Министерство образования Самарской области

АССОЦИАЦИЯ ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ "ИНСТИТУТ НАПРАВО.РУ"

УТВЕРЖДЕНО
Приказ директора
от 07.08.2024 № 012/2/с

Воробьев К.А.

УЧЕБНО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ

ОП.06 Основы проектирования баз данных

Специальность 09.02.10 Разработка компьютерных игр, дополненной и виртуальной реальности
Квалификация: разработчик компьютерных игр, дополненной и виртуальной реальности
Форма обучения: очная
На базе основного общего образования
Срок получения образования: 3 года 10 месяцев

Тольятти, 2024г.

Содержание

1. Назначение и организация практикума 3

2. Состав работ и представление результатов 3

3. Практические работы 3

Работа 1. Сущности и связи 3

Работа 2. Ограничения целостности 4

Работа 3. Выборки и соединение 5

Работа 4. Транзакция передачи предмета 5

Работа 5. Нормализация таблицы 6

Работа 6. Резервная копия и восстановление 7

Развёрнутая лабораторная работа. SQL-практикум с восстановлением исходной базы 7

4. Комплексное учебное задание 8

5. Термины и понятия 9

6. Исходные данные и учебные примеры 9

7. Использованные источники 9

1. Назначение и организация практикума

Проектирование и реализация реляционной базы данных учебного игрового проекта.

Указания предназначены для обучающихся по специальности 09.02.10 «Разработка компьютерных игр, дополненной и виртуальной реальности», квалификация — разработчик компьютерных игр, дополненной и виртуальной реальности.

Компонент	Учебная нагрузка
Индекс и наименование	ОП.06 Основы проектирования баз данных
Общий объём	98 академических часов
Практические занятия	74 академических часов
Промежуточная аттестация	Экзамен — семестр 4

Перед работой изучаются её исходные данные и проверяемые результаты. Затем выполняются последовательные действия, сопоставляется учебный пример и решаются самостоятельные варианты. Ответ включает ход работы и подтверждение результата. Использование готового примера без выполнения своего варианта не подтверждает освоение действия.

Подготовленные материалы: Диаграмма сущностей; SQL создания и наполнения; Протокол запросов, отказов и восстановления.

2. Состав работ и представление результатов

Работа	Тема и результат
1	Сущности и связи. Один предмет может принадлежать разным игрокам.
2	Ограничения целостности. Ожидаемые отказы показаны в журнале.
3	Выборки и соединение. Фильтр А возвращает только Ключ игрока 1.
4	Транзакция передачи предмета. В А операция отклоняется без изменения данных.
5	Нормализация таблицы. Цена справочника и цена конкретной покупки не смешиваются.
6	Резервная копия и восстановление. Процедура воспроизводится без исходного открытого соединения.

В отчёте указываются тема, вариант, исходные данные, ход выполнения, проверка результата и вывод. Таблицы и графики снабжаются названиями и единицами измерения, где они применимы. Результаты оцениваются по критериям фонда оценочных средств. Приложите SQL-запросы, сценарий восстановления и результаты контрольных проверок.

3. Практические работы

Работа 1. Сущности и связи

Краткие теоретические сведения

Сущность описывает класс объектов, атрибут — характеристику, связь — отношение между объектами. Повторяющаяся группа значений в одной ячейке затрудняет ограничения и запросы.

Исходные данные

Нужно хранить игроков, предметы и количество предметов у каждого игрока.

Порядок выполнения

1. Выделить три сущности: Player, Item, Inventory.

2. Назначить первичные ключи и типы атрибутов.

3. Указать кардинальности и обязательность связей.

4. Проверить, где хранить количество одного предмета.

Разбор учебного примера

Inventory содержит player_id, item_id и qty; пара player_id,item_id уникальна. Количество относится к владению, а не к справочнику Item.

Задания для самостоятельного выполнения

А: добавить редкость предмета.

Б: добавить историю выдачи предметов.

Проверка результата

— Один предмет может принадлежать разным игрокам.

— qty не хранится в Item как глобальное количество игрока.

Вопросы для самоконтроля

1. Зачем нужна промежуточная таблица?

2. Что произойдёт при списке предметов в одной строке?

Работа 2. Ограничения целостности

Краткие теоретические сведения

Ограничения проверяют не только существование столбца, но и допустимые значения и связи. Внешний ключ предотвращает ссылки на отсутствующий объект.

Исходные данные

Player: (1, Альфа), (2, Бета). Item: (10, Ключ), (20, Щит). Inventory: (1, 10, 2), (1, 20, 1), (2, 10, 1).

Порядок выполнения

1. Создать таблицы по приведённой схеме из приложения.

2. Включить контроль внешних ключей.

3. Добавить исходные строки.

4. Проверить запрет отрицательного количества, повторной пары и отсутствующего игрока.

Разбор учебного примера

Записи с qty=−1, повтором (1,10) и player_id=99 должны быть отклонены. Допустимая вставка (2,20,1) проходит.

Задания для самостоятельного выполнения

А: разрешить нулевое количество.

Б: запретить пустое имя предмета.

Проверка результата

— Ожидаемые отказы показаны в журнале.

— Отказ ограничения не обходится отключением контроля.

Вопросы для самоконтроля

1. Чем NOT NULL отличается от UNIQUE?

2. Что проверяет внешний ключ?

Работа 3. Выборки и соединение

Краткие теоретические сведения

Соединение связывает записи по ключам; неверное условие создаёт лишние сочетания. Группировка нужна для агрегирования, а не для замены корректного соединения.

Исходные данные

Используется база предыдущей работы до дополнительной вставки.

Порядок выполнения

1. Получить предметы игрока 1.

2. Соединить Inventory с Item по item_id.

3. Подсчитать суммарное количество по игрокам.

4. Отдельно проверить игрока без записей инвентаря.

Разбор учебного примера

Для игрока 1 выводятся Ключ — 2 и Щит — 1. Суммы: игрок 1 — 3, игрок 2 — 1. Для сохранения игрока без предметов применяется LEFT JOIN и обработка отсутствующей суммы.

Задания для самостоятельного выполнения

А: выбрать только qty>1.

Б: добавить игрока 3 без предметов.

Проверка результата

— Фильтр А возвращает только Ключ игрока 1.

— В варианте Б игрок 3 не должен исчезать из полного отчёта.

Вопросы для самоконтроля

1. Почему соединение без условия ошибочно?

2. Когда необходим LEFT JOIN?

Работа 4. Транзакция передачи предмета

Краткие теоретические сведения

Несколько связанных изменений образуют одну операцию. Транзакция должна завершаться полностью либо откатываться; частичная передача нарушает сохранение количества.

Исходные данные

Передать один Ключ от игрока 1 игроку 2; начальные количества 2 и 1.

Порядок выполнения

1. Зафиксировать начальный суммарный остаток.

2. Начать транзакцию и проверить наличие предмета.

3. Уменьшить первый остаток и увеличить второй.

4. Вызвать учебную ошибку между изменениями и проверить откат.

Разбор учебного примера

После успешной передачи остатки 1 и 2, сумма 3. При искусственном сбое после уменьшения откат восстанавливает 2 и 1.

Задания для самостоятельного выполнения

А: передать три предмета при наличии двух.

Б: выполнить успешную передачу и перезапустить соединение.

Проверка результата

— В А операция отклоняется без изменения данных.

— В Б подтверждённые изменения сохраняются.

Вопросы для самоконтроля

1. Что означает атомарность?

2. Почему проверка суммы полезна, но не заменяет проверку владельцев?

Работа 5. Нормализация таблицы

Краткие теоретические сведения

Функциональная зависимость показывает, от каких данных определяется атрибут. Дублирование имени предмета в каждой записи владения вызывает аномалии обновления.

Исходные данные

Таблица: player_id, player_name, item_id, item_name, qty; составной ключ player_id,item_id.

Порядок выполнения

1. Указать зависимости имени игрока и предмета.

2. Показать аномалию переименования.

3. Разделить таблицу на справочники и связь.

4. Проверить восстановление представления соединением.

Разбор учебного примера

player_name зависит только от player_id, item_name — от item_id. После выделения Player и Item имя меняется в одном месте, а количество остаётся в Inventory.

Задания для самостоятельного выполнения

А: добавить цену предмета.

Б: добавить дату конкретной покупки.

Проверка результата

— Цена справочника и цена конкретной покупки не смешиваются.

— Обратное соединение не создаёт дополнительных строк.

Вопросы для самоконтроля

1. Как отличить атрибут связи от атрибута сущности?

2. Какая аномалия возникает при удалении последнего владельца?

Работа 6. Резервная копия и восстановление

Краткие теоретические сведения

Копия полезна только при подтверждённом восстановлении. Проверяют структуру, число записей, ограничения и контрольные запросы.

Исходные данные

База с двумя игроками и тремя записями Inventory.

Порядок выполнения

1. Создать копию средствами выбранной СУБД.

2. Восстановить в отдельную учебную базу.

3. Сравнить количество строк и результаты запросов.

4. Проверить ограничения восстановленной схемы.

Разбор учебного примера

Совпадение размера файла не является проверкой. Восстановленная база должна вернуть те же суммы 3 и 1 и отклонить недопустимый внешний ключ.

Задания для самостоятельного выполнения

А: удалить одну строку в копии и выявить различие.

Б: сохранить SQL-скрипт создания и наполнения с нуля.

Проверка результата

— Процедура воспроизводится без исходного открытого соединения.

— В отчёте есть контрольные результаты до и после.

Вопросы для самоконтроля

1. Почему резервную копию проверяют в отдельной базе?

2. Что нужно сохранить кроме данных?

Развёрнутая лабораторная работа. SQL-практикум с восстановлением исходной базы

Средства выполнения

SQLite и видимый SQL-скрипт inventory.sql из этого документа. Все записи учебные.

Задание

Создать базу инвентаря, выполнить выборки и контролируемую передачу, доказать работу ограничений и восстановление из копии.

Последовательность выполнения

1. Создайте новую пустую базу и выполните скрипт. Не запускайте CREATE TABLE повторно поверх уже заполненной базы, выдавая ошибку существующей таблицы за дефект задания.

2. Проверьте включение внешних ключей для текущего соединения. Выполните оба

контрольных запроса и сопоставьте результат с приведёнными эталонами.

3. В отдельной копии попробуйте три недопустимые вставки: отрицательное qty, неизвестный player_id и повтор пары ключей. Зафиксируйте отказ каждого ограничения.

4. Добавьте игрока 3 без предметов и убедитесь, что LEFT JOIN возвращает его с нулевым итогом. Объясните различие с INNER JOIN.

5. Выполните передачу одного предмета «Ключ» от Альфы к Бете внутри транзакции. Сначала проверьте наличие; затем измените оба остатка и подтвердите транзакцию. Итоговые количества должны быть 1 и 2.

6. Верните исходное состояние и повторите операцию с искусственным отказом между изменениями. Используйте откат; результат должен совпасть с исходным состоянием 2 и 1.

7. Сформируйте отчёт по предметам и владельцам, добавьте сортировку и фильтр qty>1. Для каждого запроса заранее запишите ожидаемые строки.

8. Подготовьте SQL-скрипт восстановления и испытайте его в ещё одной пустой базе. Сравните структуру, данные и отрицательные проверки, а не только размер файла.

Контрольные сценарии и ожидаемые результаты

Сценарий	Ожидаемый результат
Исходный итог Альфы	3
Исходный итог Беты	1
Выборка Альфы	Ключ=2; Щит=1
Игрок без предметов	Не исчезает из полного отчёта
Успешная передача	1 и 2, сумма 3
Откат после отказа	2 и 1, сумма 3
Отрицательное qty	Вставка отклонена

Усложнённый вариант

Добавьте журнал передачи с уникальным идентификатором операции. Продумайте обработку повторного запроса, чтобы одна передача не была исполнена дважды. Приведите два положительных и два отрицательных теста.

4. Комплексное учебное задание

Создайте базу коллекции игровых достижений с игроками, достижениями и датами получения. Включите ограничения, минимум четыре запроса, одну транзакцию и сценарий восстановления. Исходные данные — не менее пяти игроков и пяти достижений; заранее задайте ожидаемые результаты каждого запроса.

Этапы и предъявляемые материалы

Этап	Результат
1	Диаграмма сущностей
2	SQL создания и наполнения
3	Протокол запросов, отказов и восстановления

Выполненная работа представляется с исходными данными, объяснением принятых решений и проверками. Изменения исходного условия согласуются до выполнения расчётов или реализации. При защите обучающийся воспроизводит один основной и один граничный либо отрицательный сценарий и объясняет полученный результат.

Контроль готовности

— Один предмет может принадлежать разным игрокам.

— Ожидаемые отказы показаны в журнале.

— Фильтр А возвращает только Ключ игрока 1.

— В А операция отклоняется без изменения данных.

— Цена справочника и цена конкретной покупки не смешиваются.

— Процедура воспроизводится без исходного открытого соединения.

5. Термины и понятия

Термин	Содержание
Первичный ключ	Минимальный выбранный идентификатор записи.
Внешний ключ	Ссылка на допустимую запись связанной таблицы.
Транзакция	Единица согласованного изменения данных.
Нормализация	Преобразование структуры для устранения зависимостей, вызывающих аномалии.

6. Исходные данные и учебные примеры

Сценарий inventory.sql создаёт учебную базу SQLite и заполняет её данными для практических работ. Запросы выполняются на созданной базе.

inventory.sql

```
PRAGMA foreign_keys = ON;
CREATE TABLE Player (
  id INTEGER PRIMARY KEY,
  name TEXT NOT NULL
);
CREATE TABLE Item (
  id INTEGER PRIMARY KEY,
  name TEXT NOT NULL
);
CREATE TABLE Inventory (
  player_id INTEGER NOT NULL REFERENCES Player(id),
  item_id INTEGER NOT NULL REFERENCES Item(id),
  qty INTEGER NOT NULL CHECK(typeof(qty)='integer' AND qty >= 0),
  PRIMARY KEY (player_id, item_id)
);
INSERT INTO Player VALUES (1, 'Альфа'), (2, 'Бета');
INSERT INTO Item VALUES (10, 'Ключ'), (20, 'Щит');
INSERT INTO Inventory VALUES (1,10,2), (1,20,1), (2,10,1);
SELECT p.name, COALESCE(SUM(i.qty),0) AS total
FROM Player AS p LEFT JOIN Inventory AS i ON i.player_id=p.id
GROUP BY p.id, p.name ORDER BY p.id;
SELECT t.name, i.qty FROM Inventory AS i
JOIN Item AS t ON t.id=i.item_id
WHERE i.player_id=1 ORDER BY t.id;
```

Скрипт выполняется в новой учебной базе SQLite. Первый запрос должен вернуть Альфа — 3, Бета — 1; второй — Ключ — 2, Щит — 1. После этого выполняются отрицательные проверки ограничений из работы 2.

7. Использованные источники

1. Ткаченко, С. Н., Основы проектирования баз данных: учебник / С. Н. Ткаченко. — Москва: КноРус, 2024. — 176 с. — ISBN 978-5-406-12054-5. — URL:

https://book.ru/book/950600

2. Кумскова, И. А., Базы данных: учебник / И. А. Кумскова. — Москва: КноРус, 2024. — 400 с. — ISBN 978-5-406-12899-2. — URL: https://book.ru/book/952917

3. Крикунов М. М., Поручиков А. Н. Основы баз данных. Самара: Издательство Самарского университета,

4. 84 с. URL: https://repo.ssau.ru/jspui/bitstream/123456789/56990/1/Крикунов М.М. Основы баз данных 2021.pdf