект	15
2.4.2	Як відкрити вже створений файл бази даних Кехі натисканням піктограми файлу .kexi	16
2.5	Користування вбудованою довідкою	17
3	Побудова простої бази даних	18
3.1	Вступ	18
3.2	Розробка таблиць бази даних	19
3.2.1	Вікно створення таблиць	19
3.2.1.1	Вікно створення таблиць складається з таких стовпчиків:	19
3.2.1.2	Розробка таблиці «Особи»	19
3.3	Введення даних до таблиць	20
3.3.1	Докладно про доступні дії під час введення даних до таблиць	21
3.4	Розробка запитів до баз даних	21
3.5	Розробка форм	22
3.5.1	Найважливіші терміни	22
3.5.2	Порівняння форм з таблицями	23
3.5.3	Робота з компонуванням форм	23
3.5.4	Користування вкладкою «Віджети»	24
3.5.5	Додавання віджетів — текстові поля	24
3.5.6	Визначення джерел даних	25

Підручник з Кехі

3.5.7	Додавання текстових міток	26
3.5.8	Дії	26
3.5.8.1	Призначення дій до кнопок форми	26
3.5.9	Компонування віджетів	27
3.5.9.1	Правила розмірів для віджетів у компонуванні	27
3.5.9.2	Величини правил розмірів	28
3.5.9.3	Вертикальне та горизонтальне розтягнення	29
3.6	Введення даних за допомогою форм	29
4	Як налаштувати Кехі	30
4.1	Швартування і від'єднання бічних панелей	30
5	Довідка щодо команд	31
5.1	Вкладка «Кехі»	31
5.2	Вкладка «Створення»	32
5.3	Вкладка «Дані»	33
5.4	Вкладка «Зовнішні дані»	33
5.5	Вкладка «Інструменти»	34
5.6	Вкладка «Компонування форм»	34
5.7	Вкладка «Компонування звіту»	35
6	Додаток А. Вступ до баз даних	37
6.1	Що таке база даних?	37
6.2	База даних і електронна таблиця	38
6.2.1	Чим база даних відрізняється від електронної таблиці?	38
6.2.2	Цілісність довідкових даних	39
6.2.3	Резервування даних	39
6.2.4	Цілісність і коректність даних	39
6.2.5	Обмеження перегляду даних	40
6.2.6	Швидкодія і місткість	41
6.2.7	Запис даних	41
6.2.8	Звіти	41
6.2.9	Програмування	41
6.2.10	Доступність для декількох користувачів	42
6.2.11	Безпека	42
6.3	Компонування бази даних	42
6.4	Кому потрібні бази даних?	42
6.4.1	Зверніться до електронних таблиць, якщо:	42
6.4.2	Використовуйте бази даних, якщо:	43
6.5	Програмне забезпечення для створення баз даних	43
7	Додаток Б. Порівняння Кехі з іншими програмами для роботи з базами даних	44
7.1	Типи даних	44

8	Додаток В. Зарезервовані слова SQL	46
8.1	Зарезервовані слова Кехі для SQL	46
8.2	Зарезервовані слова Кехі для SQLite	48
8.3	Зарезервовані слова Кехі для MySQL	48
8.4	Зарезервовані слова Кехі для PostgreSQL	57
8.5	Зарезервовані слова Кехі для драйвера Oracle	62
8.6	Зарезервовані слова Кехі для драйвера Sybase	67
8.7	Зарезервовані слова Кехі для драйвера xBase	75
9	Додаток Г. Підтримувані формати файлів	76
9.1	Формат значень, відокремлених комами (CSV)	76
9.2	Формат файлів Microsoft Access (MDB)	76
9.2.1	Огляд	76
9.2.2	Можливості	77
9.2.3	Підтримувані можливості	77
9.2.4	Непідтримувані можливості	77
10	Подяки і ліцензія	78

Перелік таблиць

6.1	Таблиця контактів	37
6.2	Таблиця контактів	38
6.3	Таблиця осіб	39
6.4	Таблиця осіб	40
6.5	Таблиця осіб	40
7.1	Порівняння типів даних у Кехі та інших програмах для роботи з базами даних	44

Анотація

Kexi — це програма для створення баз даних і керування даними у комплексі програм Calligra.

Розділ 1

Вступ

Кехі — це програма для керування базами даних. Нею можна скористатися для створення баз даних, додавання даних, виконання запитів і обробки даних. Передбачено створення форм, за допомогою яких ви можете отримувати доступ до ваших даних. Всі об'єкти бази даних (таблиці, запити і форми) зберігаються у базі даних, що спрощує поширення даних та дизайну бази даних.

Кехі є частиною комплексу офісних програм Calligra.

Окрім зберігання ваших баз даних Кехі у файлах, Кехі також здатна зберігати ваші дані на сервері баз даних. За допомогою сервера баз даних ви можете спільно використовувати ваші бази даних з іншими користувачами. Крім того, сервер баз даних надає змогу декільком користувачам одночасно користуватися однією базою даних. Кехі підтримує такі сервери баз даних:

- [MySQL](#)
- [PostgreSQL](#)

Докладніші дані щодо Кехі можна знайти на [сторінці Кехі на сайті Calligra](#) та на [власному сайті Кехі](#).

Якщо у вас виникли якісь питання щодо Кехі, ви можете скористатися одним зі списків листування програми. Список листування користувачів Кехі призначено для питань щодо користування Кехі та самого проєкту Кехі. Список листування розробників Кехі використовується для обговорення питань, пов'язаних з розробкою Кехі. Подальші настанови щодо оформлення підписки на ці списки листування, а також обговорення інших способів встановлення зв'язку з розробниками Кехі можна знайти [тут](#).

ПРИМІТКА

Цей підручник з Кехі 2.5 засновано на підручнику з Кехі 1.1. Будь ласка, погоджуйте всі його редагування з Jarosław Staniek kexi@kde.org

Розділ 2

Основи Кехі

- Бази даних Кехі
- Створення файла нової бази даних
- Головне вікно Кехі
 - Основні елементи вікна програми
- Відкриття існуючого файла бази даних Кехі
 - Відкриття файла бази даних за допомогою діалогового вікна **Відкрити проект**
 - Як відкрити вже створений файл бази даних Кехі натисканням піктограми файла .kexi
- Користування вбудованою довідкою

2.1 Бази даних Кехі

Багато програм, зокрема OpenOffice.org[®], LibreOffice[®] та Microsoft[®] Excel створюють файли, які називають документами. Кехі також створює файли, але ці файли ми будемо називати файлами баз даних Кехі або просто файлами баз даних.

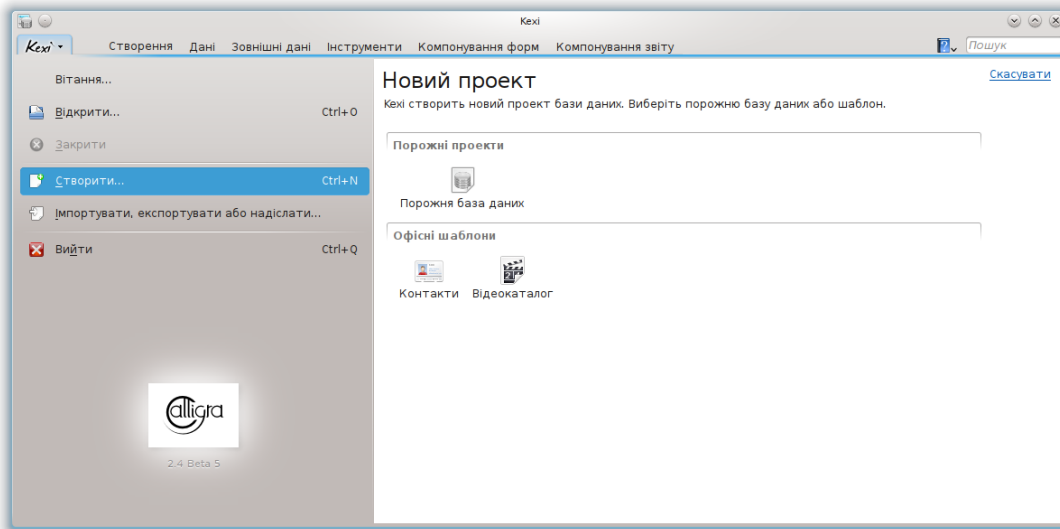
Окрім зберігання ваших баз даних у файлах баз даних, Кехі може використовувати бази даних на серверах баз даних, ось чому ми будемо називати файли саме *файлами баз даних*, а не просто *базами даних*.

Термін проект Кехі або просто проект також використовується на позначення бази даних Кехі, незалежно від того, зберігається ця база даних у файлі чи на сервері бази даних.

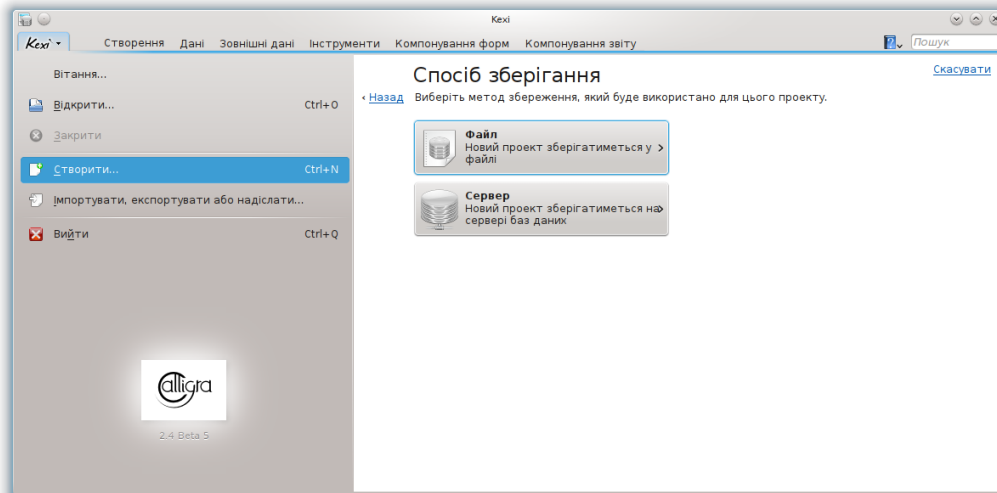
ПРИМІТКА

Зазвичай, файли баз даних Кехі мають суфікс назви .kexi.

2.2 Створення файлу нової бази даних

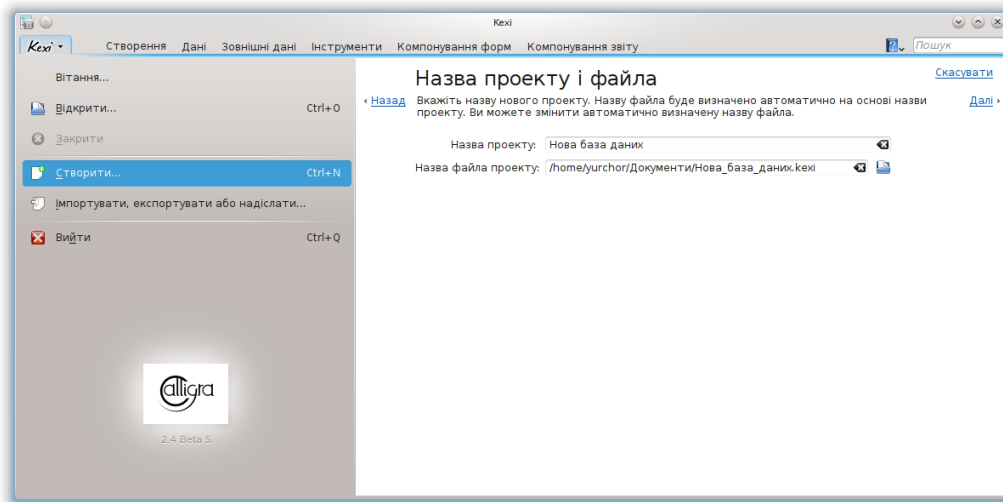


1. Запустіть Kexi, якщо програми ще не запущено, і скористайтеся пунктом меню **Kexi** → **New...** (**Ctrl+N**).
2. На сторінці **Новий проект** у розділі **Порожні проекти** виберіть пункт **Порожня база даних**.



3. На сторінці **Спосіб зберігання** натисніть кнопку **Файл**.

Підручник з Kexi

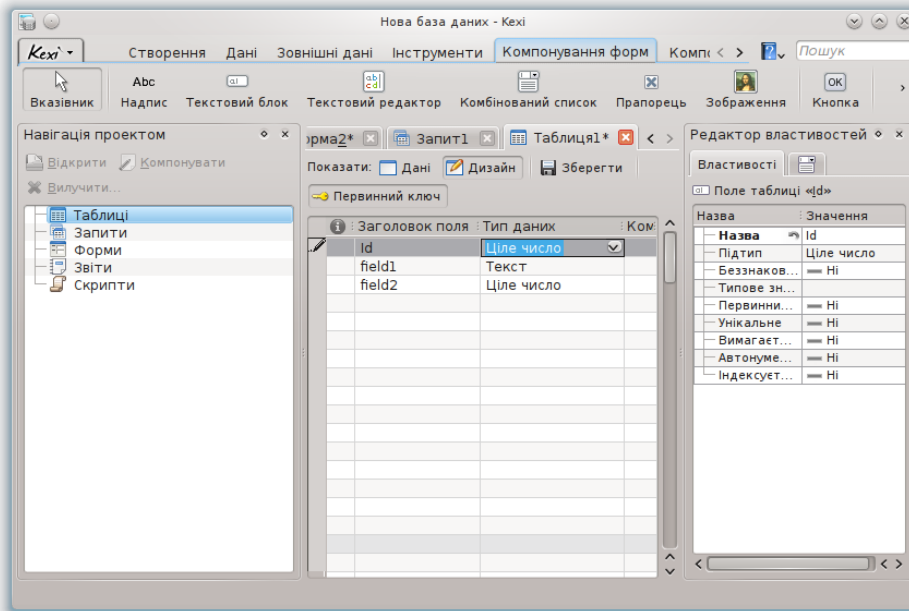


4. На сторінці **Назва проекту і файла** вкажіть назву проекту і файла, у якому зберігатиметься цей проект.
5. Натисніть кнопку **Далі**, щоб створити проект.

ПРИМІТКА

- У разі зміни назви проекту запропоновану назву файла буде також змінено відповідним чином.
- Ви можете скористатися засобом перегляду ієрархії файлів для вибору теки, у якій слід зберегти ваш файл бази даних.

2.3 Головне вікно Kexi



За допомогою **панелі вкладок**, розташованої у верхній частині вікна ви зможете отримати доступ до типових дій та пунктів меню.

По обидва боки від дочірнього вікна буде показано панелі **Навігація проектом** та **Редактора властивостей**. За потреби, ці ви можете змінити розміри цих панелей або наказати програмі приховати їх. Приховати панель можна натисканням маленького хрестика у верхній частині панелі (одразу під панельлю інструментів).

Список об'єктів бази даних (таблиць, запитів тощо) наведено на панелі **Навігація проектом**. Об'єкт можна відкрити натисканням (подвійним клацанням кнопкою миші, залежно від загальних налаштувань KDE) пункту з його назвою.

2.3.1 Основні елементи вікна програми

Серед основних елементів вікна програми Kexi:

Панель вкладок

Містить пункти керування програмою. Докладний опис відповідних команд можна знайти у [додатку](#) до цього підручника.

Панель навігації проектом

Містить список всіх об'єктів (таблиць, запитів, форм...), створених у межах поточного відкритого проекту бази даних. На панелі навігації також міститься невеличка панель з кнопками найпоширеніших команд, пов'язаних з об'єктами бази даних.

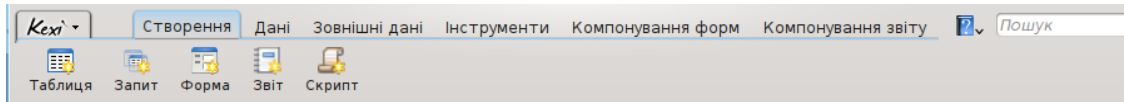
Панель об'єктів відкритої бази даних / Вікна у вкладках

Центральна частина вікна програми, яка займає більшість його площі. У інтерфейсі користувача передбачено перемикання між вкладками. Активну вкладку завжди розгорнуто на всю площу панелі.

Панель редактора властивостей

Містить список властивостей поточного активного об'єкта бази даних. Для деяких об'єктів (наприклад, віджетів форми) може містити декілька вкладок.

2.3.2 Панель вкладок



За допомогою панелі інструментів ви зможете отримати доступ до більшості команд та дій у Кехі.

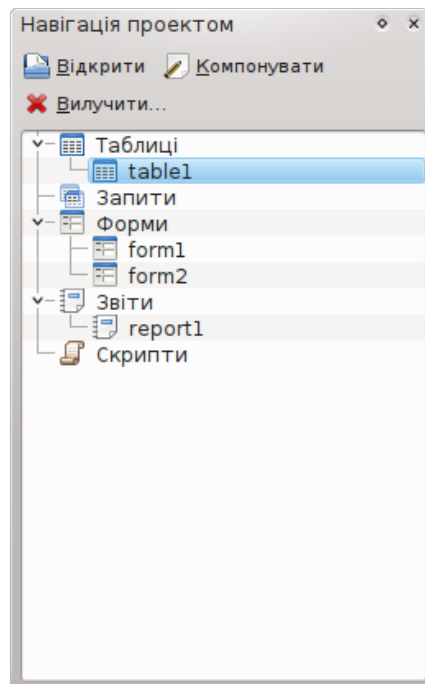
За допомогою кнопок дій, розташованих на панелях інструментів різних вкладок, ви зможете:

- Створювати, відкривати та закривати проекти Кехі.
- Створювати об'єкти бази даних.
- Імпортувати та експортувати дані.

Залежно від контексту, може бути показано додаткові вкладки:

- Вкладку **Компонування форм** буде показано, якщо ви користуєтеся засобом компонування форм.
- Вкладку **Компонування звіту** буде показано, якщо ви користуєтеся засобом компонування звітів.

2.3.3 Панель навігації проектом



Панель **Навігація проектом** є одним з найчастіше використовуваних елементів головного вікна Кехі. На панелі міститься список всіх об'єктів у межах відкритого проекту бази даних Кехі. Об'єкти поділено за групами: таблиці, запити, форми, звіти і скрипти.

На панелі Навігація проектом також розташовано невеличку панель інструментів з кнопками найчастіше виконуваних дій (зліва праворуч): **Відкрити** позначений об'єкт, **Компонувати** позначений об'єкт і **Вилучити** позначений об'єкт.

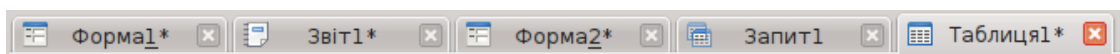
Доступ до контекстного меню кожного з об'єктів можна отримати за допомогою клацання правою кнопкою миші.

Наведення вказівника миші на пункт назви об'єкта у списку з наступним подвійним клацання лівою кнопкою миші надає змогу відкрити об'єкт на **панелі перегляду даних**. Якщо вікно об'єкта вже було відкрито, ця дія просто активує вікно без перемикання його у режим перегляду.


ПРИМІТКА

Навіть якщо вашу операційну систему або програму для керування вікнами налаштовано так, що одинарні клацання кнопками миші обробляються як подвійні, на панелі **Навігація проектом** Кехі все одно буде використано подвійні клацання, щоб запобігти випадковому відкриттю великих наборів даних або виконанню небажаних запитів.

2.3.4 Панель відкритих об'єктів бази даних або панель з вкладками вікон



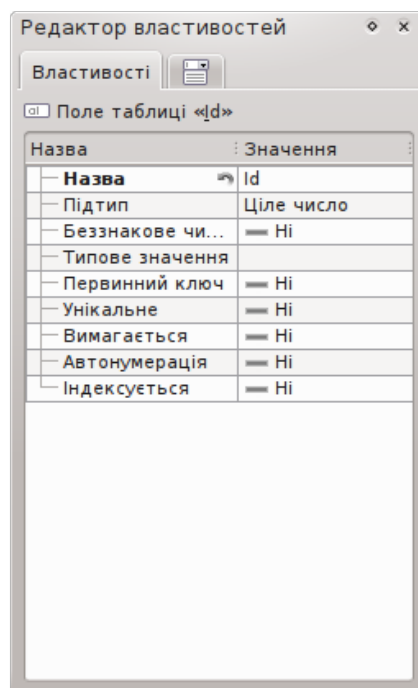
Після натискання пункту об'єкта на панелі навігації проектом цей об'єкт буде відкрито на **панелі відкритих об'єктів бази даних**. Кожному відкритому об'єкту у Кехі відповідає одна вкладка.

Перевпорядкувати вкладки можна перетягуванням зі скиданням вказівником миші. Закрити вкладку можна натисканням кнопки , розташованої поблизу правої межі смужки вкладок.

ПРИМІТКА

Можливо у пізніших версіях програми буде можливість від'єднувати вкладки з метою створення чогось подібного до багатовіконного інтерфейсу. Такий інтерфейс може бути корисним для нетипових робочих середовищ або середовищ з декількома дисплеями.

2.3.5 Панель редактора властивостей



За допомогою панелі **Редактора властивостей** ви можете змінити властивості об'єкта, показаного у активному вікні. Залежно від контексту, на цій панелі буде показано одну або декілька вкладок. Перша, завжди видима вкладка, **Властивості**, містить список можливих властивостей.

Правила використання редактора властивостей:

- У кожному з рядків міститься лише одна властивість.
- Ви можете використовувати мишу або клавіатуру для зміни значень окремих властивостей.
- Найчастіше використовуються значення таких властивостей:
 - число; ви можете ввести значення напряму або збільшити чи зменшити значення натисканням на кнопки зі стрілочками.
 - текст
 - спадний список значень
 - значення «Так/Ні»; переключити значення можна натисканням кнопки; «Так» (true) відповідає натиснутому стану кнопки, «Ні» (false) — ненатиснутому.

ПРИМІТКА

- Немає потреби у підтвердженні зміненого значення: зміни стануть видимими негайно після переходу до іншого рядка редактора властивостей або після натискання клавіші **Enter**.
- Назви нещодавно змінених властивостей, які ще не було збережено у бази даних, буде позначено напівжирним шрифтом.
- Після зміни значення властивості на бічній панелі списку редактора властивостей з'явиться особлива кнопка **Скасувати зміни**. Натисканням цієї кнопки ви можете повернути властивості початкове значення, яке було завантажено з бази даних під час відкриття об'єкта бази даних. Кнопка стане видимою, лише у разі позначення пункту властивості.

Панель редактора властивостей буде порожньою, якщо:

- не відкрито жодного об'єкта бази даних або
- у вікні активного об'єкта бази даних взагалі не показано властивості, — таке буває, якщо об'єкт відкрито у режимі перегляду даних, а не у режимі компонування

2.4 Відкриття існуючого файлу бази даних Кехі

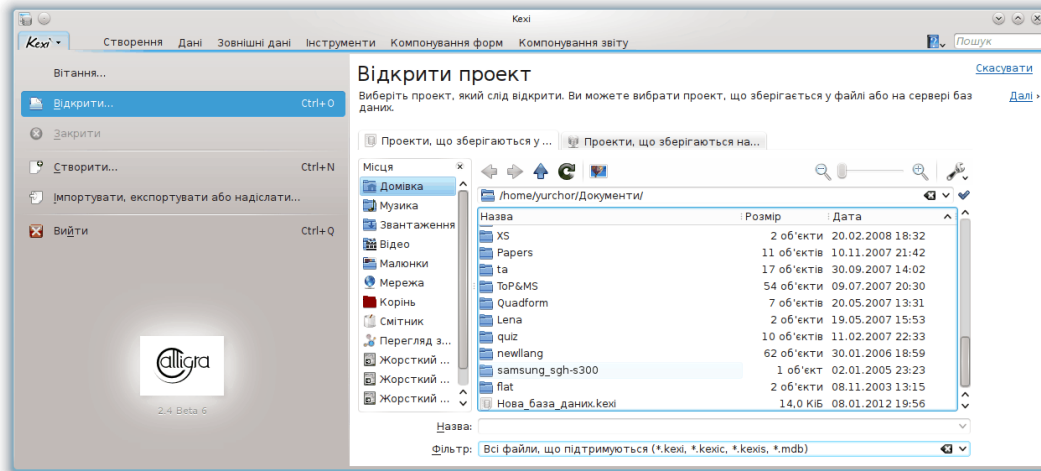
Передбачено два способи відкриття вже створеного файлу бази даних Кехі:

2.4.1 Відкриття файлу бази даних за допомогою діалогового вікна Відкрити проєкт

- Запустіть Кехі. Ви маєте побачити початкове діалогове вікно **Вітаємо у Кехі**. За допомогою цього вікна ви зможете вибрати один з попередніх проєктів, над якими ви працювали.

Підручник з Kexi

- Якщо пункту проекту немає на сторінці зі списком останніх використаних проектів, скористайтеся пунктом **Відкрити...** (**Ctrl+O**). У відповідь буде відкрито таке діалогове вікно:



- На панелі адреси виберіть теку, у якій міститься файл, який вам потрібен.
- Ви можете натиснути пункт файла, і програма негайно його відкриє. Крім того, ви можете вказати його назву у полі **Назва:** і натиснути кнопку **Далі**.

Нотатки

- Типово, у спадному списку **Фільтр:** буде позначено пункт **Всі файли, що підтримуються (*.kexi, *.kexic, *.kexis, *.mdb)**. Якщо назва потрібного вам файла має інший суфікс, ви можете обрати у спадному списку **Фільтр:** пункт **Всі файли**, щоб побачити список всіх файлів у теці (безвідносно до суфікса назви).
- Якщо ви оберете тип файла, створеного сторонньою програмою, наприклад файл **MS Access .mdb**, Kexi запропонує вам імпортувати такий файл.
- Якщо ви оберете файл *даних з'єднання* (з суфіксом назви **.kexic**) або *файл посилання на проект сервера бази даних* (з суфіксом назви **.kexis**), Kexi відкриє відповідні діалогові вікна.

2.4.2 Як відкрити вже створений файл бази даних Kexi натисканням піктограми файла .kexi

- Натисніть піктограму файла у програмі для керування файлами або на стільниці. Kexi відкриє вказаний проект бази даних у автоматичному режимі.


Нотатки

Зауваження щодо доступу до віддалених файлів баз даних. Можливо, вам потрібно буде відкрити файл бази даних, який зберігається на віддаленому джерелі даних (наприклад, у інтернеті, на сервері FTP або на спільному ресурсі мережі MS Windows). У стільничному середовищі KDE ви можете відкривати файли з віддалених джерел безпосередньо у програмах і зберігати внесені вами зміни до початкових файлів, але виконувати подібні операції з файлами баз даних не можна. Якщо ви натиснете позначку файла бази даних, що зберігається на віддаленому джерелі даних, програма отримає копію цього файла до тимчасового каталогу на вашому комп'ютері. Всі ваші зміни буде внесено до цього локального файла. Віддалений початковий файл залишатиметься незмінним, отже, рекомендуємо вам спочатку зробити копію (отримати) файла бази даних на вашому комп'ютері, потім відкрити цей файл, внести зміни, зберегти і скопіювати назад до віддаленого джерела даних.


2.5 Користування вбудованою довідкою

Існують такі способи для отримання вбудованої довідки Кехі:

Довідник у формі електронного документа.

Доступ до підручника можна отримати натисканням кнопки **F1** або натисканням кнопки  на панелі меню з наступним вибором пункту **Довідка**.

Підказки *Що це?*.

Виберіть пункт **Що це?** з меню, яке буде показано у відповідь на натискання кнопки , потім натисніть на області інтерфейсу програми, щоб отримати довідкову інформацію щодо нього.

Розділ 3

Побудова простої бази даних

- Вступ
- Розробка таблиць бази даних
 - Вікно створення таблиць
- Введення даних до таблиць
- Розробка запитів до баз даних
- Розробка форм
 - Найважливіші терміни
 - Порівняння форм з таблицями
 - Робота з компоуванням форм
 - Користування вкладкою «Віджети»
 - Додавання віджетів — текстові поля
 - Визначення джерел даних
 - Додавання текстових міток
 - Дії
 - Компоування віджетів
- Введення даних за допомогою форм

3.1 Вступ

З метою вивчення основних прийомів роботи у Кехі, ви можете створити просту базу даних, у якій використовуватимуться найпростіші можливості Кехі. Щоб не ускладнювати вивчення програми, ми не торкатимемося у цій главі складних питань побудови баз даних.

Почнімо зі створення нової порожньої *Телефонної книги*.

Після створення нового порожнього проєкту бази даних, виконайте такі кроки:

1. Розробіть таблиці бази даних. Для цього ознайомтесь з розділом [Розробка таблиць бази даних](#).
2. Введіть дані до таблиць. Для цього ознайомтесь з розділом [Введення даних до таблиць](#).
3. Розробіть форми запитів до бази даних. Для цього ознайомтесь з розділом [Розробка запитів до бази даних](#).
4. Розробіть форми. Для цього ознайомтесь з розділом [Розробка форм](#).
5. Скористайтесь формами для введення даних. Для цього ознайомтесь з розділом [Введення даних за допомогою форм](#).


3.2 Розробка таблиць бази даних

По-перше, до вашої бази даних буде додано дві таблиці: *persons* і *Phones*. Ці таблиці ми вже описували у главі [База даних і електронна таблиця](#). З компонуванням таблиці *Persons* можна ознайомитися у розділі [Цілісність і коректність даних](#) тієї ж глави.

1. Натисніть кнопку **Таблиця** на панелі інструментів. Крім того, ви можете просто натиснути кнопку **Створити об'єкт: таблиця** на панелі інструментів «Навігація проектом».
2. Програма відкриє вікно інструменту компонування таблиці. За верхньою частиною вікна компонувальника ви можете помітити, що Кехі запропонувала для вашої таблиці типову назву, подібну до *Таблиця1*. Компонування таблиці ще не було збережено, отже пізніше ви можете призначити таблиці кращу назву. Крім того, з тих самих причин, пункту таблиці ще немає у списку панелі **Навігації проектом**.

3.2.1 Вікно створення таблиць

3.2.1.1 Вікно створення таблиць складається з таких стовпчиків:

-  Піктограма інформації — додаткові дані щодо поля.
- **Підпис поля** — підпис поля, який буде показано під час введення даних.
- **Тип даних** — спадний список, у якому містяться пункти типів даних. За допомогою цього списку ви зможете встановити основне правило введення даних для відповідного поля. Наприклад, якщо для поля встановлено тип даних «цілі числа», користувач бази даних просто не зможе ввести у це поле літери.
- **Коментарі** — тут ви можете навести будь-які відомості, корисні для розуміння призначення відповідного поля. Цей додатковий текст буде збережено у компонуванні таблиці, його можна буде переглянути лише у режимі компонування.

У вікні **Компонувальника таблиці** кожен рядок відповідає одному полю таблиці. Визначити, що програма працює у *режимі компонування* можна за натиснутим станом кнопки **Компонування** панелі вікна компонування таблиці.

3.2.1.2 Розробка таблиці «Особа»

У першому рядку натисніть комірку у стовпчику **Підпис поля** і введіть підпис поля *Name* (Ім'я). Заповнення поля **Підпис поля** призведе до автоматичного заповнення поля **Назва**, як можна бачити на панелі **Редактор властивостей**.

Зауваження щодо назв полів і підписів

- Кожне з полів таблиці повинне мати назву і підпис, ці поля не можна залишати порожніми.
- Назву поля буде використано самою базою даних, звичайні користувачі бази даних її не бачитимуть >. У назві поля не повинно міститися спеціальних символів (символів кирилиці, ±, ¶, Ü) та пробілів. Назва має складатися лише з латинських літер, цифр та знаків підкреслювання «_». Останніми можна скористатися для заміни пробілів та дефісів.
- Назви полів мають починатися з літери або символу підкреслювання «_», але не з цифри.
- Можна використовувати літери у будь-якому регістрі (малі або великі). Для Кехі назва поля *Persons* є тотожною до назви *persons*.
- З іншого боку, у поле підпису можна вводити будь-які літери та спеціальні символи. Вміст цього поля буде показано користувачам бази даних.

У подібний же спосіб, введіть такі поля до компонування таблиці:

- *surname*
- *street*
- *house_number*
- *city*

Для всіх згаданих вище полів, окрім *house_number* (номер будинку), типом даних буде *text*. Змініть тип даних для поля *house_number* на *Ціле число*. Щоб зробити це, натисніть комірку у стовпчику **Тип даних** рядка *house_number*, а потім натисніть кнопку спадного списку або натиснути клавішу **F4**. Ви також можете скористатися клавішами зі стрілочками ↑ та ↓, щоб вибрати інший тип. У відповідь програма відкриє список типів даних. Виберіть у цьому списку пункт *Ціле число*.

Після цього у поле *house_number* можна буде вводити лише числа.

Компонування таблиці *Persons* завершено. Натисніть кнопку **Дані** на панелі інструментів, щоб перемкнути програму у режим перегляду даних таблиці. Тепер до таблиці можна вводити дані.

Оскільки компонування ще не збережено до бази даних, програма відкриє діалогове вікно **Зберегти об'єкт як**. За допомогою цього вікна вам слід вказати назву для нової таблиці.

Кехі запропонує вам загальну назву, наприклад, *Table1*. Щоб змінити цю назву, введіть *Persons* у поле *Підпис* і натисніть клавішу **Enter** або кнопку **Гаразд**. Поле **Підпис** буде використано для показу таблиці кінцевим користувачам, наприклад, як назву форми. На відміну від назви, у підписі можна використовувати будь-які символи, зокрема пробіли і спеціальні символи.

Зауважте, що заповнення поля **Підпис** призведе до автоматичного заповнення поля **Назва**. Для вашої зручності вміст підпису буде відфільтровано так, що у назві міститимуться лише латинські літери, цифри і символи «_». Ви можете змінити вміст поля **Назва**, якщо це потрібно.

1. Програма запитає вас щодо автоматичного додавання основного ключа до таблиці. Для продовження роботи натисніть кнопку **Додати основний ключ**.
2. Програма створить таблицю *Persons* і відкриє її у вікні перегляду даних. Відповідна назва з'явиться у списку панелі **Навігатора проектом**.
3. Створіть таблицю *Phones* у спосіб, подібний до способу створення таблиці *persons*.
4. Створіть поле *person* (особа) типу *Ціле число* і поле *phone* (телефон) типу *Текст*. Використовувати числовий тип даних для цього поля не можна, оскільки телефонні номери мають різну форму запису та містять префікси.
5. Натисніть на панелі інструментів кнопку **Дані** і вкажіть для таблиці підпис *Phones*. Як і для попередньої таблиці, дозвольте Кехі автоматично створити основний ключ.

3.3 Введення даних до таблиць

Ви створили дві таблиці: *Persons* і *Phone*. Жодна з них ще не містить жодних даних. Ви можете ввести ці дані. У цій главі ми навчимося робити це швидко і ефективно.

Почнімо з таблиці *Persons*. Відкрийте її у Перегляді даних за допомогою пункту **Відкрити** контекстного меню панелі Навігація проектом або кнопки панелі інструментів. Поточну комірку буде позначено товстою (зазвичай, чорною) рамкою, курсором комірок. Колір тексту вмісту комірки, якщо у комірці щось міститься, буде позначено змінено на колір підсвічування. Поточний рядок, тобто рядок, у якому розташовано ваш прямокутний курсор, буде позначено ліворуч символом стрілки.

Здійснювати навігацію комірками таблиці можна за допомогою клавіш зі стрілочками, клавіш **Page Down**, **Page Down**, **Home**, **End**; крім того, ви можете навести вказівник на комірку таблиці і клацнути лівою кнопкою, щоб позначити цю комірку.

Одразу після відкриття таблиці *Persons*, курсор буде розташовано у стовпчику *id*. Для стовпчика буде визначено властивість автоматичного нумерування, що буде позначено синім

написом (*autonumber*) у останньому рядку. Це означає, що вам не потрібно вводити дані вручну у новий рядок, оскільки комірки буде автоматично заповнено послідовними числами.

Спосіб додавання нових рядків і введення даних у ці рядки у Кехі відрізняється від способу, до якого ви звикли під час роботи з електронними таблицями. Щоб ввести дані у новий рядок, вам слід скористатися клавішами зі стрілочками або миші для пересування курсора у особливий порожній останній рядок, позначений символом «плюс». Розташуйте курсор у стовпчику *name* (другому стовпчику) і введіть ім'я особи. Крім того, введіть прізвище, назву вулиці, номер будинку та назву міста. Після введення даних пересуньте курсор комірок до останнього порожнього рядка за допомогою або натискання клавіші ↓, або за допомогою натискання останнього рядка вказівником миші, — програма додасть новий рядок.

3.3.1 Докладно про доступні дії під час введення даних до таблиць

- Після введення першого символу розпочнеться редагування поточного рядка. Ліворуч від таблиці даних буде показано символ олівця.
- Наведення вказівника миші на комірку з наступним подвійним клацанням лівою кнопкою миші, натискання клавіші **Enter** або клавіші **F2** також накажуть програмі розпочати редагування поточного рядка.
- Натискання клавіші **Esc** під час редагування вмісту комірки *скасовує зміни, внесені до вмісту цієї комірки*. Але значок олівця не зникне, оскільки ви все ще можете перейти до іншої комірки рядка для зміни її вмісту. Щоб *скасувати зміни, внесені до всього рядка*, натисніть клавішу **Esc** ще раз.
- Замість натискання клавіші **Esc** ви можете натиснути кнопку **Скасувати запис змін** панелі інструментів.
- Натисніть комбінацію клавіш **Shift+Enter**, щоб підтвердити всі зміни, внесені до всіх комірок у рядку, який ви редагуєте.

Вкажіть дані у таблиці *Phones*. У стовпчику *person* вам слід вказати номер (*id*) особи з таблиці *Persons*.

3.4 Розробка запитів до баз даних

Основним призначенням бази даних є зберігання даних та полегшення отримання потрібної вам інформації. На відміну від баз даних, записаних на листках паперу, база даних Кехі надає вам змогу вказувати більше критеріїв пошуку. Результати буде повернуто швидше, навіть для баз даних великого об'єму. Це переваги використання баз даних, але, щоб мати змогу створювати ефективні *запити* у вашій базі даних, вам слід навчитися повідомляти базі даних про те, що ви шукаєте.

За допомогою запитів до бази даних ви можете обмежити дані, який буде отримано від таблиці з даними, попередньо визначеним набором рядків і стовпчиків, а також у динамічному режимі об'єднувати дані, що знаходяться від декількох таблиць.

Щоб ознайомитися з тим, як працюють запити, розглянемо приклад. Створимо запит *contacts* (контакти) у якому поєднуватимуться дані з двох таблиць: *persons* і *Phones* (створення цієї таблиці обговорено [тут](#), а її заповнення даними [тут](#)).

1. Створіть порожній запит за допомогою пункту меню **Запит** панелі інструментів. У відповідь на вибір цього пункту програма відкриє вікно компонування. Це вікно буде розділено на дві області: взаємозв'язки запитів у верхній частині і стовпчики запитів, розташовані нижче.
2. Оберіть таблицю *persons* у спадному списку **Таблиця:**, розташованому у верхній частині вікна, і натисніть кнопку **Вставити**. На панелі зв'язків з'явиться графічне представлення для таблиці. Зробіть те саме для таблиці *Phones* і вставте її, як це показано на наведеному нижче знімку.

3. Додайте зв'язок між запитом за допомогою перетягування мишею зі скиданням: натисніть поле *id* у таблиці *persons*, перетягніть і скиньте його на поле *person* таблиці *Phones*. Таким чином, ви *з'єднаєте обидва поля створенням взаємозв'язку*.
4. Наведіть вказівник миші на поле *name* у таблиці *persons* і двічі клацніть лівою кнопкою миші, щоб додати це поле як стовпчик запиту. У подібний же спосіб додайте поля *sur name*, *street*, *house_number*, *city* з таблиці *persons* та поле *phone* з таблиці *Phones*.
5. Тепер компонування запити готове до перевірки. Натисніть кнопку **Дані** на панелі інструментів, щоб перемкнути програму з режиму компонування у режим перегляду результатів пошуків за запитом.
6. Збережіть компонування запити для наступного використання натисканням клавіші **Зберегти** на панелі інструментів. Оскільки ви ще не зберігали це компонування запити, програма попросить вас надати йому належну назву. Введіть *Contacts* у поле **Заголовок** і натисніть кнопку **Гаразд**.

3.5 Розробка форм

- [Найважливіші терміни](#)
- [Порівняння форм з таблицями](#)
- [Робота з компонуванням форм](#)
- [Користування вкладкою «Віджети»](#)
- [Додавання віджетів — текстові поля](#)
- [Визначення джерел даних](#)
- [Додавання текстових міток](#)
- [Дії](#)
- [Компонування віджетів](#)

3.5.1 Найважливіші терміни

Форма

Вікно, призначене для простого введення даних і показу цих даних на екрані комп'ютера.

Джерело даних форми

Таблиця бази даних або запит, у яких містяться дані, показані на формі. Джерело даних потрібне, оскільки самі форми є лише *інструментами* для показу і введення даних, джерелом же даних є таблиці і запити. У нових щойно створених форм немає призначених джерел даних, отже у них не буде показано ніяких даних з вашої бази даних аж доти, доки ви не пов'яжете з формою певне джерело даних.

Поле форми

Прямий аналог стовпчика у таблиці або запиті. Найчастіше поля використовують для показу тексту і чисел. Введення нового значення або зміна існуючого значення такого поля призводить до зміни у відповідному стовпчику таблиці або запити (після підтвердження зміни).

Компонування форми

Виконане вами визначення вигляду і функціональних можливостей форми. Для того, щоб виконати таке визначення, вам слід вказати джерело даних, вставити поля форми різноманітних типів і певним чином розташувати ці поля на формі.

Віджет форми

Елемент форми. Основними типами віджетів є:

- Віджети, призначені для показу інформації, наприклад, поле для показу тексту або зображення. Кожен з віджетів такого типу можна *пов'язати* з певним полем джерела даних (стовпчиком таблиці або запиту). Тому такі віджети скорочено називаються полями форми.
- Віджети, які можуть виконувати вказану для них дію, наприклад, кнопка, за допомогою якої можна закрити поточну форму. У інших програмах подібні віджети іноді називають керуванням формою, оскільки такі віджети можуть виконувати заздалегідь визначену дію з *керування* поведінкою вашої програм для роботи з базами даних.
- За допомогою інших віджетів можна збагатити вигляд форми, наприклад, «віджет лінії» може візуально відокремити дві області форми.

Віджет контейнера

Віджет, який може *містити* інші віджети у відведеній для цього віджета ділянці. Наприклад, віджет блоку або віджет вкладки є контейнерами. Сама площа форми є також контейнером. Кнопку команди не можна вважати контейнером, оскільки на неї не можна додати віджета. У складніших випадках віджети контейнерів може бути додано у інші контейнери, тобто можливе вкладення контейнерів.

3.5.2 Порівняння форм з таблицями

У розділі [Введення даних до таблиць](#) ми навчилися вводити дані безпосередньо до таблиць за допомогою панелі перегляду аркуша даних. Але здебільшого форми є придатнішими для введення даних:

- У таблиці може бути багато стовпчиків: показ всіх цих стовпчиків на екрані буде неможливим. У формі такі дані може бути показано за допомогою декількох рядків.
- За допомогою форми можна візуально поділити поля даних на логічні групи з метою полегшення прийняття. На форму можна додати мітки з додатковими відомостями, які нададуть користувачам змогу легше зрозуміти способи користування формою та призначення полів даних.
- Кнопками команд форм можна скористатися для виконання найпоширеніших команд. Користувачі таких форм зможуть працювати з ними як зі знайомими окремими програмами.

На панелі перегляду аркуша даних показати рядки даних з текстом полів або зображення так само просто, як і на формах.

3.5.3 Робота з компоуванням форм

Під час розробки таблиці або запиту ви зможете користуватися двома режимами перегляду: Перегляд даних і Перегляд дизайну. Компоування форми виконується у режимі Перегляду дизайну. Ми будемо називати вікно дизайну форми Дизайнером форми.

1. Щоб створити порожню форму, натисніть пункт **Форма** на панелі інструментів. Якщо ви хочете, ви можете скористатися пунктом **Створити об'єкт: Форма** контекстного меню панелі **Навігація проектом**.
2. Програма відкриє нову панель. Змінити розміри форми можна за допомогою пересування меж панелі. Форму буде вкрито сіткою для полегшення точного розташування віджетів.

Так само, як і у компоуванні таблиць, у Дизайнері форм передбачено Редактор властивостей. Для того, щоб зекономити місце на екрані, на панелі буде показано три вкладки, пов'язані з поточною позначеною формою:

Вкладка «Властивості»

Містить список властивостей для поточного вибраного віджета.

Вкладка «Джерело даних»

Містить властивості, пов'язані безпосередньо з джерелом даних поточного позначеного віджета або самої форми.

Вкладка «Віджети»


Містить ієрархію всіх віджетів форми. За допомогою списку спрощується пошук віджетів за назвою та навігація між ними.

Буде показано відомості щодо назви і типу поточного позначеного віджета на першій і другій вкладці.

Доступні також додаткові панелі інструментів:

- Панель інструментів **Віджети** призначено для додавання нових віджетів на форму. Натисніть кнопку **Компонування форм**, щоб побачити її.

3.5.4 Користування вкладкою «Віджети»

На вкладці  (**Віджети**) панелі **Властивості** буде показано список віджетів форми та їх ієрархічну підпорядкованість. Всі віджети у ієрархії буде показано за рівнями (віджети одного рівня буде показано у їх спільному батьківському контейнері). Пункти дочірніх віджетів (віджетів, що містяться у контейнерах) буде показано з відступом.

Для кожного з віджетів буде показано назву і тип. Для позначення типу буде також використано піктограм (ту саму, яку було показано на кнопці панелі інструментів, яку ви натискали, щоб додати відповідний віджет під час компонування).

ПРИМІТКА

- Зміна поточного позначеного пункту у списку призводить до позначення відповідного елемента на формі. Така взаємодія полегшує пошук віджетів за назвою і спрощує навігацію формою. Наприклад, можна обрати віджет за назвою, а потім перейти на вкладку **Властивості**, щоб змінити властивості віджета.
- Утримування натиснутою клавіші **Ctrl** під час позначення пунктів у списку віджетів надасть вам змогу позначити декілька віджетів одночасно. Якщо утримувати натиснутою клавішу **Shift**, можна позначити послідовності віджетів у списку.

Надання віджетам змістовних назв є корисним, але не обов'язковим кроком. Зауважте, що назва віджета є властивістю, яку не буде показано користувачеві вашої форми. Користувачі бачитимуть лише текст віджета, який визначається властивістю **Text** або іншою подібною властивістю.

3.5.5 Додавання віджетів — текстові поля

Давайте створимо форму з відомостями щодо осіб, тобто форму, пов'язану з таблицею *Persons*

Якщо на формі, яку ви розробляєте, має бути показано дані, отримані з бази даних, вам слід розташувати на цій формі відповідні поля. Для подібного розташування слід скористатися кнопками панелі інструментів **Віджети**. Кожна з кнопок відповідає певному типу віджета.


1. Натисніть кнопку **Текстове поле** на панелі інструментів **Віджети**.

2. Наведіть вказівник миші на форму і клацніть лівою кнопкою миші. У точці, де знаходився вказівник миші, буде розташовано новий віджет поля для введення тексту. Перш ніж відпустити кнопку миші ви можете спробувати перетягнути вказівник миші, щоб змінити розміри віджета на бажані.
3. Якщо потрібно, пересуньте доданий віджет за допомогою перетягування зі скиданням вказівником миші у потрібну позицію. Пізніше ви зможете змінити розміри віджета перетягуванням одного з маленьких квадратиків, розташованих у кутах віджета. Зауважте, що ці квадратики можна буде побачити, лише якщо віджет буде позначено. Якщо ви позначите інший віджет або всю поверхню форми, квадратики буде приховано.
4. Натисніть кнопку **Текстове поле** ще раз, а потім наведіть вказівник миші на поверхню форми і клацніть лівою кнопкою миші, щоб вставити ще один віджет. Повторіть цю процедуру ще раз так, щоб на вашій формі опинилося три додаткових поля для введення тексту. Для простоти ми обмежимося роботою з п'ятьма полями даних.


ПРИМІТКА

- У режимі компонування форм передбачено також контекстне меню, доступ до якого можна отримати наведенням вказівника на бажаний віджет або поверхню форми з наступним клацанням правою кнопкою миші. У цьому меню ви знайдете команди **Вирізати**, **Копіювати**, **Вставити**, **Вилучити** та інші, складніші команди. Крім того, для виконання цих команд передбачено клавіатурні скорочення. Деякими з цих команд можна скористатися лише для віджетів певних типів.
- За допомогою пунктів **Вирізати**, **Копіювати** і **Вставити** можна пересувати або копіювати віджети між формами, навіть між окремими проектами баз даних.
- Утримування натиснутою клавіші **Ctrl** під час клацання кнопкою миші надає змогу позначити декілька віджетів одразу.
- Окрім використання пунктів меню **Копіювати** і **Вставити** для дублювання віджета на одній формі, ви можете натиснути і утримувати натиснутою клавішу **Ctrl**, а потім пересунути віджет. Після відпускання кнопки миші перетягнутий віджет не буде пересунуто, а скопійовано у нову позицію.

3.5.6 Визначення джерел даних


Для вставлених вами полів ще не призначено *джерела даних*, отже у цих полях не може бути показано даних з бази даних. Щоб призначити джерело даних, скористайтесь вкладкою  (**Джерело даних**) редактора властивостей.

На першому кроці слід вказати джерело даних форми, тобто вказати сховище даних звідки слід брати показані дані Як ми вже раніше згадували, нам слід скористатися таблицею *persons* як джерелом даних для нашої нової форми

1. Наведіть вказівник миші на вікно форми і клацніть лівою кнопкою миші, так само, як під час зміни його властивостей.
2. Перемкніться на вкладку  (**Джерело даних**) і введіть назву таблиці *persons* (особи) у спадний список **Джерело даних форми**. Крім того, ви можете просто обрати цю назву зі спадного списку.

Отже ми призначили джерело даних форми. Тепер нам треба визначити джерело даних віджетів.



1. Наведіть вказівник миші на перший з віджетів текстових полів у верхній частині форми і клацніть лівою кнопкою миші.

2. На вкладці  (**Джерело даних**) панелі властивостей введіть назву поля `name` (ім'я) у спадному списку *Джерело даних віджета*. Крім того, ви можете обрати цю назву зі спадного списку.
3. Натисніть наступне поле тексту і введіть у поле джерела даних `surname` (прізвище).
4. Введіть джерела даних для текстових полів `street` (вулиця), `house_number` (номер будинку) і `city` (місто) у подібний же спосіб.

Тепер можна зберегти компонування форми (випробувати форму у дії не обов'язково). Щоб зберегти компонування, натисніть кнопку панелі інструментів **Зберегти**. У процесі зберігання програма попросить вас ввести назву форми. Введіть *Persons* у поле підпису і натисніть кнопку **Гаразд**. Поле назви форми буде заповнено автоматично.

Настав час перевірити нашу форму у дії. Натисніть кнопку панелі інструментів **Дані**. Якщо під час введення даних ви не помилилися, ви побачите поля форми, заповнені даними з таблиці *persons*.

ПРИМІТКА

- Якщо ви бажаєте вилучити прив'язку джерела даних до віджета форми, натисніть кнопку  у правій частині рядка **Джерело даних віджета**. У такий самий спосіб ви можете вилучити прив'язку до джерела даних.
- Скористайтеся натисканням кнопки  (**Перейти до вибраного джерела даних форми**), щоб вибрати відповідну таблицю або запит на панелі Навігації проєктом. Таким чином, ви зможете швидко відкрити таблицю або запит, який буде джерелом даних форми.

3.5.7 Додавання текстових міток

Для того, щоб полегшити користувачеві форми визначення призначення полів віджета, до цих полів слід долучати текстові мітки з відповідними написами. Для створення текстових міток використовують віджет **Надпис**.

Додайте на форму три віджети текстових міток, розташувавши їх ліворуч від полів для введення тексту (або праворуч, якщо у вашій операційній системі використовується лівостороннє компонування) Після додавання нової мітки на формі з'являтиметься курсор у місці, куди слід вводити текст. Послідовно введіть мітки: *Name* (ім'я), *Surname* (прізвище) та *Street* (вулиця). Крім того, у верхній частині форми додайте ще одну мітку, яка визначатиме назву форми, тобто *Persons*. Збільшіть розміри цієї мітки та розмір шрифту її тексту за допомогою пункту меню **Шрифт** вкладки **Властивості**.

3.5.8 Дії

Action (дія) — це якась виокремлена функціональна можливість програми, дію якої може бути активовано користувачем. Дію може бути виконано і у автоматичному режимі у відповідь на вказану подію (наприклад, відкриття форми).

3.5.8.1 Призначення дій до кнопок форми

З кнопками форми можна пов'язати багато дій. Призначену дію буде виконано після натискання кнопки.

Для призначення дії:

1. Перемкніть програму у режим перегляду Компонування, якщо ви ще цього не зробили.

2. Позначте вже створений віджет кнопки наведенням на нього вказівника миші з наступним клацанням лівою кнопкою миші або додайте на форму новий віджет кнопки. Якщо ви додали нову кнопку, вкажіть її назву і натисніть клавішу **Enter**.
3. Наведіть вказівник миші на віджет кнопки і клацніть правою кнопкою миші для відкриття контекстного меню.
4. З контекстного меню оберіть пункт **Призначити дію...**
5. У відповідь програма відкриє діалогове вікно **Призначення дії для кнопки**, де буде показано список можливих дій. Якщо для віджета вже було призначено дію, один з пунктів дій у цьому списку буде позначено. Якщо ж дії ще не було призначено, у спадному списку **Категорія дії** буде позначено пункт **Без типу**.
6. У спадному списку **Категорія дії** виберіть пункт **Дії програми**. У відповідь програма покаже список можливих дій програми.
7. Виберіть одну з дій зі списку (наприклад, **Вилучити позначений об'єкт**).

Після перемикаання у режимі перегляду даних форми ви можете перевірити, чи працює дія.

ПРИМІТКА

- Щоб вилучити прив'язку дії, оберіть пункт **Без дії** у спадному списку **Категорія дії** діалогового вікна **Призначення дії для кнопки**.
- Дії працюють лише у режимі *перегляду даних* форми. Не кожне з призначення дій має сенс. Наприклад, ви можете отримати доступ до дії **Шрифт...** у режимі перегляду даних, але лише якщо у перегляді Дизайн позначено якийсь віджет. Якщо ви внесете зміни до параметрів шрифту, зміни буде застосовано до тексту цього позначеного віджета.

3.5.9 Компонування віджетів

Віджети форми варто розташувати зручно для користувача та вирівняти. Розташування, вирівнювання і зміна розмірів віджетів вручну не є простою справою. Ці параметри віджета не коригуватимуться, коли користувач змінюватиме розміри форми. Фактично, стан справ ще гірший, оскільки ви не зможете передбачити потрібний для віджетів простір за умови, що на комп'ютері користувача буде встановлено інші розміри шрифтів і іншу роздільну здатність екрана.

За допомогою особливого інструменту, який називається «компонування віджетів» можна автоматично розкласти віджети формою. Компонування віджетів — це дія з групування декількох віджетів так, щоб їх було розташовано у відповідних місцях, і так, щоб вони мали відповідні розміри.

Використання компоновання покращує вирівнювання елементів форми. Крім того, краще використовується площа форми. Текстові поля наближаються одне до одного, інтервали стають однорідними.

3.5.9.1 Правила розмірів для віджетів у компонованні

Замість встановлення для віджетів фіксованого розміру у Кехі ви можете скористатися одним з декількох правил встановлення розмірів віджетів. *Правила розмірів* — це гнучка стратегія керування способом розтягування (або стискання) віджета залежно від розміру і розташування сусідніх віджетів і доступного місця на формі.

Після додавання віджетів розміри рядка буде змінено так, щоб у нього вмістився видимий текст віджета.

Параметри правил розмірів для кожного з віджетів доданих на форму можна встановити за допомогою Редактора властивостей. Параметри буде зібрано у групі властивостей з назвою **Правила розміру**.

Ця група властивостей містить:

Правила горизонтальних розмірів

визначення горизонтального розміру віджета,

Правила вертикальних розмірів

визначення вертикального розміру віджета,

Горизонтальне розтягнення

визначення рівня застосування **Правил горизонтальних розмірів**,

Вертикальне розтягнення

визначення рівня застосування **Правил вертикальних розмірів**,

3.5.9.2 Величини правил розмірів

У спадних списках властивостей панелі Редактор властивостей **Правила горизонтальних розмірів** і **Правила вертикальних розмірів** можна буде обрати такі пункти:

Фіксований

таке значення означає, що розміри віджета не будуть змінюватися автоматично, його розміри завжди залишатимуться такими, якими їх було визначено під час компонування.

Мінімальний

такий варіант означає, що програма встановить мінімальний можливий розмір для віджета, цього буде достатньо для того, щоб не збільшувати розміри віджета, але початкові розміри буде збільшено, якщо це буде потрібно. Цим типом правил можна скористатися для примусового збільшення віджета до повної ширини або висоти, особливо, якщо ви встановите значення параметра розтягування, більше за 0.

Максимальний

такий варіант означає, що початково програма встановить максимальний можливий розмір віджета, цей розмір буде зменшено без шкоди для зручності користування віджетом та перегляду його тексту, якщо для показу інших віджетів потрібен додатковий простір,

Бажаний

цей варіант означає, що початковий розмір віджета є найкращим і бажаним; але розміри віджета буде зменшено або збільшено так, щоб розташований на ньому текст можна було читати,

Розширюється

такий варіант означає, що хоча програма намагатиметься зберегти початкові розміри віджета, але їх може бути зменшено та збільшено з метою раціонального використання площі,

Мінімально розширюється

такий варіант означає, що хоча програма намагатиметься зберегти початкові розміри віджета, їх може бути збільшено так, щоб віджет займав якомога більшу площу,

Ігнорується

такий варіант означає, що початкові розміри віджета буде проігноровано; розміри віджета може бути збільшено так, щоб вони стали найбільшими можливими, але, зазвичай, ці розміри буде обмежено іншими віджетами форми.

Для віджетів різних типів типово застосовуються різні правила встановлення розмірів; наприклад, для віджетів кнопок типово встановлюються правила розмірів **Мінімальний** (у обох напрямках), а для віджетів текстових полів встановлюються правила визначення вертикального розміру **Фіксований**.

Найпоширенішими правилами зміни розмірів віджетів є **Бажаний**, **Мінімальний** і **Максимальний**.

3.5.9.3 Вертикальне та горизонтальне розтягнення

Для властивостей **Вертикальне розтягування** і **Горизонтальне розтягування** використовуються цілі числа, більші або рівні 0. За допомогою цих властивостей ви можете скоригувати застосування правил встановлення розмірів. Типовим значенням для цих властивостей є 0. Більше значення розтягування означає, що віджет буде розтягнуто більше, ніж віджет з меншим значенням розтягування.

3.6 Введення даних за допомогою форм

Введення даних та їх редагування є, зазвичай, завданням користувача програми з керування базами даних. Розробник бази даних має потурбуватися про перевірку коректності введених до форми даних та про належну їх обробку формою.

Щоб перевірити функціонування вашої форми, перемкніть форму у режим перегляду даних. Буде показано окремий рядок бази даних (запис). Ви можете пересувати курсор між полями за допомогою клацання лівою кнопкою миші або натискання клавіш **Tab** та **Shift+Tab**. Під час редагування поряд з інструментом навігації записами буде показано піктограму з зображенням олівця. Після введення даних рядка (запису) ви можете натиснути комбінацію клавіш **Shift+Enter** або кнопку **Зберегти запис** на панелі інструментів, щоб підтвердити внесені у поточний рядок зміни. Якщо ви натиснете на панелі інструментів кнопку **Скасувати зміни у записі**, програма відкине зміни, внесені до поточного рядка і відновить значення полів даних. Для пересування до нового рядка ви можете скористатися натисканням кнопки записів інструменту навігації. Крім того, доступ до всіх функціональних можливостей можна отримати у спосіб, подібний до способу, що використовується у перегляді даних таблиці.

Розділ 4

Як налаштувати Кехі

- Швартування і від'єднання бічних панелей

4.1 Швартування і від'єднання бічних панелей

Бічні панелі **Навігатора проєктом** та **Редактора властивостей** можна розблокувати у два способи:

- двічі клацнути на смужці «захоплення» у верхній частині панелі або
- навести вказівник миші на кнопку з ромбом у верхній частині панелі, розташовану поряд з хрестиком, і один раз клацнути лівою кнопкою миші.

Після розблокування вікна панелей можна знову пришвартувати до головного вікна подібно до розблокування:

- двічі клацнути на смужці «захоплення» у верхній частині вікна або
- навести вказівник миші на кнопку з ромбом у верхній частині панелі, розташовану поряд з хрестиком, і один раз клацнути лівою кнопкою миші.

ПІДКАЗКА

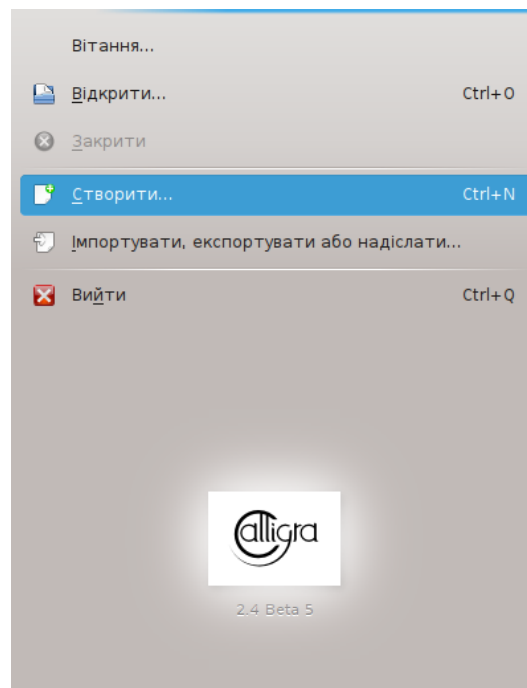
Після від'єднання панелі Кехі запам'яте останню позицію швартування, отже під час повторного швартування панелі буде відновлено її останнє розташування.

Розділ 5

Довідка щодо команд

- Вкладка «Кехі»
- Вкладка «Створення»
- Вкладка «Дані»
- Вкладка «Зовнішні дані»
- Вкладка «Інструменти»
- Вкладка «Компонування форм»
- Вкладка «Компонування звіту»

5.1 Вкладка «Кехі»



За допомогою вкладки «Кехі» ви можете керувати проектами Кехі.

Вибір пункту дії призведе до відкриття відповідного вікна праворуч. У цьому вікні ви зможете керувати послідовністю потрібних вам дій.

Пункти меню вкладки:

Вітання

За допомогою цього пункту ви можете відкрити проєкт, над яким ви працювали раніше.

Відкрити

За допомогою цього пункту ви можете відкрити проєкт Кехі, що зберігається у **файлі** або на **сервері бази даних**.

Закрити

За допомогою цього пункту можна закрити проєкт Кехі, який зараз відкрито.

Створити

За допомогою цього пункту можна відкрити вікно майстра створення проєкту Кехі.

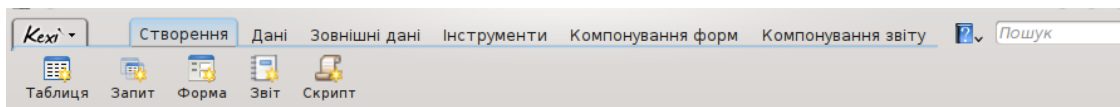
Імпортувати, експортувати або надіслати

За допомогою цього пункту можна відкрити вікно майстра імпортування даних до поточного відкритого проєкту Кехі.

Вийти

Завершує роботу Кехі.

5.2 Вкладка «Створення»



За допомогою вкладки **Створення** ви можете створити об'єкти, які буде додано до вашого проєкту.

Таблиця

Вибір пункту **Таблиця** відкриє вікно **Дизайнера таблиць** у режимі **Перегляд компонування** і надасть вам змогу додати визначення полів до вашої таблиці.

Запит

Вибір пункту **Запит** відкриє вікно **Дизайнера запитів** у режимі **Перегляд компонування** і надасть вам змогу створити запит щодо отримання певних даних з ваших таблиць у проєкті.

Форма

Вибір пункту **Форма** відкриє вікно **Дизайнера форм** у режимі **Перегляд компонування** і надасть вам змогу додати віджети на вашу форму.

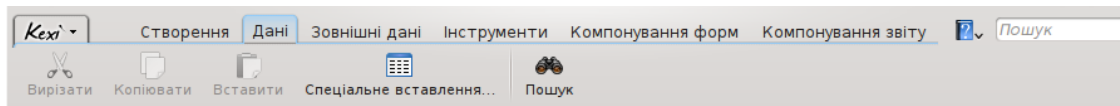
Звіт

Вибір пункту **Звіт** відкриє вікно **Дизайнера звітів** у режимі **Перегляд компонування** і надасть вам змогу додати віджети до вашого звіту.

Скрипт

Вибір пункту **Скрипт** відкриє вікно **Редактор скриптів**, за допомогою якого ви зможете додати до вашого проєкту та/або об'єктів власний код обробки.

5.3 Вкладка «Дані»



За допомогою вкладки **Дані** ви можете керувати даними у ваших таблицях або віджетах об'єктів (форм, звітів тощо).

Вирізати

За допомогою пункту **Вирізати** можна пересунути дані або віджет до буфера обміну даними і вилучити їх з поточної позиції.

Копіювати

За допомогою пункту **Копіювати** можна пересунути дані або віджет до буфера обміну даними без вилучення їх з поточної позиції.

Вставити

За допомогою пункту **Вставити** можна вставити дані або віджет з буфера обміну даними до таблиці або об'єкта.

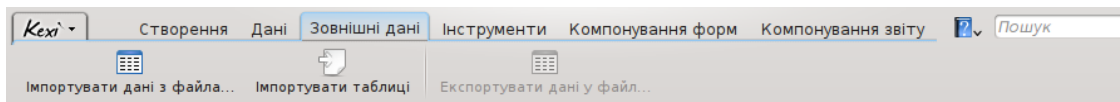
Спеціальне вставлення

За допомогою пункту **Спеціальне вставлення** можна розташувати у таблиці довільні дані з буфера обміну даними у такий спосіб, щоб дані було розміщено у таблиці відповідно з її визначенням.

Знайти

За допомогою пункту **Знайти** можна відкрити вікно пошуку з метою пошуку певного фрагмента тексту у даних бази даних.

5.4 Вкладка «Зовнішні дані»



За допомогою вкладки **Зовнішні дані** ви можете імпортувати до проекту Kexi дані з інших джерел.

Імпортувати дані з файла

За допомогою пункту **Імпортувати дані з файла** можна відкрити діалогове вікно, призначене для імпортування даних у форматі CSV або у форматі звичайного тексту до нової таблиці у вашому проекті.

Імпортувати таблиці

За допомогою пункту **Імпортувати таблиці** можна відкрити вікно **Майстер імпортування таблиць**, призначене для імпортування даних з файлів `fods`, `mdb`, `ods`, `tsv` або даних з іншого сервера бази даних до нової таблиці у проекті.

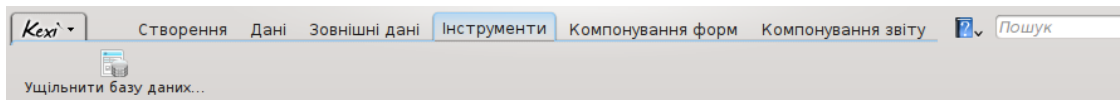
Експортувати дані у файл

За допомогою пункту **Експортувати дані до файла** можна відкрити діалогове вікно експортування даних поточної таблиці у текстовому форматі або у форматі CSV.

ПРИМІТКА

Щоб мати змогу експортувати дані з таблиці до файла, слід спочатку відкрити відповідну таблицю.

5.5 Вкладка «Інструменти»



За допомогою вкладки **Інструменти** ви можете керувати вашими проектами.

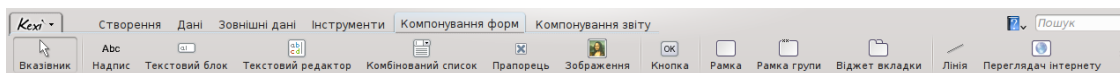
Ущільнити базу даних

За допомогою пункту **Ущільнити базу даних** можна запустити інструмент пошуку незначних помилок у базі даних та зменшення її розмірів.

ПРИМІТКА

Вам слід регулярно ущільнювати вашу базу даних з метою підтримання її у належній формі, особливо після внесення до бази даних значних змін (масованого додавання або вилучення даних).

5.6 Вкладка «Компонування форм»



За допомогою вкладки **Компонування форм** ви можете додавати віджети на вашу форму.

Вказівник

Вказівник перемикає форму у режим вибору віджета.

ПРИМІТКА

Після вибору віджета форму буде переведено у режим додавання віджета. У цьому режимі вам достатньо клацнути у потрібному місці форми лівою кнопкою миші, щоб додати на форму віджет.

Мітка

За допомогою віджета **Надпис** можна показати на формі попередньо визначений напис. Зазвичай, цей віджет застосовується для додавання підписів поряд з іншими віджетами показу даних.

Текстовий блок

Віджет **Текстовий блок** є однорядковим контейнером для даних, що містяться у вашій таблиці.

Текстовий редактор

Віджет **Текстовий редактор** є багаторядковим контейнером для даних, що містяться у вашій таблиці.

Комбінований список

Віджет **Комбінований список** показує спадний список, у якому ви можете вибрати певний пункт.

Прапорець

Віджет **Прапорець** може мати два або три стани позначеності (наприклад, «Увімкнено/Вимкнено»)

Зображення

Віджет **Зображення** може містити зображення, пов'язане з полем у таблиці.

Кнопка

За допомогою віджета **Кнопка** ви можете визначати дії, які буде виконано у відповідь на натискання кнопки на формі.

Рамка

Віджет **Рамка** використовується для зберігання інших віджетів.

Рамка групи

Віджет **Рамка групи** використовується для групування інших віджетів з елементами, які керують станом цих віджетів.

Віджет вкладки

Віджет вкладки — контейнер для інших віджетів, який може складатися з декількох сторінок, на кожній з яких міститимуться різні віджети.

Рядок

За допомогою віджета **Лінія** можна відокремити частини форми.

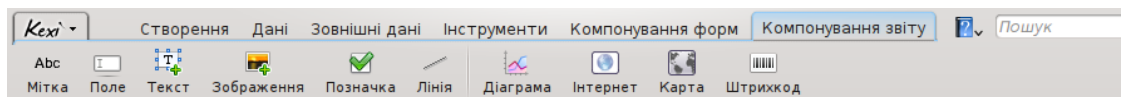
Переглядач інтернету

За допомогою віджета **Переглядач інтернету** можна наказати програмі показувати сторінку інтернету на формі.

Призначити дію

Віджет **Призначити дію** використовується для призначення дії, яку буде виконано у відповідь на певну подію (наприклад натискання кнопки).

5.7 Вкладка «Компонування звіту»



За допомогою вкладки **Компонування звіту** ви можете додавати віджети до вашого звіту.

Мітка

За допомогою віджета **Надпис** можна показати у звіті попередньо визначений напис. Зазвичай, цей віджет застосовується для додавання підписів поряд з іншими віджетами показу даних.

Поле

Віджет **Поле** є однорядковим контейнером для даних, що містяться у вашій таблиці.

Текст

Віджет **Текст** є однорядковим контейнером для даних, що містяться у вашій таблиці.

Зображення

Віджет **Зображення** може містити зображення, пов'язане з полем у таблиці.

Прапорець

Віджет **Прапорець** може мати два або три стани позначеності (наприклад, «Увімкнено/Вимкнено»)

Рядок

За допомогою віджета **Лінія** можна відокремити частини форми.

Діаграма

За допомогою віджета **Діаграма** ви можете додати до звіту візуальне представлення його даних у формі графіка.

Інтернет

За допомогою віджета **Інтернет** до звіту можна додати мінімалістичну форму перегляду інтернету, яка показуватиме дані з локального сайта або сайта у інтернеті.

Штрихкод

Віджет **Штрихкод** використовується для створення штрихкодів у друкованій версії звіту на основі даних, що містяться у полі.

Розділ 6

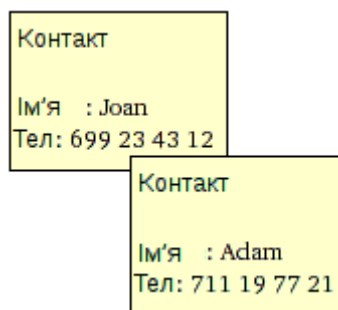
Додаток А. Вступ до баз даних

- Що таке база даних?
- База даних і електронна таблиця
- Компонування бази даних
- Кому потрібні бази даних?
- Програмне забезпечення для створення баз даних

6.1 Що таке база даних?

Базу даних можна визначити як збірку даних з певної теми. Цю збірку певним чином впорядковано, що спрощує перегляд інформації, внесення змін та додавання нових записів.

На діаграмі наведено один з прикладів, про який ми говорили раніше: проста телефонна книга.



На наведеному вище малюнку показано набір з двох записів контактів, кожен з яких зображено як окрему картку. Зрозуміло, що такій картці можна поставити у відповідність окремий рядок таблиці:

Назва	Номер телефону
Джоан	699 23 43 12
Адам	711 19 77 21

Табл. 6.1: Таблиця контактів

Терміни та визначення: окремий елемент даних, що є частиною більшої збірки можна назвати рядком або, якщо сказати професійніше, записом. Сама збірка записів називається таблицею. Крім того, найприроднішою назвою для такої таблиці є та назва, яка описує дані, що зберігаються у таблиці, тобто **Контакти**. Крім того, кожен з рядків таблиці складається зі стовпчиків, які часто називають полями. У таблиці **Contacts** два стовпчики (поля): **Name** і **Tel No.**

У простих випадках база даних складається з єдиної таблиці. Багато хто вважає, що база даних — це і є одна таблиця. Як ми згодом переконаємося, для створення справжніх баз даних, зазвичай, потрібно декілька таблиць.

Ну от, у нас вже є проста база даних з однією таблицею *Contacts*.

ПРИМІТКА

Ознайомтеся з даними на сторінці <https://www.zoho.com/creator/database-software-vs-spreadsheet.html>

6.2 База даних і електронна таблиця

Дуже ймовірно, ви вже колись користувалися програмами для роботи з електронними таблицями, зокрема Calligra Sheets, LibreOffice Calc або Microsoft® Excel. Якщо це так, ви певно могли подумати: гаразд, у електронних таблицях і у базах даних є таблиці, навіщо ж користуватися останніми?

Порівнюючи електронні таблиці і бази даних ви можете зіткнутися з такими питаннями (пізніше ми розглянемо ці питання докладніше):

6.2.1 Чим база даних відрізняється від електронної таблиці?

Поступово розширюючи можливості мобільного телефону, додайте до вашої таблиці **Contacts** стовпчик (поле) **Адреса**. Додайте більше телефонних номерів (робочі, домашні) для кожної з осіб у списку, а також додайте до імен прізвища. Щоб це було простіше зрозуміти, зробимо такі припущення:

- кількість осіб у таблиці обмежено двома (очевидно, у справжній базі даних цих осіб можуть бути сотні або тисячі)
- осіб з однаковими іменами і прізвищами у таблиці немає

Ім'я і прізвище	Телефон	Адреса
Joan Smith	699 23 43 12	Вестерн-гейт 1, Варшава
Adam Willson	711 19 77 21	Лондон, Фрогс-драйв 5
Joan Smith	110 98 98 00	Вестерн-гейт 1
Smith Joan	312 43 42 22	Варшава, Вестерн-гейт 1
ADAM Willson	231 83 02 04	Frogs Drive 5, London

Табл. 6.2: Таблиця контактів

Подібну таблицю можна створити як у формі електронної таблиці, так і у формі бази даних. Користуватися електронною таблицею, звичайно ж, дуже просто. З якими ж проблемами ми можемо зіткнутися на цьому етапі?

6.2.2 Цілісність довідкових даних

Припустімо, що ви користуєтеся електронною таблицею і маєте бажання змінити адресу принаймні однієї особи. З цим пов'язано невеличку проблему: вам доведеться міняти цю адресу у багатьох рядках. Наприклад, Джоан зустрічається у трьох рядках. Справжня проблема виникне, якщо ви забудете змінити один з рядків: адреса, пов'язана з однією з осіб стане *неоднозначною*, оскільки *ваші дані втраять цілісність*.

Крім того, не існує простих способів вилучення даних певної особи з таблиці, оскільки для подібного вилучення слід вилучити всі пов'язані з цією особою рядки таблиці.

6.2.3 Резервування даних

Цю проблему безпосередньо пов'язано з попередньою. У полях **Ім'я та прізвище** та **Адреса** ті самі дані доведеться вводити декілька разів. Це типовий для електронних таблиць неефективний спосіб зберігання даних, оскільки за його використання розміри бази даних не виправдано збільшуються, а отже робота з даними вимагатиме підвищеного використання ресурсів комп'ютера (більший об'єм даних і повільніший доступ).

Як можна розв'язати ці проблеми за допомогою бази даних? Ви можете розділити інформацію на менші фрагменти створенням додаткової таблиці *Особа* з лише двома стовпчиками: **Ім'я та прізвище** і **Адреса**:

Ім'я і прізвище	Адреса
Joan Smith	Вестерн-гейт 1, Варшава
Adam Willson	Frogs Drive 5, London

Табл. 6.3: Таблиця осіб

Кожен рядок у таблиці *Особа* відповідатиме *одній особі*. Віднині таблицю **Контакти** буде пов'язано або, як ще кажуть, вона стане реляційною для таблиці **Особа**.

6.2.4 Цілісність і коректність даних

Зауважте спосіб, у який вводяться дані до полів **Назва і прізвище** та **Адреса**. Люди, які вводять дані можуть помилитися, а іноді є просто неуважними. У даних нашого прикладу у обох полях даних імені та прізвища можна використати різні послідовності імені і прізвища (Джоан Сміт і Сміт Джоан; Адам і АДАМ та багато інших варіантів).

Наведена вище проблема демонструє те, що, наприклад, пошук номеру телефону особи з адресою «Вестерн-гейт 1, Варшава» не дасть вам повних результатів. Ви отримаєте лише один рядок з трьох можливих. Крім того, ви не знайдете всіх телефонних номерів, якщо шукатимете за ключовими словами «Джоан Сміт» у полі **Ім'я та прізвище**, оскільки запис «Сміт Джоан» не відповідає ключу пошуку «Джоан Сміт».

Як же розв'язати ці проблеми за допомогою бази даних? Ви можете виконати це завдання зміною компонування таблиці **Особа** у такий спосіб:

1. *Поділити дані* у полі **Ім'я і прізвище** на два окремих поля: **Ім'я і Прізвище**.
2. *Поділити дані* у полі **Адреса** між трьома окремими полями: **Вулиця, Номер будинку і Місто**.
3. *Забезпечити коректність даних*: програма має переконатися, що жодне з полів не є порожнім, наприклад, ви завжди маєте вказувати номер будинку.

Змінена таблиця виглядає десь так:

Підручник з Кехі

Назва	Прізвище	Вулиця	Номер будинку	Місто
Джоан	Сміт	Вестерн-гейт	1	Варшава
Адам	Вілсон	Фрогс-драйв	5	Лондон
<i>Умови</i>				
обов'язкове поле	обов'язкове поле	обов'язкове поле	обов'язкове поле	обов'язкове поле

Табл. 6.4: Таблиця осіб

Завдяки введенню умови **обов'язкове поле** ми можемо забезпечити повноту введення даних. У інших таблицях ви можете, звичайно ж, пропустити деякі поля під час введення даних.

6.2.5 Обмеження перегляду даних

У електронних таблицях всі рядки і стовпчики таблиці показано у дещо важкій для сприйняття формі, особливо у таблицях, що містять дуже великий обсяг даних. Ви, звичайно ж, можете фільтрувати і впорядковувати рядки у електронній таблиці, але вам слід бути дуже обережним з такими діями. Користувачі електронних таблиць ризикують забути, що дані, які вони переглядають, було фільтровано, що може призвести до помилок. Наприклад, під час обчислення сум ви можете вважати, що у таблиці 100 рядків з даними, а насправді, у ній міститься ще 20 прихованих рядків.

Якщо вам потрібно попрацювати над невеличкою підмножиною даних, наприклад, надіслати її іншим користувачам для редагування, ви можете скопіювати частину електронної таблиці і вставити її до іншої електронної таблиці, а після редагування вставити змінені дані назад до основної електронної таблиці. Подібне «ручне» редагування може спричинити втрату даних або помилку у обчисленнях.

Для обмеження кількості записів у *перегляді даних* у програмах для роботи з базами даних передбачено *запити, форми і звіти*.

Дуже практичний спосіб обмеження показаних даних розширеної версії описаної раніше таблиці *Особи*:

Назва	Прізвище	Вулиця	Номер будинку	Місто	Прибуток
Джоан	Сміт	Вестерн-гейт	1	Варшава	2300
Адам	Вілсон	Фрогс-драйв	5	Лондон	1900

Табл. 6.5: Таблиця осіб

Припустімо, ви додали новий стовпчик **Прибутки**, що містить конфіденційні дані. Як тепер поділитися записом контакту з вашими колегами, не *розкриваючи даних щодо прибутків*? Це можна зробити, якщо *ви надасте їм лише результати запиту, а не цілу таблицю*. У запиті можна обрати всі стовпчики, окрім стовпчика **Прибуток**. У галузі роботи з базами даних такі запити часто називають **переглядами**.

6.2.6 Швидкодія і місткість

Навіть якщо ваш комп'ютер є досить потужним, ви зможете легко помітити, що всієї його потужності не вистачить на роботу з повільними великими електронними таблицями. Основною причиною малоефективної роботи з електронними таблицями є відсутність покажчиків, які б пришвидшували пошук даних (у базах даних такі покажчики є). Крім того, якщо ви користуєтеся інструментами на зразок буфера обміну даними, навіть копіювання даних у електронній таблиці іноді може бути досить довготривалою дією.

Відкриття електронної таблиці, де міститься величезний обсяг даних, може тривати дуже довго. Під час відкриття з електронної таблиці до оперативної пам'яті вашого комп'ютера завантажуються досить багато даних. Більша частина цих даних, ймовірно, не знадобиться або і зовсім не потрібна вам для роботи з таблицею. На відміну від електронних таблиць, у базах даних дані завантажуються зі сховища у оперативну пам'ять лише у разі потреби.

Здебільшого у вас немає причин перейматися тим, як саме дані зберігаються у базі даних. Це означає, що на відміну від електронних таблиць, у базах даних не потрібно знайти про:

- Послідовність рядків, оскільки ви можете впорядкувати рядки відповідно до ваших потреб. Крім того, ви можете переглянути одні і ті самі дані у різний спосіб у різному порядку.
- Те саме стосується стовпчиків (полів) таблиці.

Разом з обмеженням показаних даних, описаним у попередньому розділі, ці якості і визначають переваги баз даних.

6.2.7 Запис даних

У сучасних версіях програм для створення електронних таблиць передбачено можливості розробки форм запиту даних. Такі форми корисні, якщо дані не може бути показано у зручній формі у табличній формі, наприклад, якщо у таблиці є рядки з великими фрагментами тексту або всі стовпчики не вміщуються у вікні.

У такому разі вся робота з електронною таблицею буде проблематичною. Поля для введення даних буде досить широко рознесено електронною таблицею. Дуже часто такі поля не буде забезпечено від (навмисного чи ненавмисного) втручання користувачів.

6.2.8 Звіти

У базах даних можливе групування, обмеження та підбиття підсумків щодо даних у формі звіту. Електронні ж таблиці зазвичай друкуються як невеличкі таблички без автоматичного керування розбиттям на сторінки та компоновання полів.

6.2.9 Програмування

У програмах для створення баз даних часто передбачено можливість використання повноцінних мов програмування. У сучасних електронних таблицях також передбачено таку можливість, але за допомогою методів електронних таблиць можна здебільшого змінювати вміст полів електронної таблиці і виконувати просте копіювання даних без врахування правил відповідності та цілісності, про які ми говорили раніше.

Обробка даних у електронній таблиці, зазвичай, виконується за допомогою графічного інтерфейсу користувача, що може сповільнити обробку даних. Бази даних здатні працювати у фоновому режимі без графічних інтерфейсів.

6.2.10 Доступність для декількох користувачів

Важко собі уявити спільне користування одною електронною таблицею декількома користувачами одночасно. Навіть якщо це технічно можливо у сучасних програмах, подібне користування потребує від користувачів дисциплінованості, уваги і знань, а цього гарантувати аж ніяк не можна.

Класичним способом спільного використання даних електронної таблиці є надсилання цілого файлу з нею (зазвичай, електронною поштою) або відкриття доступу до файлу електронної таблиці у комп'ютерній мережі. Цей спосіб виявляється малоефективним, якщо йдеться про великі групи людей: дані, які потрібні вже зараз, може бути заблоковано внаслідок того, що зараз ними користується хтось інший.

З іншого ж боку, бази даних і було розроблено для одночасного користування багатьма користувачами. Навіть у найпростішій версії бази даних можливе блокування лише окремого рівня рядка таблиці, що спрощує спільне використання даних таблиці.

6.2.11 Безпека

Убезпечення електронної таблиці або її окремих розділів за допомогою пароля є лише символічною дією. Після того, як доступ до вашого файлу буде відкрито у комп'ютерній мережі, будь-хто зможе скопіювати файл і спробувати зламати пароль. Здебільшого подібний злам не є аж надто важким, якщо пароль зберігається у самому файлі електронної таблиці.

Блокування редагування або копіювання у електронній таблиці (або її частині) так само легко усунути.

Для функціонування бази даних (окрім тих, які зберігаються у файлі, а не на сервері) не потрібне зберігання у окремому файлі, доступ до якого слід надавати користувачам. Користувачі отримують доступ до даних за допомогою комп'ютерної мережі, для чого, зазвичай, слід вказати ім'я користувача і пароль. Ви отримаєте доступ лише до тих даних (таблиць, форм або навіть вибраних рядків та стовпчиків), доступ до яких було відкрито за допомогою визначення відповідних прав доступу.

Права доступу можуть стосуватися як редагування даних, так і їх читання. Якщо якісь дані вам недоступні, ці дані не буде надіслано на ваш комп'ютер, отже у вас не буде змоги зробити їх копію, подібну до копії файлів електронних таблиць.

6.3 Компонування бази даних

Компонування бази даних потребує ретельного планування. Зауважте, що зміна компоновання таблиці *Persons*, запропонована у розділі *Цілісність і коректність даних*, може призвести до виникнення проблем, якщо таблицю вже заповнено даними. Наприклад, перейменування поля є простим завданням, але розбиття поля **Адреса** на окремі поля потребує ретельного виконання певної нудної роботи.

Щоб уникнути подібних ускладнень, *обдумайте проект вашої бази даних* до її створення на комп'ютері і до того, як ви та інші користувачі почнуть нею користуватися. Таким чином, приділивши компонуванню таблиці певний час до її створення, ви, ймовірно, зекономите час під час щоденного використання цієї бази даних.

6.4 Кому потрібні бази даних?

6.4.1 Зверніться до електронних таблиць, якщо:

- Ваші потреби є досить обмеженими, а об'єм ваших даних ніколи не буде значним (ви можете це точно знати?)
- Вам не вдається осягнути методологію побудови баз даних. Але ви можете перекласти це завдання на когось іншого або скористатися простішими інструментами.

- Ви користуєтеся складними електронними таблицями і не маєте часу або грошей, щоб перейти на бази даних. Обдумайте перехід або порадьтеся з кимось компетентним, щоб не набрити лиха. Не покладайтеся на магичні інструменти, які перетворять вашу електронну таблицю (байдуже якої якості) на базу даних.

6.4.2 Використовуйте бази даних, якщо:

- Ваша збірка даних зростає кожного тижня.
- Вам часто доводиться створювати електронні таблиці, копіюючи до них багато даних, і вам здається, що ця робота стає все нуднішою. У такому разі перехід на роботу з базами даних дуже швидко себе виправдає.
- Ви створюєте звіти та бюлетені, які виходять за межі можливостей електронних таблиць. У такому разі вам слід спробувати скористатися базою даних з формами.

6.5 Програмне забезпечення для створення баз даних

Досі ми обговорювали лише загальні характеристики баз даних без заглиблення у подробиці щодо спеціальних програм для компонування таких баз даних.

Перші бази даних було створено на великих комп'ютерах 60-их років минулого століття, наприклад, на IBM System/360. Це були часи, коли персональних комп'ютерів ще не було, отже обслуговування таких баз даних потребувало високоспеціалізованого персоналу. Хоча тодішнє комп'ютерне обладнання було ненадійним, комп'ютери були незрівнянно повільнішими і мали набагато менший обсяг пам'яті, ніж сучасні, у баз даних вже тоді була беззаперечна перевага: можливість одночасного доступу до них багатьох користувачів за допомогою мережі.

У 1970-их роках науковці розробили теорію реляційних баз даних (зокрема терміни таблиця, запис, стовпчик (поле) і реляційність та багато інших). На цій основі було створено бази даних IBM DB2 і Oracle, які розвиваються і використовуються і до сьогодні. Наприкінці 1970-их з'явилися перші персональні комп'ютери (ПК). Поступово їх користувачі отримали доступ до багатьох типів програмного забезпечення, зокрема і до баз даних.

Якщо йдеться про великі бази даних компаній, ситуація не змінилася: для їх роботи потрібні потужні комп'ютери або комп'ютерні комплекси, які називають кластерами. Але ця тема вже виходить за межі питань, які ми висвітлюємо у цьому підручнику.

Серед «доступних» баз даних з графічним інтерфейсом, призначених для роботи на ПК ви можете обрати одну з таких:

- **DBase** — інструмент для роботи з базами даних у DOS, популярний у 1980-і. Файли у форматі DBase все ще інколи використовуються через їх простоту.
- **FoxPro** — програма, подібна до DBase (початок 1990-их). Після купівлі компанією Microsoft програма отримала графічний інтерфейс користувача, тому використовується для створення баз даних на ПК. Цей програмний продукт все ще продається, хоча вже є трохи застарілим.
- **Microsoft Access** — програма для роботи з базами даних (компонування даних і графічного інтерфейсу) з багатьма зручними спрощеннями, тому придатна для початківців. Цю програму розроблено наприкінці 1980-их на базі 16-біткової архітектури. Цей програмний продукт є дуже поширеним, особливо у невеликих компаніях, де ефективність і можливість одночасного користування декількома користувачами не є критичною.
- **FileMaker** — популярна програма, подібна за простотою до MS Access, працює на платформах Windows і Macintosh, продається з 1985 року.
- **Kexi** — програма, що може працювати на багатьох платформах (UNIX[®]/Linux[®], Windows, Mac[®] OS X), створена у 2003 році. Розробка цієї програми проводиться у межах принципів відкритого програмного забезпечення. Програма є частиною **стілничного середовища KDE**, тобто графічного робочого середовища для систем UNIX[®]/Linux[®]. Значний внесок у розробку Kexi зроблено польською компанією OpenOffice.

Розділ 7

Додаток Б. Порівняння Кехі з іншими програмами для роботи з базами даних

- [Типи даних](#)

7.1 Типи даних

Хоча різні програми для роботи з базами даних надають у розпорядження користувачів майже однакові функціональні можливості, у цих програмах часто використовується зовсім різна термінологія. Для того, щоб вам легше було орієнтуватися, у цьому додатку ми навели таблицю відповідності термінології, яку використано у Кехі, до термінології, що використовується у інших програм для роботи з базами даних. Отже, ця глава буде корисною під час міграції баз даних з одної програми до іншої.

У наведеній нижче таблиці показано відповідність типів даних Кехі типам даних інших програм для роботи з базами даних.

Деякі з наведених типів даних є підтипами інших типів. Наприклад, тип даних *Довгий текст* є підтипом типу *Текст*. Щоб скористатися таким підтипом у Кехі, вам слід спочатку обрати відповідний базовий тип (у нашому прикладі, **Текст**) у дизайнері таблиць, а потім обрати підтип за допомогою пункту **Підтип у Редакторі властивостей**.

Кехі	MS Access	dBase/FoxPro	Paradox
Текст (Текст)	Text	Character	Alphanumeric
Довгий текстовий	Memo	Memo	Memo
Дата/Час (Дата/Час)	Date, Time	Date	DateTime
Ціле число (Ціле число)	Число (Ціле)	Numeric	Integer
Велике ціле число (Велике ціле число)	Long Integer	Numeric	Long Integer
Число з плаваючою комою (Число з плаваючою комою)	Single/Double precision number	Float	Number

Табл. 7.1: Порівняння типів даних у Кехі та інших програмах для роботи з базами даних

Підручник з Кехі

Розділ 8

Додаток В. Зарезервовані слова SQL

Нижче наведено списки слів, які використовуються для внутрішніх потреб Кехі під час обробки джерел даних.

ЗАСТЕРЕЖЕННЯ

Під час створення вашої бази даних вам слід уникати використання зарезервованих слів, оскільки, якщо ви цього не зробите, у вас можуть виникнути проблеми під час роботи з файлом бази даних або навіть проблеми з пошкодженням даних.

ПІДКАЗКА

Якщо вам все ж потрібно використати ці слова, беріть їх у подвійні лапки: "".

8.1 Зарезервовані слова Кехі для SQL

У цьому списку містяться ключові слова, які зарезервовано для використання додатком SQL до Кехі:

- AFTER
- ALL
- ASC
- BEFORE
- BEGIN
- BETWEEN
- BY
- CASCADE
- CASE
- CHECK
- COLLATE
- COMMIT

- CONSTRAINT
- CROSS
- DATABASE
- DEFAULT
- DELETE
- DESC
- DISTINCT
- DROP
- END
- ELSE
- EXPLAIN
- FOR
- FOREIGN
- FULL
- GROUP
- HAVING
- IGNORE
- INDEX
- INNER
- INSERT
- INTO
- KEY
- LIMIT
- MATCH
- NATURAL
- OFFSET
- ORDER
- OUTER
- PRIMARY
- REFERENCES
- REPLACE
- RESTRICT
- ROLLBACK
- ROW
- SET
- TEMPORARY
- THEN
- TRANSACTION
- UNION
- UNIQUE
- UPDATE
- USING
- VALUES
- WHEN

8.2 Зарезервовані слова Кехі для SQLite

У цьому списку містяться ключові слова, які зарезервовано для використання додатком SQLite до Кехі:

- ABORT
- ATTACH
- CLUSTER
- CONFLICT
- DEFERRED
- DEFERRABLE
- DETACH
- EACH
- EXCEPT
- FAIL
- GLOB
- IMMEDIATE
- INITIALLY
- INSTEAD
- INTERSECT
- ISNULL
- NOTNULL
- OF
- PRAGMA
- RAISE
- STATEMENT
- TEMP
- TRIGGER
- VACUUM
- VIEW

8.3 Зарезервовані слова Кехі для MySQL

У цьому списку містяться ключові слова, які зарезервовано для використання додатком MySQL до Кехі:

- ACTION
- ADD
- AGAINST
- AGGREGATE
- ALTER
- ANALYZE
- ANY

- ASCII
- AUTO_INCREMENT
- AVG
- AVG_ROW_LENGTH
- BACKUP
- BDB
- BERKELEYDB
- BIGINT
- BINARY
- BINLOG
- BIT
- BLOB
- BOOL
- BOOLEAN
- BOTH
- BTREE
- BYTE
- CACHE
- CHANGE
- CHANGED
- CHAR
- CHARACTER
- CHARSET
- CHECKSUM
- CIPHER
- CLIENT
- CLOSE
- COLLATION
- COLUMN
- COLUMNS
- COMMENT
- COMMITTED
- COMPRESSED
- CONCURRENT
- CONVERT
- CUBE
- CURRENT_DATE
- CURRENT_TIME
- CURRENT_TIMESTAMP
- CURRENT_USER
- DATA

- DATABASES
- DATE
- DATETIME
- DAY
- DAY_HOUR
- DAY_MICROSECOND
- DAY_MINUTE
- DAY_SECOND
- DEALLOCATE
- DEC
- DECIMAL
- DELAYED
- DELAY_KEY_WRITE
- DESCRIBE
- DES_KEY_FILE
- DIRECTORY
- DISABLE
- DISCARD
- DISTINCTROW
- DIV
- DO
- DOUBLE
- DUAL
- DUMPFILE
- DUPLICATE
- DYNAMIC
- ENABLE
- ENCLOSED
- ENGINE
- ENGINES
- ENUM
- ERRORS
- ESCAPE
- ESCAPED
- EVENTS
- EXECUTE
- EXISTS
- EXPANSION
- EXTENDED
- FALSE
- FAST

- FIELDS
- FILE
- FIRST
- FIXED
- FLOAT
- FLOAT4
- FLOAT8
- FLUSH
- FORCE
- FULLTEXT
- FUNCTION
- GEOMETRY
- GEOMETRYCOLLECTION
- GET_FORMAT
- GLOBAL
- GRANT
- GRANTS
- HANDLER
- HASH
- HELP
- HIGH_PRIORITY
- HOSTS
- HOUR
- HOUR_MICROSECOND
- HOUR_MINUTE
- HOUR_SECOND
- IDENTIFIED
- IF
- IMPORT
- INDEXES
- INFILE
- INNOBASE
- INNODB
- INSERT_METHOD
- INT
- INT1
- INT2
- INT3
- INT4
- INT8
- INTERVAL

- IO_THREAD
- ISOLATION
- ISSUER
- KEYS
- KILL
- LAST
- LEADING
- LEAVES
- LEVEL
- LINES
- LINESTRING
- LOAD
- LOCAL
- LOCALTIME
- LOCALTIMESTAMP
- LOCK
- LOCKS
- LOGS
- LONG
- LONGBLOB
- LONGTEXT
- LOW_PRIORITY
- MASTER
- MASTER_CONNECT_RETRY
- MASTER_HOST
- MASTER_LOG_FILE
- MASTER_LOG_POS
- MASTER_PASSWORD
- MASTER_PORT
- MASTER_SERVER_ID
- MASTER_SSL
- MASTER_SSL_CA
- MASTER_SSL_CAPATH
- MASTER_SSL_CERT
- MASTER_SSL_CIPHER
- MASTER_SSL_KEY
- MASTER_USER
- MAX_CONNECTIONS_PER_HOUR
- MAX_QUERIES_PER_HOUR

- MAX_ROWS
- MAX_UPDATES_PER_HOUR
- MEDIUM
- MEDIUMBLOB
- MEDIUMINT
- MEDIUMTEXT
- MICROSECOND
- MIDDLEINT
- MINUTE
- MINUTE_MICROSECOND
- MINUTE_SECOND
- MIN_ROWS
- MOD
- MODE
- MODIFY
- MONTH
- MULTILINESTRING
- MULTIPOINT
- MULTIPOLYGON
- NAMES
- NATIONAL
- NDB
- NDBCLUSTER
- NCHAR
- NEW
- NEXT
- NO
- NONE
- NO_WRITE_TO_BINLOG
- NUMERIC
- NVARCHAR
- OLD_PASSWORD
- ONE_SHOT
- OPEN
- OPTIMIZE
- OPTION
- OPTIONALLY
- OUTFILE
- PACK_KEYS
- PARTIAL

- PASSWORD
- POINT
- POLYGON
- PRECISION
- PREPARE
- PREV
- PRIVILEGES
- PROCEDURE
- PROCESS
- PROCESSLIST
- PURGE
- QUERY
- QUICK
- RAID0
- RAID_CHUNKS
- RAID_CHUNKSIZE
- RAID_TYPE
- READ
- REAL
- REGEXP
- RELAY_LOG_FILE
- RELAY_LOG_POS
- RELAY_THREAD
- RELOAD
- RENAME
- REPAIR
- REPEATABLE
- REPLICATION
- REQUIRE
- RESET
- RESTORE
- RETURNS
- REVOKE
- RLIKE
- ROLLUP
- ROWS
- ROW_FORMAT
- RTREE
- SAVEPOINT
- SECOND

- SECOND_MICROSECOND
- SEPARATOR
- SERIAL
- SERIALIZABLE
- SESSION
- SHARE
- SHOW
- SHUTDOWN
- SIGNED
- SIMPLE
- SLAVE
- SMALLINT
- SOME
- SONAME
- SOUNDS
- SPATIAL
- SQL_BIG_RESULT
- SQL_BUFFER_RESULT
- SQL_CACHE
- SQL_CALC_FOUND_ROWS
- SQL_NO_CACHE
- SQL_SMALL_RESULT
- SQL_THREAD
- SSL
- START
- STARTING
- STATUS
- STOP
- STORAGE
- STRAIGHT_JOIN
- STRING
- STRIPED
- SUBJECT
- SUPER
- TABLES
- TABLESPACE
- TERMINATED
- TEXT
- TIME
- TIMESTAMP

- TINYBLOB
- TINYINT
- TINYTEXT
- TRAILING
- TRUE
- TRUNCATE
- TYPE
- TYPES
- UNCOMMITTED
- UNICODE
- UNLOCK
- UNSIGNED
- UNTIL
- USAGE
- USE
- USER
- USER_RESOURCES
- USE_FRM
- UTC_DATE
- UTC_TIME
- UTC_TIMESTAMP
- VALUE
- VARBINARY
- VARCHAR
- VARCHARACTER
- VARIABLES
- VARYING
- WARNINGS
- WITH
- WORK
- WRITE
- X509
- YEAR
- YEAR_MONTH
- ZEROFILL

8.4 Зарезервовані слова Кехі для PostgreSQL

У цьому списку містяться ключові слова, які зарезервовано для використання драйвером рқхх Кехі:

- ABORT
- ABSOLUTE
- ACCESS
- ACTION
- ADD
- AGGREGATE
- ALTER
- ANALYSE
- ANALYZE
- ANY
- ARRAY
- ASSERTION
- ASSIGNMENT
- AT
- AUTHORIZATION
- BACKWARD
- BIGINT
- BINARY
- BIT
- BOOLEAN
- BOTH
- CACHE
- CALLED
- CAST
- CHAIN
- CHAR
- CHARACTER
- CHARACTERISTICS
- CHECKPOINT
- CLASS
- CLOSE
- CLUSTER
- COALESCE
- COLUMN
- COMMENT
- COMMITTED
- CONSTRAINTS
- CONVERSION

- CONVERT
- COPY
- CREATEDB
- CREATEUSER
- CURRENT_DATE
- CURRENT_TIME
- CURRENT_TIMESTAMP
- CURRENT_USER
- CURSOR
- CYCLE
- DAY
- DEALLOCATE
- DEC
- DECIMAL
- DECLARE
- DEFAULTS
- DEFERRABLE
- DEFERRED
- DEFINER
- DELIMITER
- DELIMITERS
- DO
- DOMAIN
- DOUBLE
- EACH
- ENCODING
- ENCRYPTED
- ESCAPE
- EXCEPT
- EXCLUDING
- EXCLUSIVE
- EXECUTE
- EXISTS
- EXTERNAL
- EXTRACT
- FALSE
- FETCH
- FIRST
- FLOAT
- FORCE
- FORWARD

- FREEZE
- FUNCTION
- GLOBAL
- GRANT
- HANDLER
- HOLD
- HOUR
- ILIKE
- IMMEDIATE
- IMMUTABLE
- IMPLICIT
- INCLUDING
- INCREMENT
- INHERITS
- INITIALLY
- INOUT
- INPUT
- INSENSITIVE
- INSTEAD
- INT
- INTERSECT
- INTERVAL
- INVOKER
- ISNULL
- ISOLATION
- LANCOMPILER
- LANGUAGE
- LAST
- LEADING
- LEVEL
- LISTEN
- LOAD
- LOCAL
- LOCALTIME
- LOCALTIMESTAMP
- LOCATION
- LOCK
- MAXVALUE
- MINUTE
- MINVALUE
- MODE
- MONTH

- MOVE
- NAMES
- NATIONAL
- NCHAR
- NEW
- NEXT
- NO
- NOCREATEDB
- NOCREATEUSER
- NONE
- NOTHING
- NOTIFY
- NOTNULL
- NULLIF
- NUMERIC
- OF
- OFF
- OIDS
- OLD
- ONLY
- OPERATOR
- OPTION
- OUT
- OVERLAPS
- OVERLAY
- OWNER
- PARTIAL
- PASSWORD
- PATH
- PENDANT
- PLACING
- POSITION
- PRECISION
- PREPARE
- PRESERVE
- PRIOR
- PRIVILEGES
- PROCEDURAL
- PROCEDURE
- READ
- REAL
- RECHECK

- REINDEX
- RELATIVE
- RENAME
- RESET
- RESTART
- RETURNS
- REVOKE
- ROWS
- RULE
- SCHEMA
- SCROLL
- SECOND
- SECURITY
- SEQUENCE
- SERIALIZABLE
- SESSION
- SESSION_USER
- SETOF
- SHARE
- SHOW
- SIMPLE
- SMALLINT
- SOME
- STABLE
- START
- STATEMENT
- STATISTICS
- STDIN
- STDOUT
- STORAGE
- STRICT
- SUBSTRING
- SYSID
- TEMP
- TEMPLATE
- TIME
- TIMESTAMP
- TOAST
- TRAILING
- TREAT
- TRIGGER

- TRIM
- TRUE
- TRUNCATE
- TRUSTED
- TYPE
- UNENCRYPTED
- UNKNOWN
- UNLISTEN
- UNTIL
- USAGE
- USER
- VACUUM
- VALID
- VALIDATOR
- VARCHAR
- VARYING
- VERBOSE
- VERSION
- VIEW
- VOLATILE
- WITH
- WITHOUT
- WORK
- WRITE
- YEAR
- ZONE

8.5 Зарезервовані слова Кехі для драйвера Oracle

У цьому списку містяться ключові слова, які зарезервовано для використання драйвером Oracle Кехі:

- ADMIN
- AFTER
- ALLOCATE
- ANALYZE
- ARCHIVE
- ARCHIVELOG
- AUTHORIZATION
- AVG
- BACKUP
- BECOME

- BEFORE
- BEGIN
- BLOCK
- BODY
- CACHE
- CANCEL
- CASCADE
- CHANGE
- CHARACTER
- CHECKPOINT
- CLOSE
- COBOL
- COMMIT
- COMPILE
- CONSTRAINT
- CONSTRAINTS
- CONTENTS
- CONTINUE
- CONTROLFILE
- COUNT
- CURSOR
- CYCLE
- DATABASE
- DATAFILE
- DATE
- DBA
- DEC
- DECLARE
- DISABLE
- DISMOUNT
- DOUBLE
- DUMP
- EACH
- ENABLE
- END
- ESCAPE
- EVENTS
- EXCEPT
- EXCEPTIONS
- EXEC
- EXECUTE
- EXPLAIN

- EXTENT
- EXTERNALLY
- FETCH
- FLUSH
- FORCE
- FOREIGN
- FORTRAN
- FOUND
- FREELIST
- FREELISTS
- FUNCTION
- GO
- GOTO
- GROUPS
- INCLUDING
- INDICATOR
- INTRANS
- INSTANCE
- INT
- KEY
- LANGUAGE
- LAYER
- LINK
- LISTS
- LOGFILE
- MANAGE
- MANUAL
- MAX
- MAXDATAFILES
- MAXINSTANCES
- MAXLOGFILES
- MAXLOGHISTORY
- MAXLOGMEMBERS
- MAXTRANS
- MAXVALUE
- MIN
- MINEXTENTS
- MINVALUE
- MODULE
- MOUNT
- NEW
- NEXT

- NOARCHIVELOG
- NOCACHE
- NOCYCLE
- NOMAXVALUE
- NOMINVALUE
- NONE
- NOORDER
- NORESETLOGS
- NORMAL
- NOSORT
- NUMERIC
- OFF
- OLD
- ONLY
- OPEN
- OPTIMAL
- OWN
- PACKAGE
- PARALLEL
- PCTINCREASE
- PCTUSED
- PLAN
- PLI
- PRECISION
- PRIMARY
- PRIVATE
- PROCEDURE
- PROFILE
- QUOTA
- READ
- REAL
- RECOVER
- REFERENCES
- REFERENCING
- RESETLOGS
- RESTRICTED
- REUSE
- ROLE
- ROLES
- ROLLBACK
- SAVEPOINT

- SCHEMA
- SCN
- SECTION
- SEGMENT
- SEQUENCE
- SHARED
- SNAPSHOT
- SOME
- SORT
- SQL
- SQLCODE
- SQLERROR
- SQLSTATE
- STATEMENT_ID
- STATISTICS
- STOP
- STORAGE
- SUM
- SWITCH
- SYSTEM
- TABLES
- TABLESPACE
- TEMPORARY
- THREAD
- TIME
- TRACING
- TRANSACTION
- TRIGGERS
- TRUNCATE
- UNDER
- UNLIMITED
- UNTIL
- USE
- USING
- WHEN
- WORK
- WRITE

8.6 Зарезервовані слова Кехі для драйвера Sybase

У цьому списку містяться ключові слова, які зарезервовано для використання драйвером Sybase Кехі:

- ACTION
- ADD
- AGAINST
- AGGREGATE
- ALTER
- ANALYZE
- ANY
- ASCII
- AUTOINCREMENT
- AVG
- AVG_ROW_LENGTH
- BACKUP
- BDB
- BERKELEYDB
- BIGINT
- BINARY
- BINLOG
- BIT
- BLOB
- BOOL
- BOOLEAN
- BOTH
- BTREE
- BYTE
- CACHE
- CHANGE
- CHANGED
- CHAR
- CHARACTER
- CHARSET
- CHECKSUM
- CIPHER
- CLIENT
- CLOSE
- COLLATION
- COLUMN
- COLUMNS
- COMMENT

- COMMITTED
- COMPRESSED
- CONCURRENT
- CONVERT
- CUBE
- CURRENT_DATE
- CURRENT_TIME
- CURRENT_TIMESTAMP
- CURRENT_USER
- DATA
- DATABASES
- DATE
- DATETIME
- DAY
- DAY_HOUR
- DAY_MICROSECOND
- DAY_MINUTE
- DAY_SECOND
- DEALLOCATE
- DEC
- DECIMAL
- DELAYED
- DELAY_KEY_WRITE
- DESCRIBE
- DES_KEY_FILE
- DIRECTORY
- DISABLE
- DISCARD
- DISTINCTROW
- DIV
- DO
- DOUBLE
- DUAL
- DUMPFILE
- DUPLICATE
- DYNAMIC
- ENABLE
- ENCLOSED
- ENGINE
- ENGINES

- ENUM
- ERRORS
- ESCAPE
- ESCAPED
- EVENTS
- EXECUTE
- EXISTS
- EXPANSION
- EXTENDED
- FALSE
- FAST
- FIELDS
- FILE
- FIRST
- FIXED
- FLOAT
- FLOAT4
- FLOAT8
- FLUSH
- FORCE
- FULLTEXT
- FUNCTION
- GEOMETRY
- GEOMETRYCOLLECTION
- GET_FORMAT
- GLOBAL
- GRANT
- GRANTS
- HANDLER
- HASH
- HELP
- HIGH_PRIORITY
- HOSTS
- HOUR
- HOUR_MICROSECOND
- HOUR_MINUTE
- HOUR_SECOND
- IDENTIFIED
- IF
- IMPORT
- INDEXES

- INFILE
- INNOBASE
- INNODB
- INSERT_METHOD
- INT
- INT1
- INT2
- INT3
- INT4
- INT8
- INTERVAL
- IO_THREAD
- ISOLATION
- ISSUER
- KEYS
- KILL
- LAST
- LEADING
- LEAVES
- LEVEL
- LINES
- LINESTRING
- LOAD
- LOCAL
- LOCALTIME
- LOCALTIMESTAMP
- LOCK
- LOCKS
- LOGS
- LONG
- LONGBLOB
- LONGTEXT
- LOW_PRIORITY
- MASTER
- MASTER_CONNECT_RETRY
- MASTER_HOST
- MASTER_LOG_FILE
- MASTER_LOG_POS
- MASTER_PASSWORD
- MASTER_PORT

- MASTER_SERVER_ID
- MASTER_SSL
- MASTER_SSL_CA
- MASTER_SSL_CAPATH
- MASTER_SSL_CERT
- MASTER_SSL_CIPHER
- MASTER_SSL_KEY
- MASTER_USER
- MAX_CONNECTIONS_PER_HOUR
- MAX_QUERIES_PER_HOUR
- MAX_ROWS
- MAX_UPDATES_PER_HOUR
- MEDIUM
- MEDIUMBLOB
- MEDIUMINT
- MEDIUMTEXT
- MICROSECOND
- MIDDLEINT
- MINUTE
- MINUTE_MICROSECOND
- MINUTE_SECOND
- MIN_ROWS
- MOD
- MODE
- MODIFY
- MONTH
- MULTILINESTRING
- MULTIPOINT
- MULTIPOLYGON
- NAMES
- NATIONAL
- NDB
- NDBCLUSTER
- NCHAR
- NEW
- NEXT
- NO
- NONE
- NO_WRITE_TO_BINLOG

- NUMERIC
- NVARCHAR
- OLD_PASSWORD
- ONE_SHOT
- OPEN
- OPTIMIZE
- OPTION
- OPTIONALLY
- OUTFILE
- PACK_KEYS
- PARTIAL
- PASSWORD
- POINT
- POLYGON
- PRECISION
- PREPARE
- PREV
- PRIVILEGES
- PROCEDURE
- PROCESS
- PROCESSLIST
- PURGE
- QUERY
- QUICK
- RAID0
- RAID_CHUNKS
- RAID_CHUNKSIZE
- RAID_TYPE
- READ
- REAL
- REGEXP
- RELAY_LOG_FILE
- RELAY_LOG_POS
- RELAY_THREAD
- RELOAD
- RENAME
- REPAIR
- REPEATABLE
- REPLICATION
- REQUIRE

- RESET
- RESTORE
- RETURNS
- REVOKE
- RLIKE
- ROLLUP
- ROWS
- ROW_FORMAT
- RTREE
- SAVEPOINT
- SECOND
- SECOND_MICROSECOND
- SEPARATOR
- SERIAL
- SERIALIZABLE
- SESSION
- SHARE
- SHOW
- SHUTDOWN
- SIGNED
- SIMPLE
- SLAVE
- SMALLINT
- SOME
- SONAME
- SOUNDS
- SPATIAL
- SQL_BIG_RESULT
- SQL_BUFFER_RESULT
- SQL_CACHE
- SQL_CALC_FOUND_ROWS
- SQL_NO_CACHE
- SQL_SMALL_RESULT
- SQL_THREAD
- SSL
- START
- STARTING
- STATUS
- STOP
- STORAGE

- STRAIGHT_JOIN
- STRING
- STRIPED
- SUBJECT
- SUPER
- TABLES
- TABLESPACE
- TERMINATED
- TEXT
- TIME
- TIMESTAMP
- TINYBLOB
- TINYINT
- TINYTEXT
- TRAILING
- TRUE
- TRUNCATE
- TYPE
- TYPES
- UNCOMMITTED
- UNICODE
- UNLOCK
- UNSIGNED
- UNTIL
- USAGE
- USE
- USER
- USER_RESOURCES
- USE_FRM
- UTC_DATE
- UTC_TIME
- UTC_TIMESTAMP
- VALUE
- VARBINARY
- VARCHAR
- VARCHARACTER
- VARIABLES
- VARYING
- WARNINGS
- WITH
- WORK

- WRITE
- X509
- YEAR
- YEAR_MONTH
- ZEROFILL

8.7 Зарезервовані слова Кехі для драйвера xBase

У цьому списку містяться ключові слова, які зарезервовано для використання драйвером xBase Кехі:

- ABORT
- ATTACH
- CLUSTER
- CONFLICT
- DEFERRED
- DEFERRABLE
- DETACH
- EACH
- EXCEPT
- FAIL
- GLOB
- IMMEDIATE
- INITIALLY
- INSTEAD
- INTERSECT
- ISNULL
- NOTNULL
- OF
- PRAGMA
- RAISE
- STATEMENT
- TEMP
- TRIGGER
- VACUUM
- VIEW

ПРИМІТКА

Зарезервовані слова наведено окремо для кожного з драйверів з метою використання цього списку як довідника.

Розділ 9

Додаток Г. Підтримувані формати файлів

9.1 Формат значень, відокремлених комами (CSV)

Кехі може імпортувати і експортувати дані у широкому діапазоні форматів значень, відокремлених певним знаками (файли CSV). Дані у таких форматах здатні імпортувати та експортувати більшість програм для роботи з електронними таблицями. Отже, такий формат є зручним для обміну текстовими даними між різними програмами.

Крім того, у Кехі передбачено можливість внесення змін у декілька параметрів запису даних до виконання імпортування. Зокрема, ви можете змінити:

- тип даних у кожному стовпчику,
- визначити інший роздільник, зокрема символ табуляції,
- визначити лапки, у яких записано текст,
- вказати кількість записів, які слід пропускати, якщо це потрібно,
- наказати програмі пропускати подвійні роздільники,
- наказати програмі використовувати значення з першого рядка як назви стовпчиків,
- вказати кодування символів (типовим кодуванням є UTF-8),
- вказати формат дати (типово визначається за параметрами операційної системи),
- наказати програмі вилучити початкові та кінцеві пробіли з текстових значень.

Під час імпортування до Кехі ви зможете попередньо переглянути імпортовані дані. Оптимальний набір параметрів автоматично визначатиметься Кехі на основі наданого файла CSV.

9.2 Формат файлів Microsoft Access (MDB)

9.2.1 Огляд

Кехі має вбудовану підтримку імпортування баз даних **Microsoft Access** (2003 та старіших версій). Передбачено лише можливість *імпортування* баз даних **Access** до баз даних Кехі. Програмою не можна скористатися для редагування бази даних або експортування даних у форматі **Access**. Втім, програма здатна імпортувати бази даних, створені за допомогою

інших програм, що використовують рушій роботи з базами даних JET. У поточній версії не передбачено можливості імпортування баз даних .accdb, створених за допомогою MS Access 2007 та пізніших версій.

Щоб імпортувати базу даних,

1. У меню Кехі виберіть пункт **Імпортувати, експортувати або надіслати...**
2. Натисніть кнопку **Імпортувати базу даних**.
3. Скористайтеся майстром імпортування, вибравши потрібний вам файл .mdb.

9.2.2 Можливості

Можливість імпортування було успішно протестовано на базі даних *Northwind*, добре відомій користувачам MS Access. Також виконано перевірку імпортування багатьох інших шаблонних баз даних Access.

9.2.3 Підтримувані можливості

Імпортування таких типів полів перевірено, вважається, що результати є задовільними:

- Текстові поля
- Поле нотаток
- Поля дат
- Числові значення

9.2.4 Непідтримувані можливості

У поточній версії імпортуються лише таблиці. Під час імпортування Кехі пропускає такі об'єкти:

- Запити
- Форми
- Скрипти
- Звіти

Імпортування цих компонентів ще недостатньо перевірено:

- Двійкові об'єкти

Розділ 10

Подяки і ліцензія

Авторські права на програму Кехі належать Команді Кехі, 2002–2012
Розробники Кехі

- Jaroslaw Staniek staniek@kde.org
- OpenOffice Polska, LLC info@openoffice.com.pl
- Lucijan Busch lucijan@kde.org
- Cedric Pasteur cedric.pasteur@free.fr
- Adam Pigg adam@piggy.fsnet.co.uk
- Martin Ellis martin.ellis@kdemail.net
- Sebastian Sauer mail@dipe.org
- Christian Nitschkowski segfault_ii@web.de
- Peter Simonsson psn@linux.se
- Joseph Wenninger jowenn@kde.org
- Seth Kurzenberg seth@cql.com
- Laurent Montel montel@kde.org
- Till Busch till@bux.at

Авторами документації є Martin A. Ellis martin.ellis@kdemail.net, Jaroslaw Staniek staniek@kde.org, окремий внесок зробили Anne-Marie Mahfouf, Raphael Langerhorst, Michal Kubicki і Aron Stansvik.

Програма поширюється за умов дотримання [GNU Lesser General Public License](#).

Переклад українською: Юрій Чорноіван yurchor@ukr.net

Цей документ поширюється за умов дотримання [GNU Free Documentation License](#).