Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ

УНИВЕРСИТЕТ»

«Проектирование и разработка баз данных интернет-приложений»

Отчёт по лабораторной работе №2

Реализация базы данных.

Выполнил: Подобед В.Г.

ФИТ 3 курс 6 группа

Преподаватель: Нистюк О.А.

Минск 2024

**Описание таблиц MS SQL**

Таблица "BRAND" предназначена для хранения информации о брендах. В неё входят следующие столбцы:

Таблица 1.1 – Столбцы таблицы **BRAND**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор бренда | INT |
| BRAND\_NAME | хранит название бренда | NVARCHAR(40) |

Таблица “COLOR” предназначена для хранения информации о цветах. В неё входят следующие столбцы:

Таблица 1.2 – Столбцы таблицы **COLOR**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор цвета | INT |
| COLOR\_NAME | хранит название цвета | NVARCHAR(20) |

Таблица "CAR" предназначена для хранения информации об автомобилях. В неё входят следующие столбцы:

Таблица 1.3 – Столбцы таблицы **CAR**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| REGISTRATION\_NUM | хранит уникальный номер регистрации автомобиля | INT |
| BRAND\_ID | хранит идентификатор бренда автомобиля | INT |
| MODEL | хранит модель автомобиля | VARCHAR2(20) |
| YEAR | хранит год выпуска автомобиля | DATE |
| COLOR\_ID | хранит идентификатор цвета автомобиля | INT |
| CAPACITY | хранит информации о количестве пассажиров | INT |

Таблица "**CAR**" содержит следующие ограничения:

* Ограничение первичного ключа (PRIMARY KEY) на столбец "REGISTRATION\_NUM".
* Ограничение внешнего ключа (FOREIGN KEY) FK\_CAR\_BRAND\_ID, ссылается на столбец "ID" таблицы "BRAND" и при удалении записи из таблицы "BRAND" устанавливает значение NULL в столбце "BRAND\_ID".
* Ограничение внешнего ключа (FOREIGN KEY) FK\_CAR\_COLOR\_ID, ссылается на столбец "ID" таблицы "COLOR" и при удалении записи из таблицы "COLOR" устанавливает значение NULL в столбце "COLOR\_ID".

Таблица "DRIVER" предназначена для хранения информации о водителях. В неё входят следующие столбцы

Таблица 1.4 – Столбцы таблицы **DRIVER**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор водителя | INT |
| NAME | хранит имя водителя | NVARCHAR2(40) |
| SURNAME | хранит фамилию водителя | NVARCHAR2(40) |
| LICENSE | хранит номер водительского удостоверения | INT |
| PHONE\_NUMBER | хранит номер телефона водителя | VARCHAR2(20) |
| EMAIL | хранит адрес электронной почты водителя |  |

Таблица "**DRIVER**" содержит следующее ограничение:

Ограничение первичного ключа (PRIMARY KEY) на столбец "ID".

Таблица "DRIVER\_CAR" предназначена для связи водителей с автомобилями. В неё входят следующие столбцы:

Таблица 1.5 – Столбцы таблицы **DRIVER\_CAR**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| DRIVER\_ID | хранит идентификатор водителя | INT |
| CAR\_ID | хранит идентификатор автомобиля | INT |
| DATE | хранит дату связи между водителем и автомобилем | DATE |

Таблица "**DRIVER\_CAR**" содержит следующие ограничения:

* Ограничение внешнего ключа (FOREIGN KEY) FK\_DRIVER\_CAR\_DRIVER\_ID, ссылается на столбец "ID" таблицы "DRIVER" и при удалении записи из таблицы "DRIVER" автоматически удаляет связанные записи из таблицы "DRIVER\_CAR" (ON DELETE CASCADE).
* Ограничение внешнего ключа (FOREIGN KEY) FK\_DRIVER\_CAR\_CAR\_ID, ссылается на столбец "REGISTRATION\_NUM" таблицы "CAR" и при удалении записи из таблицы "CAR" автоматически удаляет связанные записи из таблицы "DRIVER\_CAR" (ON DELETE CASCADE).

Таблица "CLIENT" предназначена для хранения информации о клиентах. В неё входят следующие столбцы:

Таблица 1.6 – Столбцы таблицы **CLIENT**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор клиента | INT |
| NAME | хранит имя клиента | NVARCHAR2(40) |
| SURNAME | хранит фамилию клиента | NVARCHAR2(40) |
| ADDRESS | хранит адрес клиента | NVARCHAR2(100) |
| PHONE\_NUMBER | хранит номер телефона клиента | VARCHAR2(20) |
| EMAIL | хранит адрес электронной почты клиента | NVARCHAR2(50) |

Таблица "**CLIENT**" содержит следующие ограничения:

* Ограничение первичного ключа (PRIMARY KEY) на столбец "ID".
* Ограничение проверки (CHECK) CK\_CLIENT\_CONTACT\_NOT\_NULL, которое гарантирует, что либо столбец "PHONE\_NUMBER", либо столбец "EMAIL" содержит непустое значение.

Таблица "ORDER" предназначена для хранения информации о заказах. В неё входят следующие столбцы:

Таблица 1.7 – Столбцы таблицы **ORDER**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор заказа | INT |
| CLIENT\_ID | хранит идентификатор клиента, связанного с заказом | INT |
| DRIVER\_ID | хранит идентификатор водителя, связанного с заказом | INT |
| INFO | хранит информацию о заказе | NVARCHAR2(300) |
| DATETIME | хранит дату и время заказа | DATE |
| COST | хранит стоимость заказа | FLOAT |
| STATUS | хранит идентификатор статуса заказа | INT |
| ARRIVAL | хранит место прибытия заказа | NVARCHAR2(100) |

Таблица "**ORDER**" содержит следующие ограничения:

* Ограничение внешнего ключа (FOREIGN KEY) FK\_ORDER\_CLIENT\_ID, ссылается на столбец "ID" таблицы "CLIENT" и при удалении записи из таблицы "CLIENT" устанавливает значение NULL в столбце "CLIENT\_ID".
* Ограничение внешнего ключа (FOREIGN KEY) FK\_ORDER\_DRIVER\_ID, ссылается на столбец "ID" таблицы "DRIVER" и при удалении записи из таблицы "DRIVER" устанавливает значение NULL в столбце "DRIVER\_ID".
* Ограничение первичного ключа (PRIMARY KEY) на столбец "ID".

Таблица "RIVIEW" предназначена для хранения информации о отзывах о заказах. В неё входят следующие столбцы:

Таблица 1.8 – Столбцы таблицы **RIVIEW**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ORDER\_ID | хранит идентификатор заказа, к которому относится отзыв | INT |
| TEXT | хранит текст отзыва | NVARCHAR2(500) |
| STARS | хранит оценку отзыва в виде количества звезд | INT |

Таблица "RIVIEW" содержит следующие ограничения:

* Ограничение внешнего ключа (FOREIGN KEY) FK\_REVIEW\_ORDER\_ID, ссылается на столбец "ID" таблицы "ORDER".
* Ограничение первичного ключа (PRIMARY KEY) на столбец "ORDER\_ID".
* Ограничение проверки (CHECK) CK\_REVIEW\_STARS\_RANGE, которое гарантирует, что значение в столбце "STARS" находится в диапазоне от 1 до 10

**Описание таблиц Oracle**

Таблица "BRAND" предназначена для хранения информации о брендах. В неё входят следующие столбцы:

Таблица 2.1 – Столбцы таблицы **BRAND**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор бренда | INT |
| BRAND\_NAME | хранит название бренда | NVARCHAR2(40) |

Значение BRAND\_NAME не может быть NULL (NOT NULL).

Таблица “COLOR” предназначена для хранения информации о цветах. В неё входят следующие столбцы:

Таблица 2.2 – Столбцы таблицы **COLOR**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор цвета | INT |
| COLOR\_NAME | хранит название цвета | NVARCHAR2(20) |

Таблица "CAR" предназначена для хранения информации об автомобилях. В неё входят следующие столбцы:

Таблица 2.3 – Столбцы таблицы **CAR**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| REGISTRATION\_NUM | хранит уникальный номер регистрации автомобиля | INT |
| BRAND\_ID | хранит идентификатор бренда автомобиля | INT |
| MODEL | хранит модель автомобиля | VARCHAR2(20) |
| YEAR | хранит год выпуска автомобиля | DATE |
| COLOR\_ID | хранит идентификатор цвета автомобиля | INT |
| CAPACITY | хранит информации о количестве пассажиров | INT |

Таблица "**CAR**" содержит следующие ограничения:

* Ограничение первичного ключа (PRIMARY KEY) на столбец "REGISTRATION\_NUM".
* Ограничение внешнего ключа (FOREIGN KEY) FK\_CAR\_BRAND\_ID, ссылается на столбец "ID" таблицы "BRAND" и при удалении записи из таблицы "BRAND" устанавливает значение NULL в столбце "BRAND\_ID".
* Ограничение внешнего ключа (FOREIGN KEY) FK\_CAR\_COLOR\_ID, ссылается на столбец "ID" таблицы "COLOR" и при удалении записи из таблицы "COLOR" устанавливает значение NULL в столбце "COLOR\_ID".

Таблица "DRIVER" предназначена для хранения информации о водителях. В неё входят следующие столбцы

Таблица 2.4 – Столбцы таблицы **DRIVER**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор водителя | INT |
| NAME | хранит имя водителя | NVARCHAR2(40) |
| SURNAME | хранит фамилию водителя | NVARCHAR2(40) |
| LICENSE | хранит номер водительского удостоверения | INT |
| PHONE\_NUMBER | хранит номер телефона водителя | VARCHAR2(20) |
| EMAIL | хранит адрес электронной почты водителя | NVARCHAR2(50) |

Таблица "**DRIVER**" содержит следующее ограничение:

Ограничение первичного ключа (PRIMARY KEY) на столбец "ID".

Таблица "DRIVER\_CAR" предназначена для связи водителей с автомобилями. В неё входят следующие столбцы:

Таблица 2.5 – Столбцы таблицы **DRIVER\_CAR**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| DRIVER\_ID | хранит идентификатор водителя | INT |
| CAR\_ID | хранит идентификатор автомобиля | INT |
| DATE | хранит дату связи между водителем и автомобилем | DATE |

Таблица "**DRIVER\_CAR**" содержит следующие ограничения:

* Ограничение внешнего ключа (FOREIGN KEY) FK\_DRIVER\_CAR\_DRIVER\_ID, ссылается на столбец "ID" таблицы "DRIVER" и при удалении записи из таблицы "DRIVER" автоматически удаляет связанные записи из таблицы "DRIVER\_CAR" (ON DELETE CASCADE).
* Ограничение внешнего ключа (FOREIGN KEY) FK\_DRIVER\_CAR\_CAR\_ID, ссылается на столбец "REGISTRATION\_NUM" таблицы "CAR" и при удалении записи из таблицы "CAR" автоматически удаляет связанные записи из таблицы "DRIVER\_CAR" (ON DELETE CASCADE).

Таблица "CLIENT" предназначена для хранения информации о клиентах. В неё входят следующие столбцы:

Таблица 2.6 – Столбцы таблицы **CLIENT**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор клиента | INT |
| NAME | хранит имя клиента | NVARCHAR2(40) |
| SURNAME | хранит фамилию клиента | NVARCHAR2(40) |
| ADDRESS | хранит адрес клиента | NVARCHAR2(100) |
| PHONE\_NUMBER | хранит номер телефона клиента | VARCHAR2(20) |
| EMAIL | хранит адрес электронной почты клиента | NVARCHAR2(50) |

Таблица "**CLIENT**" содержит следующие ограничения:

* Ограничение первичного ключа (PRIMARY KEY) на столбец "ID".
* Ограничение проверки (CHECK) CK\_CLIENT\_CONTACT\_NOT\_NULL, которое гарантирует, что либо столбец "PHONE\_NUMBER", либо столбец "EMAIL" содержит непустое значение.

Таблица "ORDER" предназначена для хранения информации о заказах. В неё входят следующие столбцы:

Таблица 2.7 – Столбцы таблицы **ORDER**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ID | хранит уникальный идентификатор заказа | INT |
| CLIENT\_ID | хранит идентификатор клиента, связанного с заказом | INT |
| DRIVER\_ID | хранит идентификатор водителя, связанного с заказом | INT |
| INFO | хранит информацию о заказе | NVARCHAR2(300) |
| DATETIME | хранит дату и время заказа | DATE |
| COST | хранит стоимость заказа | FLOAT |
| STATUS | хранит идентификатор статуса заказа | INT |
| ARRIVAL | хранит место прибытия заказа | NVARCHAR2(100) |

Таблица "**ORDER**" содержит следующие ограничения:

* Ограничение внешнего ключа (FOREIGN KEY) FK\_ORDER\_CLIENT\_ID, ссылается на столбец "ID" таблицы "CLIENT" и при удалении записи из таблицы "CLIENT" устанавливает значение NULL в столбце "CLIENT\_ID".
* Ограничение внешнего ключа (FOREIGN KEY) FK\_ORDER\_DRIVER\_ID, ссылается на столбец "ID" таблицы "DRIVER" и при удалении записи из таблицы "DRIVER" устанавливает значение NULL в столбце "DRIVER\_ID".
* Ограничение первичного ключа (PRIMARY KEY) на столбец "ID".

Таблица "RIVIEW" предназначена для хранения информации о отзывах о заказах. В неё входят следующие столбцы:

Таблица 2.8 – Столбцы таблицы **RIVIEW**

|  |  |  |
| --- | --- | --- |
| Наименование | Описание | Тип |
| ORDER\_ID | хранит идентификатор заказа, к которому относится отзыв | INT |
| TEXT | хранит текст отзыва | NVARCHAR2(500) |
| STARS | хранит оценку отзыва в виде количества звезд | INT |

Таблица "RIVIEW" содержит следующие ограничения:

* Ограничение внешнего ключа (FOREIGN KEY) FK\_REVIEW\_ORDER\_ID, ссылается на столбец "ID" таблицы "ORDER".
* Ограничение первичного ключа (PRIMARY KEY) на столбец "ORDER\_ID".
* Ограничение проверки (CHECK) CK\_REVIEW\_STARS\_RANGE, которое гарантирует, что значение в столбце "STARS" находится в диапазоне от 1 до 10

**Описание объектов MS SQL**

**Представление "VW\_CAR":**

* Это представление позволяет получить информацию о машинах, включая их регистрационный номер, марку, цвет, модель и год выпуска.
* Оно объединяет данные из таблицы "CAR" с таблицами "BRAND" и "COLOR" по соответствующим идентификаторам.
* Представление может использоваться для получения общей информации о доступных автомобилях в системе.

|  |
| --- |
| CREATE VIEW VW\_CAR AS  SELECT c.REGISTRATION\_NUM, b.BRAND\_NAME AS BRAND, col.COLOR\_NAME AS COLOR, c.MODEL, c.YEAR  FROM CAR c  INNER JOIN BRAND b ON c.BRAND\_ID = b.ID  INNER JOIN COLOR col ON c.COLOR\_ID = col.ID; |

**Представление "VW\_DRIVER\_CAR":**

* Это представление объединяет информацию о водителях и их автомобилях.
* Оно содержит данные о идентификаторе водителя, имени водителя, идентификаторе автомобиля и модели автомобиля.
* Представление связывает таблицы "DRIVER\_CAR", "DRIVER" и "CAR" для получения соответствующих данных.
* Это представление может быть полезным для отображения списка водителей и связанных с ними автомобилей.

|  |
| --- |
| CREATE OR ALTER VIEW VW\_DRIVER\_CAR AS  SELECT dc.DRIVER\_ID, d.NAME AS DRIVER\_NAME, dc.CAR\_ID, c.MODEL  FROM DRIVER\_CAR dc  INNER JOIN DRIVER d ON dc.DRIVER\_ID = d.ID  INNER JOIN CAR c ON dc.CAR\_ID = c.REGISTRATION\_NUM; |

# Представление "VW\_ORDER":

* Это представление объединяет информацию о заказах и связанных с ними данными.
* Включает идентификатор заказа, статус, дату и время, стоимость, данные о клиенте (имя, электронная почта), данные о водителе (имя, номер телефона) и данные об автомобиле (регистрационный номер, марка, модель, цвет).
* Данные объединяются из таблиц "ORDER", "CLIENT", "DRIVER", "DRIVER\_CAR" и представления "VW\_CAR".

Представление предоставляет полную информацию о заказах и связанных с ними данными, что может быть полезно для анализа и управления заказами

|  |
| --- |
| CREATE OR ALTER VIEW VW\_ORDER AS  SELECT o.ID, o.DATETIME, o.COST,  c.NAME AS CLIENT\_NAME, c.EMAIL AS CLIENT\_EMAIL,  dr.NAME AS DRIVER\_NAME, dr.PHONE\_NUMBER AS DRIVER\_NUMBER,  car.REGISTRATION\_NUM AS REG\_NUM, car.BRAND, car.MODEL, car.COLOR  FROM [ORDER] o  INNER JOIN CLIENT c ON o.CLIENT\_ID = c.ID  INNER JOIN DRIVER dr ON o.DRIVER\_ID = dr.ID  INNER JOIN DRIVER\_CAR dr\_car ON dr\_car.DRIVER\_ID = dr.ID AND CONVERT(DATE, dr\_car."DATE") = CONVERT(DATE, o.DATETIME)  INNER JOIN VW\_CAR car ON car.REGISTRATION\_NUM = dr\_car.CAR\_ID; |

# Представление "VW\_REVIEW":

* Это представление объединяет информацию об отзывах о заказах.
* Включает идентификатор заказа, текст отзыва, оценку (звезды) и имя клиента.
* Данные объединяются из таблиц "REVIEW", "ORDER" и "CLIENT".
* Представление позволяет получить информацию о отзывах и связанных с ними заказах, что может быть полезно для анализа удовлетворенности клиентов.

|  |
| --- |
| CREATE OR ALTER VIEW VW\_REVIEW AS  SELECT r.ORDER\_ID AS ID, r.TEXT, r.STARS, c.NAME  FROM REVIEW r  INNER JOIN [ORDER] o ON r.ORDER\_ID = o.ID  INNER JOIN CLIENT c ON c.ID = o.CLIENT\_ID; |

**Представление " VW\_ACTIVE\_ORDERS":**

* Это представление получает данные из представления VW\_ORDER, но отфильтровывает только те заказы, которые активны.

|  |
| --- |
| CREATE OR ALTER VIEW VW\_ACTIVE\_ORDERS AS  SELECT \*  FROM VW\_ORDER |

# Процедура "UPDATE\_ORDER\_STATUS":

* Эта процедура позволяет обновить статус заказа на новое значение.
* Принимает входные параметры: идентификатор заказа (p\_order\_id) и новое значение статуса (p\_new\_status).
* Обновляет запись в таблице "ORDER" с указанным идентификатором заказа.
* После обновления происходит фиксация изменений.

|  |
| --- |
| CREATE OR ALTER PROCEDURE UPDATE\_ORDER\_STATUS  @p\_order\_id INT,  @p\_new\_status INT  AS  BEGIN  UPDATE [ORDER]  SET STATUS = @p\_new\_status  WHERE ID = @p\_order\_id;  COMMIT;  END; |

**Процедура "ADD\_DRIVER":**

* Эта процедура позволяет добавить нового водителя в систему.
* Принимает входные параметры: имя водителя (p\_name), фамилия водителя (p\_surname), номер лицензии (p\_license), номер телефона (p\_phone\_number) и электронная почта (p\_email).
* Создает новую запись в таблице "DRIVER" с указанными данными.
* После добавления водителя происходит фиксация изменений.

|  |
| --- |
| CREATE OR ALTER PROCEDURE ADD\_REVIEW  @p\_order\_id INT,  @p\_text NVARCHAR(1000),  @p\_stars INT  AS  BEGIN  INSERT INTO REVIEW (ORDER\_ID, TEXT, STARS)  VALUES (@p\_order\_id, @p\_text, @p\_stars);  COMMIT;  END;  GO |

# Процедура "GET\_CARS\_BY\_COLOR":

* Эта процедура позволяет получить информацию о машинах по их цвету.
* Принимает входной параметр: идентификатор цвета (p\_color\_id).
* Выполняет выборку регистрационного номера, названия марки, модели и года выпуска машин из таблицы CAR.
* Происходит объединение таблицы CAR с таблицей BRAND по идентификатору марки (BRAND\_ID), чтобы получить соответствующее название марки.
* Отбираются только те машины, у которых идентификатор цвета (COLOR\_ID) соответствует указанному значению p\_color\_id.

Процедура "ADD\_REVIEW":

* Эта процедура позволяет добавить отзыв к заказу.
* Принимает входные параметры: идентификатор заказа (p\_order\_id), текст отзыва (p\_text) и количество звезд (p\_stars).
* Вставляет новую запись в таблицу REVIEW с указанными данными: идентификатор заказа, текст отзыва и количество звезд.
* После вставки происходит фиксация изменений с помощью команды COMMIT, чтобы сделанные изменения стали постоянными.

|  |
| --- |
| CREATE OR ALTER PROCEDURE GET\_CARS\_BY\_COLOR  @p\_color\_id INT  AS  BEGIN  SELECT c.REGISTRATION\_NUM, b.BRAND\_NAME, c.MODEL, c.YEAR  FROM CAR c  INNER JOIN BRAND b ON c.BRAND\_ID = b.ID  WHERE c.COLOR\_ID = @p\_color\_id;  END; |

# Индекс "IDX\_ORDER\_CLIENT\_ID"

Индекс "IDX\_ORDER\_CLIENT\_ID" на таблице "ORDER" (CLIENT\_ID):

* Этот индекс создан на столбце CLIENT\_ID в таблице "ORDER".
* Индекс помогает ускорить выполнение запросов, которые выполняются на основе столбца CLIENT\_ID в таблице "ORDER".
* Индекс позволяет быстро находить записи в таблице "ORDER", относящиеся к определенному клиенту по его идентификатору.

|  |
| --- |
| CREATE INDEX IDX\_ORDER\_CLIENT\_ID ON "ORDER" (CLIENT\_ID); |

# Индекс "IDX\_DRIVER\_LICENSE

Уникальный индекс "IDX\_DRIVER\_LICENSE" на таблице DRIVER (LICENSE):

* Этот индекс создан на столбце LICENSE в таблице DRIVER.
* Уникальный индекс гарантирует, что значения в столбце LICENSE являются уникальными и не повторяются в таблице.
* Индекс позволяет эффективно проверять уникальность значений в столбце LICENSE и обеспечивает быстрый доступ к данным по лицензии водителя.

|  |
| --- |
| CREATE UNIQUE INDEX IDX\_DRIVER\_LICENSE ON DRIVER (LICENSE); |

# Индекс "IDX\_CAR\_BRAND\_ID"

Индекс "IDX\_CAR\_BRAND\_ID" на таблице CAR (BRAND\_ID):

* Этот индекс создан на столбце BRAND\_ID в таблице CAR.
* Индекс улучшает производительность запросов, которые выполняются на основе столбца BRAND\_ID в таблице CAR.
* Индекс позволяет быстро находить записи в таблице CAR по идентификатору марки автомобиля.

|  |
| --- |
| CREATE INDEX IDX\_CAR\_BRAND\_ID ON CAR (BRAND\_ID); |

# Индекс "IDX\_CLIENT\_NAME\_SURNAME"

Индекс "IDX\_CLIENT\_NAME\_SURNAME" на таблице CLIENT (NAME, SURNAME):

* Этот индекс создан на столбцах NAME и SURNAME в таблице CLIENT.
* Индекс создан с использованием нескольких столбцов, что позволяет эффективно выполнять запросы, которые фильтруются по имени (NAME) и фамилии (SURNAME) клиента одновременно.
* Индекс обеспечивает быстрый доступ к данным в таблице CLIENT на основе комбинации значений столбцов NAME и SURNAME.

|  |
| --- |
| CREATE INDEX IDX\_CLIENT\_NAME\_SURNAME ON CLIENT (NAME, SURNAME); |

**Функция GET\_ALL\_BRANDS**

Получение всех марок автомобилей

|  |
| --- |
| CREATE OR ALTER FUNCTION GET\_ALL\_BRANDS()  RETURNS TABLE  AS  RETURN (  SELECT BRAND\_NAME  FROM BRAND  ); |

**Функция GET\_ALL\_BRANDS**

Получение всех цветов автомобилей

|  |
| --- |
| CREATE OR ALTER FUNCTION GET\_ALL\_COLORS()  RETURNS TABLE  AS  RETURN (  SELECT COLOR\_NAME  FROM COLOR  ); |

**Функция GET\_CAR\_BY\_REGISTRATION\_NUM**

Получение информации о конкретном автомобиле по его регистрационному номеру

|  |
| --- |
| CREATE OR ALTER FUNCTION GET\_CAR\_BY\_REGISTRATION\_NUM  (  @p\_registration\_num INT  )  RETURNS TABLE  AS  RETURN (  SELECT c.REGISTRATION\_NUM, b.BRAND\_NAME, c.MODEL, c.YEAR, cl.COLOR\_NAME, c.CAPACITY  FROM CAR c  INNER JOIN BRAND b ON c.BRAND\_ID = b.ID  INNER JOIN COLOR cl ON c.COLOR\_ID = cl.ID  WHERE c.REGISTRATION\_NUM = @p\_registration\_num  ); |

**Описание Объектов Oracle**

# Представление "VW\_CAR":

* Это представление позволяет получить информацию о машинах, включая их регистрационный номер, марку, цвет, модель и год выпуска.
* Оно объединяет данные из таблицы "CAR" с таблицами "BRAND" и "COLOR" по соответствующим идентификаторам.
* Представление может использоваться для получения общей информации о доступных автомобилях в системе.

|  |
| --- |
| CREATE VIEW VW\_CAR AS  SELECT c.REGISTRATION\_NUM, b.BRAND\_NAME AS BRAND, col.COLOR\_NAME AS COLOR, c.MODEL, c.YEAR  FROM CAR c  INNER JOIN BRAND b ON c.BRAND\_ID = b.ID  INNER JOIN COLOR col ON c.COLOR\_ID = col.ID; |

# Представление "VW\_DRIVER\_CAR":

* Это представление объединяет информацию о водителях и их автомобилях.
* Оно содержит данные о идентификаторе водителя, имени водителя, идентификаторе автомобиля и модели автомобиля.
* Представление связывает таблицы "DRIVER\_CAR", "DRIVER" и "CAR" для получения соответствующих данных.
* Это представление может быть полезным для отображения списка водителей и связанных с ними автомобилей.

|  |
| --- |
| CREATE OR ALTER VIEW VW\_DRIVER\_CAR AS  SELECT dc.DRIVER\_ID, d.NAME AS DRIVER\_NAME, dc.CAR\_ID, c.MODEL  FROM DRIVER\_CAR dc  INNER JOIN DRIVER d ON dc.DRIVER\_ID = d.ID  INNER JOIN CAR c ON dc.CAR\_ID = c.REGISTRATION\_NUM; |

# Представление "VW\_ORDER":

* Это представление объединяет информацию о заказах и связанных с ними данными.
* Включает идентификатор заказа, статус, дату и время, стоимость, данные о клиенте (имя, электронная почта), данные о водителе (имя, номер телефона) и данные об автомобиле (регистрационный номер, марка, модель, цвет).
* Данные объединяются из таблиц "ORDER", "CLIENT", "DRIVER", "DRIVER\_CAR" и представления "VW\_CAR".
* Представление предоставляет полную информацию о заказах и связанных с ними данными, что может быть полезно для анализа и управления заказами.

|  |
| --- |
| CREATE OR ALTER VIEW VW\_ORDER AS  SELECT o.ID, o.DATETIME, o.COST,  c.NAME AS CLIENT\_NAME, c.EMAIL AS CLIENT\_EMAIL,  dr.NAME AS DRIVER\_NAME, dr.PHONE\_NUMBER AS DRIVER\_NUMBER,  car.REGISTRATION\_NUM AS REG\_NUM, car.BRAND, car.MODEL, car.COLOR  FROM [ORDER] o  INNER JOIN CLIENT c ON o.CLIENT\_ID = c.ID  INNER JOIN DRIVER dr ON o.DRIVER\_ID = dr.ID  INNER JOIN DRIVER\_CAR dr\_car ON dr\_car.DRIVER\_ID = dr.ID AND CONVERT(DATE, dr\_car."DATE") = CONVERT(DATE, o.DATETIME)  INNER JOIN VW\_CAR car ON car.REGISTRATION\_NUM = dr\_car.CAR\_ID; |

# Представление "VW\_REVIEW":

* Это представление объединяет информацию об отзывах о заказах.
* Включает идентификатор заказа, текст отзыва, оценку (звезды) и имя клиента.
* Данные объединяются из таблиц "REVIEW", "ORDER" и "CLIENT".
* Представление позволяет получить информацию о отзывах и связанных с ними заказах, что может быть полезно для анализа удовлетворенности клиентов.

|  |
| --- |
| CREATE OR REPLACE VIEW VW\_REVIEW AS  SELECT r.ORDER\_ID AS ID, r.TEXT, r.STARS, c.NAME  FROM REVIEW r  INNER JOIN "ORDER" o ON r.ORDER\_ID = o.ID  INNER JOIN CLIENT c ON c.ID =o.CLIENT\_ID; |

# Представление " VW\_ACTIVE\_ORDERS":

* Это представление получает данные из представления VW\_ORDER, но отфильтровывает только те заказы, которые активны.

|  |
| --- |
| CREATE OR REPLACE VIEW VW\_ACTIVE\_ORDERS AS SELECT \* FROM VW\_ORDER; |

# Процедура "ADD\_DRIVER":

* Эта процедура позволяет добавить нового водителя в систему.
* Принимает входные параметры: имя водителя (p\_name), фамилия водителя (p\_surname), номер лицензии (p\_license), номер телефона (p\_phone\_number) и электронная почта (p\_email).
* Создает новую запись в таблице "DRIVER" с указанными данными.
* После добавления водителя происходит фиксация изменений.

|  |
| --- |
| CREATE OR REPLACE PROCEDURE ADD\_DRIVER(  p\_name IN NVARCHAR2,  p\_surname IN NVARCHAR2,  p\_license IN INT,  p\_phone\_number IN VARCHAR2,  p\_email IN NVARCHAR2  )  AS  BEGIN  INSERT INTO DRIVER (NAME, SURNAME, LICENSE, PHONE\_NUMBER, EMAIL)  VALUES (p\_name, p\_surname, p\_license, p\_phone\_number, p\_email);  COMMIT;  END; |

# Процедура "GET\_CARS\_BY\_COLOR":

* Эта процедура позволяет получить информацию о машинах по их цвету.
* Принимает входной параметр: идентификатор цвета (p\_color\_id).
* Выполняет выборку регистрационного номера, названия марки, модели и года выпуска машин из таблицы CAR.
* Происходит объединение таблицы CAR с таблицей BRAND по идентификатору марки (BRAND\_ID), чтобы получить соответствующее название марки.
* Отбираются только те машины, у которых идентификатор цвета (COLOR\_ID) соответствует указанному значению p\_color\_id.

Процедура "ADD\_REVIEW":

* Эта процедура позволяет добавить отзыв к заказу.
* Принимает входные параметры: идентификатор заказа (p\_order\_id), текст отзыва (p\_text) и количество звезд (p\_stars).
* Вставляет новую запись в таблицу REVIEW с указанными данными: идентификатор заказа, текст отзыва и количество звезд.
* После вставки происходит фиксация изменений с помощью команды COMMIT, чтобы сделанные изменения стали постоянными.

|  |
| --- |
| CREATE OR REPLACE PROCEDURE GET\_CARS\_BY\_COLOR(  p\_color\_id IN INT  )  AS  BEGIN  SELECT c.REGISTRATION\_NUM, b.BRAND\_NAME, c.MODEL, c.YEAR  FROM CAR c  INNER JOIN BRAND b ON c.BRAND\_ID = b.ID  WHERE c.COLOR\_ID = p\_color\_id;  END; |

# Индекс "IDX\_ORDER\_CLIENT\_ID"

Индекс "IDX\_ORDER\_CLIENT\_ID" на таблице "ORDER" (CLIENT\_ID):

* Этот индекс создан на столбце CLIENT\_ID в таблице "ORDER".
* Индекс помогает ускорить выполнение запросов, которые выполняются на основе столбца CLIENT\_ID в таблице "ORDER".
* Индекс позволяет быстро находить записи в таблице "ORDER", относящиеся к определенному клиенту по его идентификатору.

|  |
| --- |
| CREATE INDEX IDX\_ORDER\_CLIENT\_ID ON "ORDER" (CLIENT\_ID); |

# Индекс "IDX\_DRIVER\_LICENSE

Уникальный индекс "IDX\_DRIVER\_LICENSE" на таблице DRIVER (LICENSE):

* Этот индекс создан на столбце LICENSE в таблице DRIVER.
* Уникальный индекс гарантирует, что значения в столбце LICENSE являются уникальными и не повторяются в таблице.
* Индекс позволяет эффективно проверять уникальность значений в столбце LICENSE и обеспечивает быстрый доступ к данным по лицензии водителя.

|  |
| --- |
| CREATE UNIQUE INDEX IDX\_DRIVER\_LICENSE ON DRIVER (LICENSE); |

# Индекс "IDX\_CAR\_BRAND\_ID"

Индекс "IDX\_CAR\_BRAND\_ID" на таблице CAR (BRAND\_ID):

* Этот индекс создан на столбце BRAND\_ID в таблице CAR.
* Индекс улучшает производительность запросов, которые выполняются на основе столбца BRAND\_ID в таблице CAR.
* Индекс позволяет быстро находить записи в таблице CAR по идентификатору марки автомобиля.

|  |
| --- |
| CREATE INDEX IDX\_CAR\_BRAND\_ID ON CAR (BRAND\_ID); |

# Индекс "IDX\_CLIENT\_NAME\_SURNAME"

Индекс "IDX\_CLIENT\_NAME\_SURNAME" на таблице CLIENT (NAME, SURNAME):

* Этот индекс создан на столбцах NAME и SURNAME в таблице CLIENT.
* Индекс создан с использованием нескольких столбцов, что позволяет эффективно выполнять запросы, которые фильтруются по имени (NAME) и фамилии (SURNAME) клиента одновременно.
* Индекс обеспечивает быстрый доступ к данным в таблице CLIENT на основе комбинации значений столбцов NAME и SURNAME.

|  |
| --- |
| CREATE INDEX IDX\_CLIENT\_NAME\_SURNAME ON CLIENT (NAME, SURNAME); |

**Функция GET\_ALL\_BRANDS**

Получение всех марок автомобилей

|  |
| --- |
| CREATE OR REPLACE FUNCTION GET\_ALL\_BRANDS  RETURN SYS\_REFCURSOR  AS  brands\_cursor SYS\_REFCURSOR;  BEGIN  OPEN brands\_cursor FOR  SELECT BRAND\_NAME  FROM BRAND;  RETURN brands\_cursor;  END; |

**Функция GET\_ALL\_COLORS Получение всех цветов автомобилей**

Получение всех цветов автомобилей

|  |
| --- |
| CREATE OR REPLACE FUNCTION GET\_ALL\_COLORS  RETURN SYS\_REFCURSOR  AS  colors\_cursor SYS\_REFCURSOR;  BEGIN  OPEN colors\_cursor FOR  SELECT COLOR\_NAME  FROM COLOR;  RETURN colors\_cursor;  END; |

**Функция GET\_CAR\_BY\_REGISTRATION\_NUM**

Получение информации о конкретном автомобиле по его регистрационному номеру

|  |
| --- |
| CREATE OR REPLACE FUNCTION GET\_CAR\_BY\_REGISTRATION\_NUM  (  p\_registration\_num IN INT  )  RETURN SYS\_REFCURSOR  AS  car\_cursor SYS\_REFCURSOR;  BEGIN  OPEN car\_cursor FOR  SELECT c.REGISTRATION\_NUM, b.BRAND\_NAME, c.MODEL, c.YEAR, cl.COLOR\_NAME, c.CAPACITY  FROM CAR c  INNER JOIN BRAND b ON c.BRAND\_ID = b.ID  INNER JOIN COLOR cl ON c.COLOR\_ID = cl.ID  WHERE c.REGISTRATION\_NUM = p\_registration\_num;  RETURN car\_cursor;  END; |