Name: Celebrate

Surname: Mashaba

Student number: ST10055395

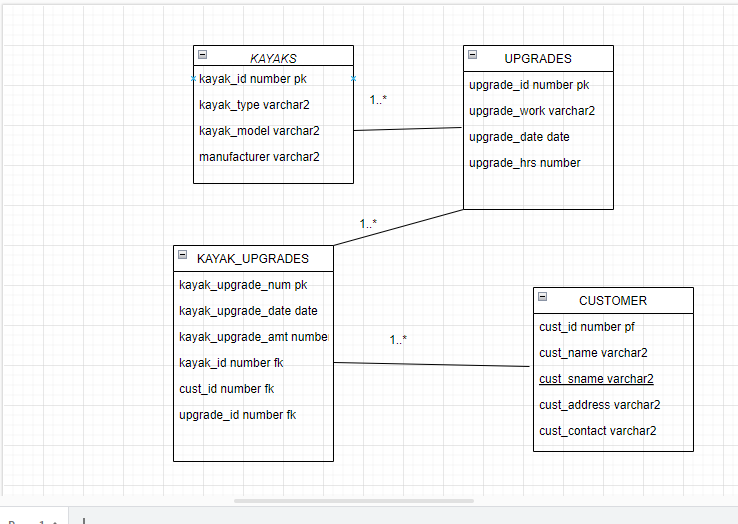
Module name: ADVANCED DATABASES

Module code: ADDB6311

Assessment Type: Assignment 1

Group: 2

Question 1



QUESTION 2

- Create KAYAKS table

CREATE TABLE KAYAKS (

KAYAK\_ID NUMBER PRIMARY KEY,

KAYAK\_TYPE VARCHAR2(50),

KAYAK\_MODEL VARCHAR2(50),

MANUFACTURER VARCHAR2(50)

);

-- Create CUSTOMER table

CREATE TABLE CUSTOMER (

CUST\_ID VARCHAR2(5) PRIMARY KEY,

CUST\_FNAME VARCHAR2(50),

CUST\_SNAME VARCHAR2(50),

CUST\_ADDRESS VARCHAR2(100),

CUST\_CONTACT VARCHAR2(15)

);

-- Create UPGRADES table

CREATE TABLE UPGRADES (

UPGRADE\_ID NUMBER PRIMARY KEY,

UPGRADE\_WORK VARCHAR2(50),

UPGRADE\_DATE DATE,

UPGRADE\_HRS NUMBER

);

-- Create KAYAK\_UPGRADES table

CREATE TABLE KAYAK\_UPGRADES (

KAYAK\_UPGRADE\_NUM NUMBER PRIMARY KEY,

KAYAK\_UPGRADE\_DATE DATE,

KAYAK\_UPGRADE\_AMT NUMBER,

KAYAK\_ID NUMBER,

CUST\_ID VARCHAR2(5),

UPGRADE\_ID NUMBER,

FOREIGN KEY (KAYAK\_ID) REFERENCES KAYAKS(KAYAK\_ID),

FOREIGN KEY (CUST\_ID) REFERENCES CUSTOMER(CUST\_ID),

FOREIGN KEY (UPGRADE\_ID) REFERENCES UPGRADES(UPGRADE\_ID)

);

-- Populate KAYAK\_UPGRADES table

insert into KAYAKS(kayak\_id, kayak\_type, kayak\_model, manufacturer) values(12345, 'Single Seater', 'K100', 'FeelFree');

insert into KAYAKS(kayak\_id, kayak\_type, kayak\_model, manufacturer) values(54321, 'Tandem Seater', 'J55', 'Roamer');

insert into KAYAKS(kayak\_id, kayak\_type, kayak\_model, manufacturer) values(78945, 'Fishing Kayak','H9000', 'Wavesport');

insert into KAYAKS(kayak\_id, kayak\_type, kayak\_model, manufacturer) values(98754, 'Hobie Kayak','A450', 'Gemini');

insert into KAYAKS(kayak\_id, kayak\_type, kayak\_model, manufacturer) values(55311, 'Canadian Style','L920', 'Surge');

Select \* from KAYAKS;

-- Populate CUSTOMER table

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C115', 'Jeff', 'Willis', '3 Main Road', '0821253659');

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C116', 'Andre', 'Watson', '13 Cape Road', '0769658547');

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C117', 'Wallis', 'Smith', '3 Mountain Road', '0863256574');

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C118', 'Alex', 'Hanson', '8 Circle Road', '0762356587');

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C119', 'Bob', 'Bitterhout', '15 Main Road', '0821235258');

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C120', 'Thando', 'Zolani', '88 Summer Road', '0847541254');

insert into CUSTOMER(cust\_id, cust\_fname, cust\_sname, cust\_address, cust\_contact) values('C121', 'Philip', 'Jackson', '3 Long Road', '0745556658');

INSERT INTO CUSTOMER (CUST\_ID, CUST\_FNAME, CUST\_SNAME, CUST\_ADDRESS, CUST\_CONTACT) VALUES ('C121', 'Philip', 'Jackson', '3 Long Road', '0745556658');

INSERT INTO CUSTOMER (CUST\_ID, CUST\_FNAME, CUST\_SNAME, CUST\_ADDRESS, CUST\_CONTACT) VALUES ('C122', 'Sarah', 'Jones', '7 Sea Road', '0814745745');

Select \* from CUSTOMER;

-- Populate UPGRADES table

insert into UPGRADES(upgrade\_id, upgrade\_work, upgrade\_date, upgrade\_hrs) values(1, 'Sonar Device', '15 July 2022', 5);

insert into UPGRADES(upgrade\_id, upgrade\_work, upgrade\_date, upgrade\_hrs) values(2, 'Padded Seats', '18 July 2022', 3);

insert into UPGRADES(upgrade\_id, upgrade\_work, upgrade\_date, upgrade\_hrs) values(3, 'GoPro Camera Mount','19 July 2022', 10);

Select \* from UPGRADES;

-- Populate KAYAK\_UPGRADES table

insert into KAYAK\_UPGRADES(kayak\_upgrade\_num, kayak\_upgrade\_date, kayak\_upgrade\_amt, kayak\_id,cust\_id, upgrade\_id) values(101, '27 July 2019', 75, 98754,'C121',3);

insert into KAYAK\_UPGRADES(kayak\_upgrade\_num, kayak\_upgrade\_date, kayak\_upgrade\_amt, kayak\_id,cust\_id, upgrade\_id) values(102, '20 July 2019', 30, 12345,'C120',2);

insert into KAYAK\_UPGRADES(kayak\_upgrade\_num, kayak\_upgrade\_date, kayak\_upgrade\_amt, kayak\_id,cust\_id, upgrade\_id) values(103, '23 July 2019', 75, 55311,'C119',1);

insert into KAYAK\_UPGRADES(kayak\_upgrade\_num, kayak\_upgrade\_date, kayak\_upgrade\_amt, kayak\_id,cust\_id, upgrade\_id) values(104, '17 July 2019', 50, 54321,'C117',1);

insert into KAYAK\_UPGRADES(kayak\_upgrade\_num, kayak\_upgrade\_date, kayak\_upgrade\_amt, kayak\_id,cust\_id, upgrade\_id) values(105, '19 July 2019', 30, 12345,'C122',2);

Select \* from KAYAK\_UPGRADES;

SELECT \* FROM KAYAKS;

SELECT \* FROM CUSTOMER;

SELECT \* FROM UPGRADES;

SELECT \* FROM KAYAK\_UPGRADES;

QUESTION 3

-- Creating users

CREATE USER Tshepo IDENTIFIED BY tmphoabc2023;

CREATE USER Mya IDENTIFIED BY mrobertabc2023;

-- Granting privileges

GRANT CREATE SESSION TO Tshepo;

GRANT CREATE SESSION TO Mya;

GRANT SELECT ANY TABLE TO Tshepo;

GRANT INSERT ANY TABLE TO Mya;

QUESTION 4

--QUESTION 4

-- Create a SQL query to display the kayak id, customer id, upgrade hours and the kayak upgrade

--amount. In your query, include the total sales amount by multiplying the upgrade hours by the

--kayak upgrade amount

SELECT K.KAYAK\_ID, CU.CUST\_ID, U.UPGRADE\_HRS UPGRADE\_HRS,

('R ' || KAYAK\_UPGRADE\_AMT) KAYAK\_UPGRADE\_AMT,

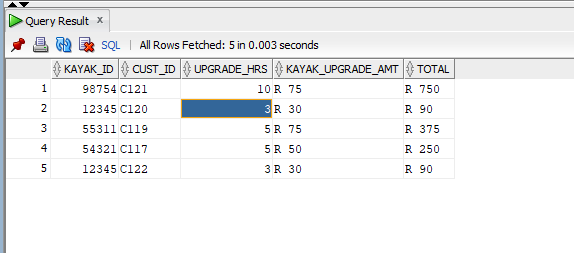
('R ' || (U.UPGRADE\_HRS \* KU.KAYAK\_UPGRADE\_AMT)) TOTAL

FROM KAYAKS K, CUSTOMER CU, UPGRADES U, KAYAK\_UPGRADES KU

WHERE K.KAYAK\_ID = KU.KAYAK\_ID

AND CU.CUST\_ID = KU.CUST\_ID

AND U.UPGRADE\_ID = KU.UPGRADE\_ID;



--QUESTION 5

--create a sql query to display the customer's full name,type of kayak, upgrade hours, upgrade work and kayak upgrade amount

SELECT

CONCAT(C.cust\_fname, C.cust\_sname) AS customer\_name,

K.kayak\_type,

UPG.upgrade\_hrs,

UPG.upgrade\_work,

KU.kayak\_upgrade\_amt

FROM

CUSTOMER C

JOIN

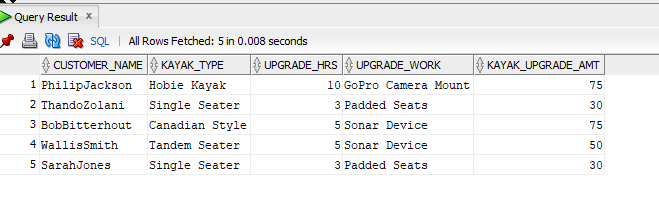
KAYAK\_UPGRADES KU ON C.cust\_id = KU.cust\_id

JOIN

KAYAKS K ON KU.kayak\_id = K.kayak\_id

JOIN

UPGRADES UPG ON KU.upgrade\_id = UPG.upgrade\_id;



QUESTION 6

--QUESTION 6

--1. Create a PL/SQL query that will display the customer id, upgrade work done and the

--upgrade amount. In your query, only display the results where the upgrade amount is

--greater than R 50.

--2. Identify the type of cursor used and provide two reasons for its suitability in this query.

--Provide this information as a comment at the end of your code.

--3. Provide screenshot of query output.

SET SERVEROUTPUT ON

DECLARE

C\_ID CUSTOMER.CUST\_ID%TYPE;

U\_WORK UPGRADES.UPGRADE\_WORK%TYPE;

U\_AMT KAYAK\_UPGRADES.KAYAK\_UPGRADE\_AMT%TYPE;

CURSOR INFO IS

SELECT CU.CUST\_ID, U.UPGRADE\_WORK, KU.KAYAK\_UPGRADE\_AMT

FROM CUSTOMER CU, UPGRADES U, KAYAK\_UPGRADES KU

WHERE CU.CUST\_ID = KU.CUST\_ID

AND U.UPGRADE\_ID = KU.UPGRADE\_ID

AND KU.KAYAK\_UPGRADE\_AMT > 50;

BEGIN

FOR REC IN INFO

LOOP

C\_ID:=REC.CUST\_ID;

U\_WORK:=REC.UPGRADE\_WORK;

U\_AMT:= REC.KAYAK\_UPGRADE\_AMT;

DBMS\_OUTPUT.PUT\_LINE('CUSTOMER\_ID: ' || C\_ID || ', ' || CHR(10) ||

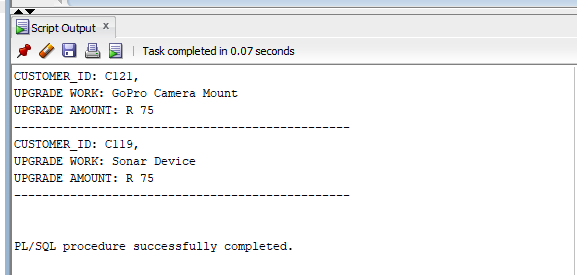
'UPGRADE WORK: ' || U\_WORK || CHR(10) ||

'UPGRADE AMOUNT: R ' || U\_AMT);

DBMS\_OUTPUT.PUT\_LINE('------------------------------------------------');

END LOOP;

END;



QUESTION 7

--QUESTION 7

--. Create a PL/SQL query that will display the customer name, type of kayak, upgrade work

--and the upgrade date. In your query display the upgrade amount and a discount amount

--(10%), that will be subtracted off customer 'C121' final bill only.

--2. Provide screenshot of query output.

SET SERVEROUTPUT ON

DECLARE

C\_NAME CUSTOMER.CUST\_SNAME%TYPE;

K\_TYPE KAYAKS.KAYAK\_TYPE%TYPE;

U\_WORK UPGRADES.UPGRADE\_WORK%TYPE;

KU\_DATE KAYAK\_UPGRADES.KAYAK\_UPGRADE\_DATE%TYPE;

U\_AMT KAYAK\_UPGRADES.KAYAK\_UPGRADE\_AMT%TYPE;

DISCOUNT KAYAK\_UPGRADES.KAYAK\_UPGRADE\_AMT%TYPE;

CURSOR INFO IS

SELECT (C.CUST\_FNAME || ' ' || C.CUST\_SNAME) CUSTOMER, K.KAYAK\_TYPE, U.UPGRADE\_WORK,

KU.KAYAK\_UPGRADE\_DATE, KU.KAYAK\_UPGRADE\_AMT, (KU.KAYAK\_UPGRADE\_AMT \* 0.10)

DISCOUNT\_AMOUNT

FROM CUSTOMER C, KAYAKS K, UPGRADES U, KAYAK\_UPGRADES KU

WHERE C.CUST\_ID = KU.CUST\_ID

AND K.KAYAK\_ID = KU.KAYAK\_ID

AND U.UPGRADE\_ID = KU.UPGRADE\_ID

AND C.CUST\_ID = 'C121';

BEGIN

FOR REC IN INFO

LOOP

C\_NAME:=REC.CUSTOMER;

K\_TYPE:=REC.KAYAK\_TYPE;

U\_WORK:=REC.UPGRADE\_WORK;

KU\_DATE:=REC.KAYAK\_UPGRADE\_DATE;

U\_AMT:=REC.KAYAK\_UPGRADE\_AMT;

DISCOUNT:=REC.DISCOUNT\_AMOUNT;

DBMS\_OUTPUT.PUT\_LINE('CUSTOMER: ' || C\_NAME || CHR(10) ||

'KAYAK TYPE: ' || K\_TYPE || CHR(10) ||

'UPGRADE WORK: ' || U\_WORK || CHR(10) ||

'UPGRADE DATE: ' || KU\_DATE || CHR(10) ||

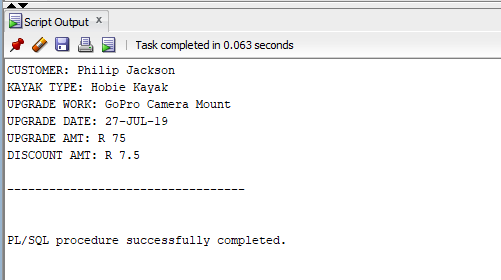
'UPGRADE AMT: R ' || U\_AMT || CHR(10) ||

'DISCOUNT AMT: R ' || DISCOUNT || CHR(10));

DBMS\_OUTPUT.PUT\_LINE('----------------------------------');

END LOOP;

END;



QUESTION 8

--QUESTION 8

--1. Create a view called vwCustomerUpgrades that will display the combined customer name,

--kayak type, upgrade work and the customer contact number. In your query, only display

--the customers whose address contains the word 'Summer'. 2. Briefly explain the concept of vertical and horizontal partitioning based on this scenario .

--Provide this information as a comment at the end of your code.

--3. Provide screenshot of query output.

CREATE OR REPLACE VIEW VWCUSTOMERUPGRADES

AS

SELECT C.CUST\_FNAME || ' , ' || C.CUST\_SNAME CUSTOMER, K.KAYAK\_TYPE, U.UPGRADE\_WORK,

C.CUST\_CONTACT

FROM CUSTOMER C, KAYAKS K, UPGRADES U, KAYAK\_UPGRADES KU

WHERE C.CUST\_ID = KU.CUST\_ID

AND K.KAYAK\_ID = KU.KAYAK\_ID

AND U.UPGRADE\_ID = KU.UPGRADE\_ID

AND C.CUST\_ADDRESS LIKE '%Summer%';

SELECT \* FROM VWCUSTOMERUPGRADES;

---4--

CREATE VIEW vwCustomerRepairs AS

SELECT CONCAT(cust.cust\_sname, ', ', cust.cust\_fname) AS CUSTOMER,

kay.kayak\_type AS CLUB\_TYPE,

ku.upgrade\_work AS REPAIR\_WORK,

cust.cust\_contact AS CONTACT

FROM CUSTOMER cust

JOIN KAYAK\_UPGRADES ku ON cust.cust\_id = ku.cust\_id

JOIN KAYAKS kay ON ku.kayak\_id = kay.kayak\_id

WHERE cust.cust\_address LIKE '%Summer%';

