**Лабораторная работа № 9. Загрузка учебной базы, запросы.**

*Отдельные выдержки из предыдущих работ:*

|  |  |
| --- | --- |
| * вести файл протокола spool. txt   SQL> set echo on  SQL> spool d:\spool. Txt | * завершить протокол лаб. работы   SQL> spool off |
| * создать пользователя   SQL> create user user1 identified by user1  default tablespace system  quota unlimited on system;   * изменить пароль   SQL> alter user user1 identified by test;   * разблокировать   SQL> alter user user1 account unlock; | - просмотреть пользователей  SQL>select username from sys.dba\_users;  -предоставить роль DBA  SQL> grant dba to user1;  -уничтожить пользователя  SQL> drop user user1 cascade;  -показать пользователя  SQL>show user |
| - просмотреть названия таблиц в схеме USER1  SQL>select table\_name from sys.all\_tables where owner ='USER1'; | - просмотреть из system структуру таблицы regions схемы HR  SQL>desc HR.regions  - просмотреть из system содержимое таблицы regions схемы HR  SQL>select \* from HR.regions |
| * создать таблицу   SQL> CREATE TABLE dolj  (kod NUMBER(10) NOT NULL PRIMARY KEY, naimen VARCHAR2(50)); | - заполнить таблицу  SQL> INSERT INTO dolj VALUES (1, ‘Программист’);  - выбрать запись  SQL> SELECT \* FROM dolj WHERE kod=1; |
| SQL> select r.region\_name, c.country\_name from regions r, countries c where r.region\_id=c.region\_id;  REGION\_NAME COUNTRY\_NAME  ------------------------- ----------------------------------------  Americas Argentina  Asia Australia  Europe Belgium  Americas Brazil  Americas Canada  Europe Switzerland  Asia China  Europe Germany  Europe Denmark  Middle East and Africa Egypt  Europe France | SQL> select r.region\_name, c.country\_name from regions r, countries c where r.region\_id=c.region\_id order by r.region\_id;  REGION\_NAME COUNTRY\_NAME  ------------------------- ----------------------------------------  Europe Netherlands  Europe France  Europe United Kingdom  Europe Denmark  Europe Belgium  Europe Switzerland  Europe Italy  Europe Germany  Americas United States of America  Americas Canada  Americas Mexico |

|  |  |
| --- | --- |
| Логическая модель | Физическая модель |
| Сущность | Таблица |
| Атрибут | Столбец |
| Логический тип  (текст, число, дата, blob) | Физический тип (корректный тип, зависящий от выбранной СУБД) |
| Первичный ключ | Первичный ключ, индекс PK |
| Внешний ключ | Внешний ключ, индекс FK |
| Альтернативный ключ | AK-индекс, уникальный,  непервичный индекс |
| Правило бизнес-логики | Триггер или сохраненная  процедура |
| Взаимосвязи | Взаимосвязи, определяемые использованием FK-атрибутов |

|  |  |  |
| --- | --- | --- |
| HR_ER_DIAGRAM_01_CORR | | |
| CREATE TABLE regions  ( region\_id NUMBER  CONSTRAINT region\_id\_nn NOT NULL  , region\_name VARCHAR2(25)  );  CREATE UNIQUE INDEX reg\_id\_pk  ON regions (region\_id);  ALTER TABLE regions  ADD ( CONSTRAINT reg\_id\_pk  PRIMARY KEY (region\_id)  ) ; | CREATE TABLE countries  ( country\_id CHAR(2)  CONSTRAINT country\_id\_nn NOT NULL  , country\_name VARCHAR2(40)  , region\_id NUMBER  , CONSTRAINT country\_c\_id\_pk  PRIMARY KEY (country\_id)  )  ORGANIZATION INDEX;  ALTER TABLE countries  ADD ( CONSTRAINT countr\_reg\_fk  FOREIGN KEY (region\_id)  REFERENCES regions(region\_id)  ) ; | CREATE TABLE locations  ( location\_id NUMBER(4)  , street\_address VARCHAR2(40)  , postal\_code VARCHAR2(12)  , city VARCHAR2(30)  CONSTRAINT loc\_city\_nn NOT NULL  , state\_province VARCHAR2(25)  , country\_id CHAR(2)  ) ;  CREATE UNIQUE INDEX loc\_id\_pk  ON locations (location\_id) ;  ALTER TABLE locations  ADD ( CONSTRAINT loc\_id\_pk  PRIMARY KEY (location\_id)  , CONSTRAINT loc\_c\_id\_fk  FOREIGN KEY (country\_id)  REFERENCES countries(country\_id) ) ; |

**Задание №1. ЗАГРУЗИТЬ УЧЕБНУЮ БАЗУ**

SQL> set echo on

SQL> spool d:\spool. txt

SQL> @ d:\demobld. txt

-- Copyright (c) Oracle Corporation 1988, 2000. All Rights Reserved.

-- NAME

-- demobld.sql

-- DESCRIPTION

-- This script creates the SQL\*Plus demonstration tables in the

-- current schema. It should be STARTed by each user wishing to

-- access the tables. To remove the tables use the demodrop.sql

-- script.

-- USAGE

-- From within SQL\*Plus, enter:

-- START demobld.sql

SET TERMOUT ON

PROMPT Building demonstration TABLES. Please wait.

SET TERMOUT OFF

DROP TABLE EMP;

DROP TABLE DEPT;

DROP TABLE BONUS;

DROP TABLE SALGRADE;

DROP TABLE DUMMY;

CREATE TABLE EMP

(EMPNO NUMBER(4) NOT NULL,

ENAME VARCHAR2(10),

JOB VARCHAR2(9),

MGR NUMBER(4),

HIREDATE DATE,

SAL NUMBER(7, 2),

COMM NUMBER(7, 2),

DEPTNO NUMBER(2));

INSERT INTO EMP VALUES

(7369, 'SMITH', 'CLERK', 7902,

TO\_DATE('17-12-1980', 'DD-MM-YYYY'), 800, NULL, 20);

INSERT INTO EMP VALUES

(7499, 'ALLEN', 'SALESMAN', 7698,

TO\_DATE('20-02-1981', 'DD-MM-YYYY'), 1600, 300, 30);

INSERT INTO EMP VALUES

(7521, 'WARD', 'SALESMAN', 7698,

TO\_DATE('22-02-1981', 'DD-MM-YYYY'), 1250, 500, 30);

INSERT INTO EMP VALUES

(7566, 'JONES', 'MANAGER', 7839,

TO\_DATE('2-4-1981', 'DD-MM-YYYY'), 2975, NULL, 20);

INSERT INTO EMP VALUES

(7654, 'MARTIN', 'SALESMAN', 7698,

TO\_DATE('28-9-1981', 'DD-MM-YYYY'), 1250, 1400, 30);

INSERT INTO EMP VALUES

(7698, 'BLAKE', 'MANAGER', 7839,

TO\_DATE('1-5-1981', 'DD-MM-YYYY'), 2850, NULL, 30);

INSERT INTO EMP VALUES

(7782, 'CLARK', 'MANAGER', 7839,

TO\_DATE('9-6-1981', 'DD-MM-YYYY'), 2450, NULL, 10);

INSERT INTO EMP VALUES

(7788, 'SCOTT', 'ANALYST', 7566,

TO\_DATE('09-12-1982', 'DD-MM-YYYY'), 3000, NULL, 20);

INSERT INTO EMP VALUES

(7839, 'KING', 'PRESIDENT', NULL,

TO\_DATE('17-11-1981', 'DD-MM-YYYY'), 5000, NULL, 10);

INSERT INTO EMP VALUES

(7844, 'TURNER', 'SALESMAN', 7698,

TO\_DATE('8-9-1981', 'DD-MM-YYYY'), 1500, 0, 30);

INSERT INTO EMP VALUES

(7876, 'ADAMS', 'CLERK', 7788,

TO\_DATE('12-1-1983', 'DD-MM-YYYY'), 1100, NULL, 20);

INSERT INTO EMP VALUES

(7900, 'JAMES', 'CLERK', 7698,

TO\_DATE('3-12-1981', 'DD-MM-YYYY'), 950, NULL, 30);

INSERT INTO EMP VALUES

(7902, 'FORD', 'ANALYST', 7566,

TO\_DATE('3-12-1981', 'DD-MM-YYYY'), 3000, NULL, 20);

INSERT INTO EMP VALUES

(7934, 'MILLER', 'CLERK', 7782,

TO\_DATE('23-1-1982', 'DD-MM-YYYY'), 1300, NULL, 10);

CREATE TABLE DEPT

(DEPTNO NUMBER(2),

DNAME VARCHAR2(14),

LOC VARCHAR2(13) );

INSERT INTO DEPT VALUES (10, 'ACCOUNTING', 'NEW YORK');

INSERT INTO DEPT VALUES (20, 'RESEARCH', 'DALLAS');

INSERT INTO DEPT VALUES (30, 'SALES', 'CHICAGO');

INSERT INTO DEPT VALUES (40, 'OPERATIONS', 'BOSTON');

CREATE TABLE BONUS

(ENAME VARCHAR2(10),

JOB VARCHAR2(9),

SAL NUMBER,

COMM NUMBER);

CREATE TABLE SALGRADE

(GRADE NUMBER,

LOSAL NUMBER,

HISAL NUMBER);

INSERT INTO SALGRADE VALUES (1, 700, 1200);

INSERT INTO SALGRADE VALUES (2, 1201, 1400);

INSERT INTO SALGRADE VALUES (3, 1401, 2000);

INSERT INTO SALGRADE VALUES (4, 2001, 3000);

INSERT INTO SALGRADE VALUES (5, 3001, 9999);

CREATE TABLE DUMMY

(DUMMY NUMBER);

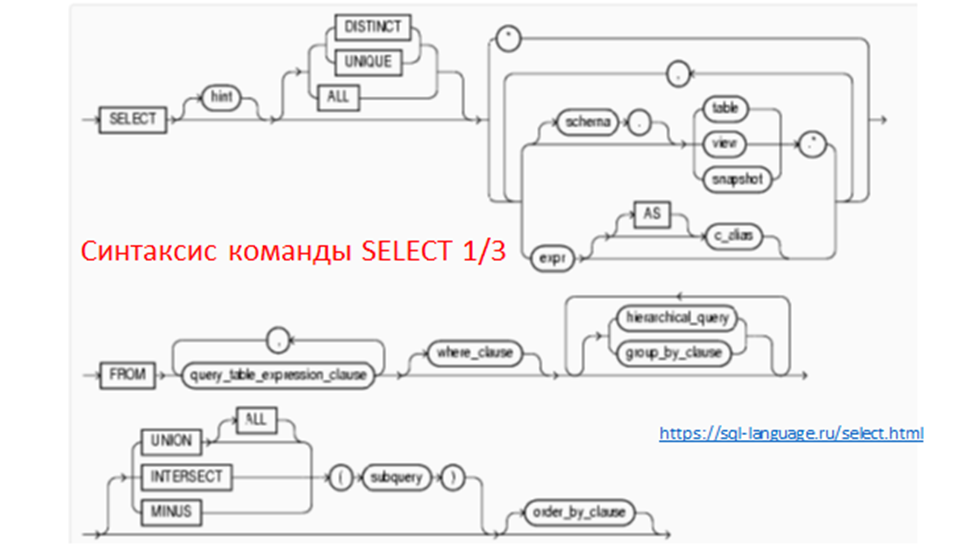
INSERT INTO DUMMY VALUES (0);

COMMIT;

SET TERMOUT ON

PROMPT DEMONSRTRATION TABLE BUILD IS COMPLETE

**Задание №2. ОСУЩЕСТВИТЬ МАНИПУЛЯЦИИ С ПРОСТЕЙШИМИ ЗАПРОСАМИ**

****

**Задание №3. РАЗРАБОТАТЬ ER Диаграмму (предложить связи).**