1. **Transaction Control**

create table Project (id number primary key,

pname varchar2(50),

cost number);

insert into project values (1, 'jupiter', 2000);

insert into project values (2, 'saturn', 1000);

insert into project values (3, 'mercury', 15000);

commit;

Write out the results of these statements:

1. Example: rollback

| **Time** | **session** | **output** |
| --- | --- | --- |
| t0 | COMMIT; |  |
| t1 | SET TRANSACTION NAME 'cost\_update'; |  |
| t2 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |
| t3 | UPDATE project  SET cost = 8000  WHERE id = 1; |  |
| t4 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |
| t5 | SELECT \* FROM project; |  |
| t6 | ROLLBACK; |  |
| t7 | SELECT \* FROM project; |  |
| t8 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |

1. Example: commit

| **Time** | **session** | **output** |
| --- | --- | --- |
| t1 | COMMIT; |  |
| t2 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |
| t3 | UPDATE project  SET cost = 6000  WHERE id = 2; |  |
| t4 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |
| t5 | Insert into project values (4, 'neptune', 19000); |  |
| t6 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |
| t7 | COMMIT; |  |
| t8 | SELECT \* FROM project; |  |
| t9 | SELECT XID, name, STATUS FROM V$TRANSACTION; |  |

1. Example: savepoint

| **Time** | **session** | **output** |
| --- | --- | --- |
| t0 | COMMIT; |  |
| t1 | Select \* from project; |  |
| t2 | Update project set cost=400000  where pname='jupiter'; |  |
| t3 | SAVEPOINT after\_jupiter\_cost; |  |
| t4 | Update project set cost=130  where pname='jupiter'; |  |
| t5 | SAVEPOINT after\_mercury\_cost; |  |
| t6 | ROLLBACK TO SAVEPOINT after\_jupiter\_cost; |  |
| t7 | Select \* from project; |  |
| t8 | Update project set cost=170  where pname='mercury'; |  |
| t9 | ROLLBACK; |  |
| t10 | Select \* from project; |  |

1. Example: DDL

| **Time** | **session** | **output** |
| --- | --- | --- |
| t0 | COMMIT; |  |
| t1 | Select \* from project; |  |
| t2 | SET TRANSACTION NAME 'cost\_update2'; |  |
| t3 | Update project set cost=12300  where pname='jupiter'; |  |
| t4 | Select \* from project; |  |
| t5 | --DDL statement  Create table test (id number); |  |
| t6 | Insert into test values (26); |  |
| t7 | Rollback; |  |
| t8 | Select \* from project;  Select \* from test; |  |

From SQLDeveloper:

You can right click on a connection and chose **'Open SQL Worksheet'** it will create another window for the existing session. (Use Alt + F10and select the connection from the list).

If you need to create another **independent** session you can use Ctrl + Shift + N for an ongoing session.

Open two session (Ctrl + Shift + N) and do the following things.

1. Compare data at time t3 and t5

| **Time** | **Session1** | **Output1** | **Session2** | **Output2** |
| --- | --- | --- | --- | --- |
| t0 | SET TRANSACTION NAME 'cost\_update3'; |  |  |  |
| t1 | Select \* from project; |  |  |  |
| t2 | Update project set cost=467  where pname='jupiter'; |  |  |  |
| t3 | Select \* from project; |  | Select \* from project; |  |
| t4 | Rollback; |  |  |  |
| t5 | Select \* from project; |  | Select \* from project; |  |

| **Time** | **Session1** | **Output1** | **Session2** | **Output2** |
| --- | --- | --- | --- | --- |
| t1 | SET TRANSACTION NAME 'cost\_update5'; |  |  |  |
| t2 | Update project set cost=1900  where pname='jupiter'; |  |  |  |
| t3 | Select \* from project; |  | Select \* from project; |  |
| t4 | Commit; |  |  |  |
| t5 | Select \* from project; |  | Select \* from project; |  |

1. Compare data at time t3, t5 and t8

| **Time** | **Session1** | **Output1** | **Session2** | **Output2** |
| --- | --- | --- | --- | --- |
| t0 | SET TRANSACTION NAME 'cost\_update6'; |  | SET TRANSACTION NAME 'cost\_update7'; |  |
| t1 | Select \* from project; |  | Select \* from project; |  |
| t2 | Update project set cost=3456  where pname='mercury'; |  |  |  |
| t3 | Select \* from project; |  | Select \* from project; |  |
| t4 |  |  | Insert into project values (5, 'mars', 14500); |  |
| t5 | Select \* from project; |  | Select \* from project; |  |
| t6 | Rollback; |  |  |  |
| t7 |  |  | Commit; |  |
| t8 | Select \* from project; |  | Select \* from project; |  |

1. F

| **Time** | **Session1** | **Output1** | **Session2** | **Output2** |
| --- | --- | --- | --- | --- |
| t0 | SET TRANSACTION NAME 'cost\_update8'; |  | SET TRANSACTION NAME 'cost\_update9'; |  |
| t1 | Update project set cost=3490  where pname='mercury'; |  |  |  |
| t2 | Select \* from project; |  | Select \* from project; |  |
| t3 |  |  | Update project set cost=298  Where pname='saturn'; |  |
| t4 | Select \* from project; |  | Select \* from project; |  |
| t5 | Create table test1 (id number); |  |  |  |
| t6 | Rollback; |  |  |  |
| t7 | Select \* from project; |  | Select \* from project; |  |
| t8 |  |  | Commit; |  |
| t9 | Select \* from project; |  | Select \* from project; |  |

1. **Transaction Processing in PL/SQL**

CREATE TABLE accounts (account\_id NUMBER(6), balance NUMBER (10,2),

check (balance>=0));

INSERT INTO accounts VALUES (7715, 6350.00);

INSERT INTO accounts VALUES (7720, 5100.50);

COMMIT;

Following is the difference circumstance of transferring money

1. Example: transfer money ($250) from account 7715 to 7720

|  |  |
| --- | --- |
| **Time** | **Session** |
| t0 | DECLARE  transfer NUMBER(8,2) := 250;  BEGIN  UPDATE accounts SET balance = balance - transfer WHERE account\_id = 7715;  UPDATE accounts SET balance = balance + transfer WHERE account\_id = 7720;  COMMIT;  END; |
| t1 | SELECT \* FROM accounts; |

1. Example: Transfer money ($9000) from account 7715 to 7720

|  |  |
| --- | --- |
| **Time** | **Session** |
| t0 | SELECT \* FROM accounts; |
| t1 | DECLARE  transfer NUMBER(8,2) := 9000;  BEGIN  UPDATE accounts SET balance = balance - transfer WHERE account\_id = 7715;  UPDATE accounts SET balance = balance + transfer WHERE account\_id = 7720;  COMMIT;  END; |
| t2 | SELECT \* FROM accounts; |

1. Example: Transfer money ($9000) from account 7715 to 7720

|  |  |
| --- | --- |
| **Time** | **Session** |
| t0 | SELECT \* FROM accounts; |
| t1 | DECLARE  transfer NUMBER(8,2) := 9000;  BEGIN  UPDATE accounts SET balance = balance + transfer WHERE account\_id = 7720;  UPDATE accounts SET balance = balance - transfer WHERE account\_id = 7715;  COMMIT;  END; |
| t2 | SELECT \* FROM accounts; |

1. Example: Transfer money ($9000) from account 7715 to 7720

|  |  |
| --- | --- |
| **Time** | **Session** |
| t0 | SELECT \* FROM accounts; |
| t1 | DECLARE  transfer NUMBER(8,2) := 9000;  BEGIN  UPDATE accounts SET balance = balance + transfer WHERE account\_id = 7720;  COMMIT;  UPDATE accounts SET balance = balance - transfer WHERE account\_id = 7715;  COMMIT;  END; |
| t2 | SELECT \* FROM accounts; |

1. Example: PL/SQL WITH EXCEPTION

|  |  |
| --- | --- |
| **Time** | **Session** |
| t0 | SELECT \* FROM accounts; |
| t1 | DECLARE  transfer NUMBER(8,2) := 9000;  BEGIN  UPDATE accounts SET balance = balance + transfer WHERE account\_id = 7720;  UPDATE accounts SET balance = balance - transfer WHERE account\_id = 7715;  COMMIT;  EXCEPTION WHEN **OTHERS** THEN  **Dbms\_output.put\_line ('error!!!!!!!!! ');**  END; |
| t2 | SELECT \* FROM accounts; |

1. Example: PL/SQL WITH EXCEPTION

|  |  |
| --- | --- |
| **Time** | **Session** |
| t0 | SELECT \* FROM accounts; |
| t1 | DECLARE  transfer NUMBER(8,2) := 9000;  BEGIN  UPDATE accounts SET balance = balance + transfer WHERE account\_id = 7720;  UPDATE accounts SET balance = balance - transfer WHERE account\_id = 7715;  COMMIT;  EXCEPTION WHEN OTHERS THEN  **RAISE;**  END; |
| t2 | SELECT \* FROM accounts; |