TSHEPISO MABITSI ST10197389

ADDB6311 PRACTICUM

QUESTION 1

```
CREATE TABLE INSTRUCTOR
(
ins_ID NUMBER(5) NOT NULL,
ins_fname VARCHAR2(15) NOT NULL,
ins_sname VARCHAR2(15) NOT NULL,
ins_contact NUMBER(10) NOT NULL,
ins_level NUMBER(5) NOT NULL,
CONSTRAINT PK_INSTRUCTOR PRIMARY KEY(ins_ID)
);

CREATE TABLE CUSTOMER
(
cust_id VARCHAR2(5) NOT NULL,
cust_fname VARCHAR2(15) NOT NULL,
cust_sname VARCHAR2(15) NOT NULL,
cust_address VARCHAR2(20) NOT NULL,
cust_contact NUMBER(10) NOT NULL,
CONSTRAINT PK_CUSTOMER PRIMARY KEY(cust_id)
);

CREATE TABLE DIVE
(
DIVE_ID NUMBER(5) NOT NULL,
DIVE_NAME VARCHAR2(50) NOT NULL,
```

```
DIVE_DURATION VARCHAR2(50) NOT NULL,
DIVE_LOCATION VARCHAR2(20) NOT NULL,
DIVE_EXP_LEVEL NUMBER(5) NOT NULL,
DIVE_COST NUMBER(10) NOT NULL,
CONSTRAINT PK_DIVE PRIMARY KEY(DIVE_ID)
);


CREATE TABLE DIVE_EVENT
(
DIVE_EVENT_ID VARCHAR2(10) NOT NULL,
DIVE_DATE DATE NOT NULL,
DIVE_PARTICIPANTS NUMBER(5) NOT NULL,
ins_ID NUMBER(5) NOT NULL,
cust_id VARCHAR2(5) NOT NULL,
DIVE_ID NUMBER(5) NOT NULL,
CONSTRAINT PK_DIVE_EVENT PRIMARY KEY(DIVE_EVENT_ID),
CONSTRAINT FK_INSTRUCTOR FOREIGN KEY (ins_ID) REFERENCES INSTRUCTOR(ins_ID),
CONSTRAINT FK_CUSTOMER FOREIGN KEY (cust_id) REFERENCES CUSTOMER(cust_id),
CONSTRAINT FK_DIVE FOREIGN KEY (DIVE_ID) REFERENCES DIVE(DIVE_ID)
);



INSERT ALL
INTO INSTRUCTOR VALUES(101, 'James', 'Willis', 0843569851, 7)
INTO INSTRUCTOR VALUES(102, 'Sam', 'Wait', 0763698521, 2)
INTO INSTRUCTOR VALUES(103, 'Sally', 'Gumede', 0786598521, 8)
```

```
INTO INSTRUCTOR VALUES(104, 'Bob', 'Du Preez', 0796369857, 3)

INTO INSTRUCTOR VALUES(105, 'Simon', 'Jones', 0826598741, 9)

SELECT * FROM DUAL;


INSERT ALL

INTO CUSTOMER VALUES('C115', 'Heinrich', 'Willis', '3 Main Road', 0821253659)

INTO CUSTOMER VALUES('C116', 'David', 'Watson', '13 Cape Road',  0769658547)

INTO CUSTOMER VALUES('C117', 'Waldo', 'Smith', '3 Mountain Road', 0863256574)

INTO CUSTOMER VALUES('C118', 'Alex', 'Hanson', '8 Circle Road',  0762356587)

INTO CUSTOMER VALUES('C119', 'Kuhle', 'Bitterhout',  '15 Main Road',
      0821235258)

INTO CUSTOMER VALUES('C120', 'Thando', 'Zolani', '88 Summer Road',
0847541254)

INTO CUSTOMER VALUES('C121', 'Philip', 'Jackson', '3 Long Road', 0745556658)

INTO CUSTOMER VALUES('C122', 'Sarah', 'Jones', '7 Sea Road', 0814745745)

INTO CUSTOMER VALUES('C123', 'Catherine', 'Howard', '31 Lake Side Road',
0822232521)

SELECT * FROM DUAL;



INSERT ALL

INTO DIVE VALUES(550, 'Shark Dive', '3 hours', 'Shark Point', 8, 500)

INTO DIVE VALUES(551, 'Coral Dive', '1 hour', 'Break Point', 7, 300)

INTO DIVE VALUES(552, 'Wave Crescent', '2 hours', 'Ship wreck ally', 3, 800)


INTO DIVE VALUES(553, 'Underwater Exploration', '1 hour', 'Coral ally', 2,
250)

INTO DIVE VALUES(554, 'Underwater Adventure', '3 hours', 'Sandy Beach', 3,
750)

INTO DIVE VALUES(555, 'Deep Blue Ocean', '30 minutes', 'Lazy Waves', 2, 120)
```

```
INTO DIVE VALUES(556, 'Rough Seas', '1 hour', 'Pipe', 9, 700)
INTO DIVE VALUES(557, 'White Water', '2 hours', 'Drifts', 5, 200)
INTO DIVE VALUES(558, 'Current Adventure', '2 hours', 'Rock Lands', 3, 150)

SELECT * FROM DUAL;


INSERT ALL
INTO DIVE_EVENT VALUES('de_101', '15-Jul-17', 5, 103, 'C115', 558)
INTO DIVE_EVENT VALUES('de_102', '16-Jul-17', 7, 102, 'C117', 555)
INTO DIVE_EVENT VALUES('de_103', '18-Jul-17', 8, 104, 'C118', 552)
INTO DIVE_EVENT VALUES('de_104', '19-Jul-17', 3, 101, 'C119', 551)
INTO DIVE_EVENT VALUES('de_105', '21-Jul-17', 5, 104, 'C121', 558)
INTO DIVE_EVENT VALUES('de_106', '22-Jul-17', 8, 105, 'C120', 556)
INTO DIVE_EVENT VALUES('de_107', '25-Jul-17', 10, 105, 'C115', 554)
INTO DIVE_EVENT VALUES('de_108', '27-Jul-17', 5, 101, 'C122', 552)
INTO DIVE_EVENT VALUES('de_109', '28-Jul-17', 3, 102, 'C123', 553)
SELECT * FROM DUAL;


SELECT * FROM INSTRUCTOR;
```

| | INS_ID | INS_FNAME | INS_SNAME | INS_CONTACT | INS_LEVEL |
|---|---|---|---|---|---|
| 1 | 101 | James | Willis | 843569851 | 7 |
| 2 | 102 | Sam | Wait | 763698521 | 2 |
| 3 | 103 | Sally | Gumede | 786598521 | 8 |
| 4 | 104 | Bob | Du Preez | 796369857 | 3 |
| 5 | 105 | Simon | Jones | 826598741 | 9 |

```
SELECT * FROM CUSTOMER;
```

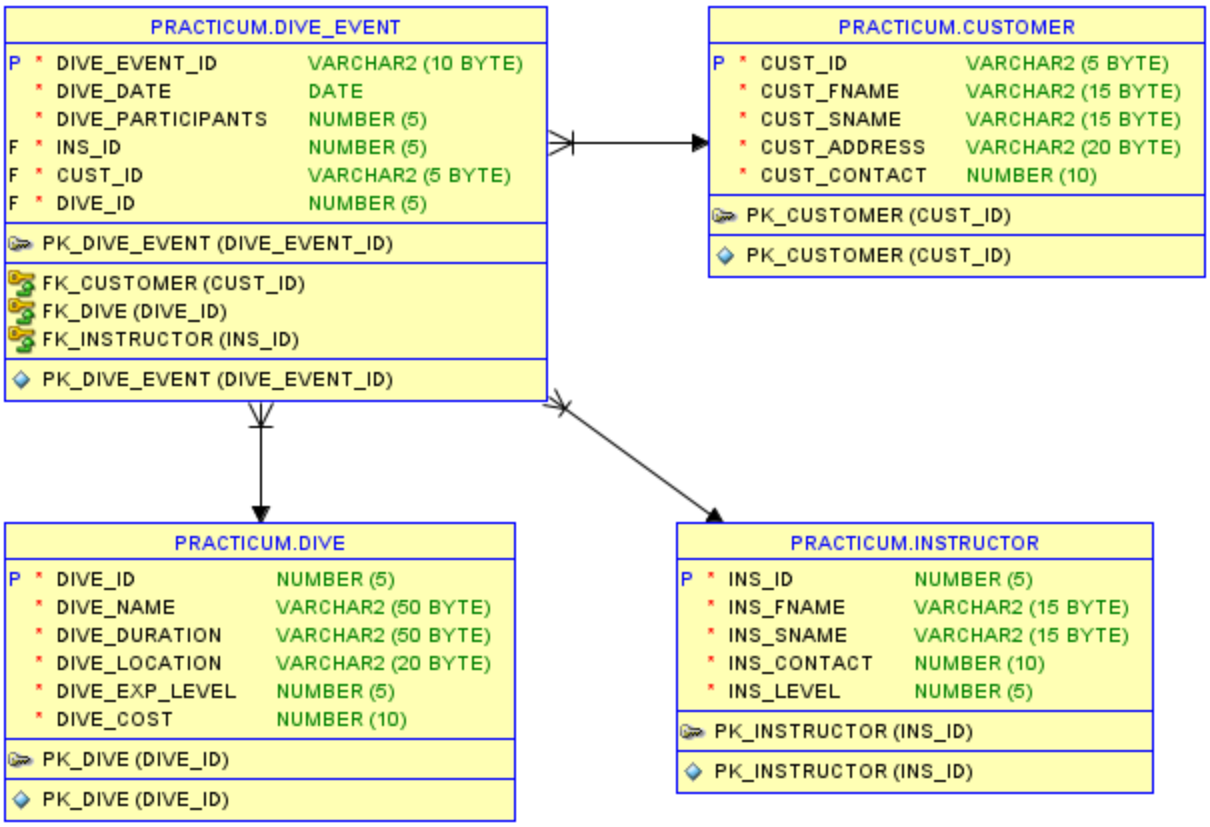
| | CUST_ID | CUST_FNAME | CUST_SNAME | CUST_ADDRESS | CUST_CONTACT |
|---|---|---|---|---|---|
| 1 | C115 | Heinrich | Willis | 3 Main Road | 821253659 |
| 2 | C116 | David | Watson | 13 Cape Road | 769658547 |
| 3 | C117 | Waldo | Smith | 3 Mountain Road | 863256574 |
| 4 | C118 | Alex | Hanson | 8 Circle Road | 762356587 |
| 5 | C119 | Kuhle | Bitterhout | 15 Main Road | 821235258 |
| 6 | C120 | Thando | Zolani | 88 Summer Road | 847541254 |
| 7 | C121 | Philip | Jackson | 3 Long Road | 745556658 |
| 8 | C122 | Sarah | Jones | 7 Sea Road | 814745745 |
| 9 | C123 | Catherine | Howard | 31 Lake Side Road | 822232521 |

```
SELECT * FROM DIVE;
```

| | DIVE_ID | DIVE_NAME | DIVE_DURATION | DIVE_LOCATION | DIVE_EXP_LEVEL | DIVE_COST |
|---|---|---|---|---|---|---|
| 1 | 550 | Shark Dive | 3 hours | Shark Point | 8 | 500 |
| 2 | 551 | Coral Dive | 1 hour | Break Point | 7 | 300 |
| 3 | 552 | Wave Crescent | 2 hours | Ship wreck ally | 3 | 800 |
| 4 | 553 | Underwater Exploration | 1 hour | Coral ally | 2 | 250 |
| 5 | 554 | Underwater Adventure | 3 hours | Sandy Beach | 3 | 750 |
| 6 | 555 | Deep Blue Ocean | 30 minutes | Lazy Waves | 2 | 120 |
| 7 | 556 | Rough Seas | 1 hour | Pipe | 9 | 700 |
| 8 | 557 | White Water | 2 hours | Drifts | 5 | 200 |
| 9 | 558 | Current Adventure | 2 hours | Rock Lands | 3 | 150 |

```
SELECT * FROM DIVE_EVENT;
```

| | DIVE_EVENT_ID | DIVE_DATE | DIVE_PARTICIPANTS | INS_ID | CUST_ID | DIVE_ID |
|---|---|---|---|---|---|---|
| 1 | de_101 | 15-JUL-17 | 5 | 103 | C115 | 558 |
| 2 | de_102 | 16-JUL-17 | 7 | 102 | C117 | 555 |
| 3 | de_103 | 18-JUL-17 | 8 | 104 | C118 | 552 |
| 4 | de_104 | 19-JUL-17 | 3 | 101 | C119 | 551 |
| 5 | de_105 | 21-JUL-17 | 5 | 104 | C121 | 558 |
| 6 | de_106 | 22-JUL-17 | 8 | 105 | C120 | 556 |
| 7 | de_107 | 25-JUL-17 | 10 | 105 | C115 | 554 |
| 8 | de_108 | 27-JUL-17 | 5 | 101 | C122 | 552 |
| 9 | de_109 | 28-JUL-17 | 3 | 102 | C123 | 553 |




```
--QUESTION 2

ALTER SESSION SET "_ORACLE_SCRIPT"=TRUE;

--CHARLIE IS THE ADMINISTRATOR

CREATE USER CHARLIE IDENTIFIED BY ADMINISTRATOR;

GRANT SELECT ANY TABLE TO CHARLIE;

GRANT INSERT ANY TABLE TO CHARLIE;

GRANT UPDATE ANY TABLE TO CHARLIE;


--JOHN IS A GENERAL USER

CREATE USER JOHN IDENTIFIED BY USER;

GRANT SELECT ANY TABLE TO JOHN;

GRANT INSERT ANY TABLE TO JOHN;
```

```
Session altered.


User CHARLIE created.


Grant succeeded.


Grant succeeded.


Grant succeeded.


User JOHN created.


Grant succeeded.


Grant succeeded.
```

```
--QUESTION 3

SELECT I.ins_fname|| ' '|| I.ins_sname AS INSTRUCTOR, C.cust_fname|| ' '||
C.cust_sname AS CUSTOMER, D.DIVE_LOCATION, DE.DIVE_PARTICIPANTS

FROM DIVE_EVENT DE

INNER JOIN INSTRUCTOR I ON DE.ins_ID = I.ins_ID

INNER JOIN CUSTOMER C ON DE.cust_id = C.cust_id

INNER JOIN DIVE D ON DE.DIVE_ID = D.DIVE_ID

WHERE DE.DIVE_PARTICIPANTS BETWEEN 8 AND 10;
```

| | INSTRUCTOR | CUSTOMER | DIVE_LOCATION | DIVE_PARTICIPANTS |
|---|---|---|---|---|
| 1 | Bob Du Preez | Alex Hanson | Ship wreck ally | 8 |
| 2 | Simon Jones | Thando Zolani | Pipe | 8 |
| 3 | Simon Jones | Heinrich Willis | Sandy Beach | 10 |

```
--QUESTION 4

SET SERVEROUTPUT ON;
```

```
DECLARE
    V_Dname DIVE.DIVE_NAME%TYPE;
    V_DATE DIVE_EVENT.DIVE_DATE%TYPE;
    V_PARTICIPANTS DIVE_EVENT.DIVE_PARTICIPANTS%TYPE;
CURSOR c1
IS
SELECT D.DIVE_NAME, DE.DIVE_DATE, DE.DIVE_PARTICIPANTS
FROM DIVE_EVENT DE
INNER JOIN DIVE D ON DE.DIVE_ID = D.DIVE_ID
WHERE DE.DIVE_PARTICIPANTS >= 10;
BEGIN
FOR rec IN c1
LOOP
    V_Dname := rec.DIVE_NAME;
    V_DATE := rec.DIVE_DATE;
    V_PARTICIPANTS := rec.DIVE_PARTICIPANTS;
DBMS_OUTPUT.PUT_LINE('DIVE NAME: ' || V_Dname);
DBMS_OUTPUT.PUT_LINE('DIVE DATE: ' || V_DATE);
DBMS_OUTPUT.PUT_LINE('PARTICIPANTS: ' || V_PARTICIPANTS);
DBMS_OUTPUT.PUT_LINE('--------------------------------');
END LOOP;
END;
/
```

```
DIVE NAME: Underwater Adventure
DIVE DATE: 25-JUL-17
PARTICIPANTS: 10
--------------------------------


PL/SQL procedure successfully completed.
```

```
--QUESTION 5
SET SERVEROUTPUT ON;
DECLARE
    V_Fname CUSTOMER.cust_fname%TYPE;
    V_Sname CUSTOMER.cust_sname%TYPE;
    V_Dname DIVE.DIVE_NAME%TYPE;
    V_PARTICIPANTS DIVE_EVENT.DIVE_PARTICIPANTS%TYPE;
CURSOR c1
IS
SELECT C.cust_fname, C.cust_sname, D.DIVE_NAME, DE.DIVE_PARTICIPANTS
FROM DIVE_EVENT DE
INNER JOIN CUSTOMER C ON DE.cust_id = C.cust_id
INNER JOIN DIVE D ON DE.DIVE_ID = D.DIVE_ID
WHERE D.DIVE_COST > 500;
BEGIN
FOR rec IN c1
LOOP
    V_Fname := rec.cust_fname;
    V_Sname := rec.cust_sname;
    V_Dname := rec.DIVE_NAME;
    V_PARTICIPANTS := rec.DIVE_PARTICIPANTS;
IF V_PARTICIPANTS <= 4 THEN
DBMS_OUTPUT.PUT_LINE('CUSTOMER:   ' || V_Fname ||', '||V_Sname);
DBMS_OUTPUT.PUT_LINE('DIVE NAME:  ' || V_Dname);
DBMS_OUTPUT.PUT_LINE('PARTICIPANTS:   ' || V_PARTICIPANTS);
DBMS_OUTPUT.PUT_LINE('STATUS: ' || '1 INSTRUCTOR REQUIRED');
DBMS_OUTPUT.PUT_LINE('--------------------------');
ELSE IF V_PARTICIPANTS >= 5 AND V_PARTICIPANTS < 7 THEN
```

```
DBMS_OUTPUT.PUT_LINE('CUSTOMER:   ' || V_Fname ||', '||V_Sname);
DBMS_OUTPUT.PUT_LINE('DIVE NAME:  ' || V_Dname);
DBMS_OUTPUT.PUT_LINE('PARTICIPANTS:   ' || V_PARTICIPANTS);
DBMS_OUTPUT.PUT_LINE('STATUS: ' || '2 INSTRUCTOR REQUIRED');
DBMS_OUTPUT.PUT_LINE('-------------------------');
ELSE
DBMS_OUTPUT.PUT_LINE('CUSTOMER:   ' || V_Fname ||', '||V_Sname);
DBMS_OUTPUT.PUT_LINE('DIVE NAME:  ' || V_Dname);
DBMS_OUTPUT.PUT_LINE('PARTICIPANTS:   ' || V_PARTICIPANTS);
DBMS_OUTPUT.PUT_LINE('STATUS: ' || '3 INSTRUCTOR REQUIRED');
DBMS_OUTPUT.PUT_LINE('-------------------------');
END IF;
END IF;
END LOOP;
END;
/
```

```
CUSTOMER:   Heinrich, Willis
DIVE NAME:  Underwater Adventure
PARTICIPANTS:   10
STATUS: 3 INSTRUCTOR REQUIRED
--------------------------
CUSTOMER:   Alex, Hanson
DIVE NAME:  Wave Crescent
PARTICIPANTS:   8
STATUS: 3 INSTRUCTOR REQUIRED
--------------------------
CUSTOMER:   Thando, Zolani
DIVE NAME:  Rough Seas
PARTICIPANTS:   8
STATUS: 3 INSTRUCTOR REQUIRED
--------------------------
CUSTOMER:   Sarah, Jones
DIVE NAME:  Wave Crescent
PARTICIPANTS:   5
STATUS: 2 INSTRUCTOR REQUIRED
--------------------------


PL/SQL procedure successfully completed.
```

```
--QUESTION 6

CREATE OR REPLACE VIEW Vw_Dive_Event

AS

SELECT I.ins_ID, C.cust_id, C.cust_address, D.DIVE_DURATION, DE.DIVE_DATE

FROM DIVE_EVENT DE

INNER JOIN CUSTOMER C ON DE.cust_id = C.cust_id

INNER JOIN INSTRUCTOR I ON DE.ins_ID = I.ins_ID

INNER JOIN DIVE D ON DE.DIVE_ID = D.DIVE_ID

WHERE DE.DIVE_DATE < '19-JUL-17';

SELECT * FROM Vw_Dive_Event;

View VW_DIVE_EVENT created.
```

| | INS_ID | CUST_ID | CUST_ADDRESS | DIVE_DURATION | DIVE_DATE |
|---|---|---|---|---|---|
| 1 | 103 | C115 | 3 Main Road | 2 hours | 15-JUL-17 |
| 2 | 102 | C117 | 3 Mountain Road | 30 minutes | 16-JUL-17 |
| 3 | 104 | C118 | 8 Circle Road | 2 hours | 18-JUL-17 |

```
--QUESTION 7

CREATE OR REPLACE TRIGGER New_Dive_Event

BEFORE INSERT ON DIVE_EVENT

FOR EACH ROW

BEGIN

    IF NEW.DIVE_PARTICIPANTS <= 0 OR NEW.DIVE_PARTICIPANTS > 20 THEN

        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Participants must be
between 1 and 20';

    END IF;

END;

/

--QUESTION 8

CREATE OR REPLACE PROCEDURE sp_Customer_Details (

    p_cust_id IN VARCHAR2,

    p_dive_date IN DATE

)

IS

    V_CUSTOMER VARCHAR2(100);

    V_Dname VARCHAR2(100);

BEGIN

    -- Retrieve customer's full name

    SELECT cust_fname || ' ' || cust_sname

    INTO V_CUSTOMER

    FROM CUSTOMER

    WHERE CUST_ID = p_cust_id;
```

```
    -- Retrieve dive name booked for the specified date and customer
    SELECT D.DIVE_NAME
    INTO V_Dname
    FROM DIVE_EVENT DE
    JOIN DIVE D ON DE.DIVE_ID = D.DIVE_ID
    WHERE DE.CUST_ID = p_cust_id
    AND DE.DIVE_DATE = p_dive_date;

    -- DISPAY RESULTS
DBMS_OUTPUT.PUT_LINE('CUSTOMER DETAILS: ' ||
V_Dname || ' booked for ' ||
V_Dname || ' on ' || TO_CHAR(p_dive_date, 'DD/MON/YYYY'));

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Error: No booking found for Customer ID ' ||
p_cust_id || ' on ' || TO_CHAR(p_dive_date, 'DD/MON/YYYY'));
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error: An error occurred while retrieving
customer details.');
END sp_Customer_Details;
/

EXEC sp_Customer_Details;
```

```
Procedure SP_CUSTOMER_DETAILS compiled
```