MODULE NAME:	MODULE CODE:
INFORMATION SYSTEMS 2C	INSY7213

ASSESSMENT TYPE: ASSIGNMENT 2 (PAPER ONLY)

TOTAL MARK ALLOCATION: 100

TIME ALLOWANCE: 15 HOURS

By submitting this assignment, you acknowledge that you have read and understood all the rules as per the terms in the registration contract, in particular the assignment and assessment rules in The IIE Assessment Strategy and Policy (IIE009), the intellectual integrity and plagiarism rules in the Intellectual Integrity and Property Rights Policy (IIE023), as well as any rules and regulations published in the student portal.

INSTRUCTIONS:

1. *The research component of the assessment must be typed as well as adequately and correctly referenced.*
2. *Where images are used for reference, they must be included in the submission in a separate document.*
3. ***The assessment must be submitted through Turnitin.***
4. ***No material may be copied from original sources, even if referenced correctly, unless it is a direct quote indicated with quotation marks. No more than 10% of the assignment may consist of direct quotes.***
5. ***Please ensure you attach an originality report to your assignment if required.***
6. ***Make a copy of your assignment before handing it in.***
7. *Consult the marking rubric to familiarise yourself with the assessment criteria to ensure you understand all the submission requirements.*
8. *This is an individual project unless stated otherwise in the brief.*
9. *Keep a clear copy of your project in the form of a digital back-up/a clear photocopy/a photograph.*
10. *Pay careful attention to the submission format and additional requirements as stated in the brief. Make sure your submission is neat and presented professionally.*
11. *All submissions must include a stamped and signed cover sheet, your name and student number.*
12. *Follow all instructions on the brief front page including the additional instructions added below.*

Assignment Instructions

1. For mark allocation, it is crucial that you refer to the Marking Rubrics at the end of each question. These rubrics provide a clear breakdown of how your work will be assessed.
2. *Do not submit AI-generated code as your own work. Submitting AI code without your analysis, synthesis, and critical thinking may be considered plagiarism or academic misconduct.*
3. *If you used AI tools in any part of your process (e.g., planning, proofreading, coding), it's advisable to disclose their use briefly in a footnote or comments section (e.g., "ChatGPT was consulted for initial topic brainstorming only.").*

Practical Assignment Outcomes

Learning Units 1 to 7 are covered in this assignment :

At the end of this assessment, students should be able to:

- *Compare the features of PL/SQL;*
- *Compare the components of the PL/SQL environment;*
- *Distinguish between the types of PL/SQL blocks;*
- *Explain how to output messages using PL/SQL.*
- *Construct PL/SQL Blocks;*
- *Design PL/SQL Queries that use Variables.*
- *Construct a PL/SQL query with a Cursor*
- *Explain how to deal with exceptions*

Case Study

Study the following case study and answer the questions that are based thereon.

CHARITY 4 AFRICA is a government initiative to allow individuals to donate their unwanted household items to a local charity store. All proceeds of the sale of the donations are provided to various charity organisations around the country. CHARITY 4 AFRICA has grown from strength to strength due to excellent service, a wide variety of donations, and a delivery service of large items, such as fridges and televisions. At present, all the data is stored in a flat file system. The charity at present is under pressure due to the increase in employees, customers, donors, donations, deliveries, returns and invoices, and requires a database system to manage their data.

You have been contracted to perform the design and implementation of a database for CHARITY 4 AFRICA. The charity store opened its doors to the public five years ago and requires a database to manage all areas of its business.

The data in flat files has been provided:

CUSTOMER

CUSTOMER_ID	FIRST_NAME	SURNAME	ADDRESS	CONTACT_NUMBER	EMAIL
11011	Jack	Smith	18 Water Rd	0877277521	jsmith@isat.com
11012	Pat	Hendricks	22 Water Rd	0863257857	ph@mcom.co.za
11013	Andre	Clark	101 Summer Lane	0834567891	aclark@mcom.co.za
11014	Kevin	Jones	55 Mountain way	0612547895	kj@isat.co.za
11015	Lucy	Williams	5 Main rd	0827238521	lw@mcal.co.za

EMPLOYEE

EMPLOYEE_ID	FIRST_NAME	SURNAME	CONTACT_NUMBER	ADDRESS	EMAIL
emp101	Jeff	Davis	0877277521	10 main road	jand@isat.com
emp102	Kevin	Marks	0837377522	18 water road	km@isat.com
emp103	Adanya	Andrews	0817117523	21 circle lane	aa@isat.com
emp104	Adebayo	Dryer	0797215244	1 sea road	aryer@isat.com
emp105	Xolani	Samson	0827122255	12 main road	xosam@isat.com

DONATOR

DONATOR_ID	FIRST_NAME	SURNAME	CONTACT_NUMBER	EMAIL
20111	Jeff	Watson	0827172250	jwatson@ymail.com
20112	Stephen	Jones	0837865670	joness@ymail.com
20113	James	Joe	0878978650	jj@isat.com
20114	Kelly	Ross	0826575650	kross@gsat.com
20115	Abraham	Clark	0797656430	aclark@ymail.com

DONATION

DONATION_ID	DONATOR_ID	DONATION	PRICE	DONATION_DATE
7111	20111	KIC Fridge	R 599	1 May 2024
7112	20112	Samsung 42inch LCD	R 1 299	3 May 2024
7113	20113	Sharp Microwave	R 1 599	3 May 2024
7114	20115	6 Seat Dining room table	R 799	5 May 2024
7115	20114	Lazyboy Sofa	R 1 199	7 May 2024
7116	20113	JVC Surround Sound System	R 179	9 May 2024

DELIVERY

DELIVERY_ID	DELIVERY_NOTES	DISPATCH_DATE	DELIVERY_DATE
511	Double packaging requested	10 May 2024	15 May 2024
512	Delivery to work address	12 May 2024	15 May 2024
513	Signature required	12 May 2024	17 May 2024
514	No notes	12 May 2024	15 May 2024
515	Birthday present wrapping required	18 May 2024	19 May 2024
516	Delivery to work address	20 May 2024	25 May 2024

RETURNS

RETURN_ID	RETURN_DATE	REASON	CUSTOMER_ID	DONATION_ID	EMPLOYEE_ID
ret001	25 May 2024	Customer not satisfied with product	11011	7116	emp101
ret002	25 May 2024	Product had broken section	11013	7114	emp103

INVOICE

INVOICE_NUM	CUSTOMER_ID	INVOICE_DATE	EMPLOYEE_ID	DONATION_ID	DELIVERY_ID
8111	11011	15 May 2024	emp103	7111	511
8112	11013	15 May 2024	emp101	7114	512
8113	11012	17 May 2024	emp101	7112	513
8114	11015	17 May 2024	emp102	7113	514
8115	11011	17 May 2024	emp102	7115	515
8116	11015	18 May 2024	emp103	7116	516

Refer to Appendix A for all Marking rubrics.

Question 1 **(Marks: 10)**

You will need to create the above tables to complete th***is Assignment,*** as well as populate them using SQL Developer or SQL*Plus.

Requirement	*Mark*
Tables created successfully	***5***
Tables populated successfully	***5***
TOTAL	***10***

Question 2 **(Marks: 10)**

CHARITY 4 AFRICA requires a report containing the combined customer name, employee ID, delivery notes, donation purchased and the invoice number. Create an SQL query to generate the required report. In your query, only display the results that have any invoice date after 16 May 2024.

Sample Results

CUSTOMER	EMPLOYEE_ID	DELIVERY_NOTES	DONATION	INVOICE_NUM	INVOICE_DATE
Pat, Hendricks	emp101	Signature required	Samsung 42inch LCD	8113	17/MAY/24
Lucy, Williams	emp102	No notes	Sharp Microwave	8114	17/MAY/24
Jack, Smith	emp102	Birthday present wrapping required	Lazyboy Sofa	8115	17/MAY/24
Lucy, Williams	emp103	Delivery to work address	JVC Surround Sound System	8116	18/MAY/24

Question 3 **(Marks: 10)**

Management of CHARITY 4 AFRICA would like you to add a new table called Funding that at present will not have any referential integrity. The organisation would like an automatically generated unique id every time a record is inserted into the new table. The attributes for the table for now are funding_id, funder, and funding amount.

Complete the following:

1. Create the new table.
2. Implement a solution to automatically generate the unique ids with every new insert.
3. Provide an example of the insert statement.
4. Add a brief comment to justify your solution.

Question 4 **(Marks: 10)**

Create a PL/SQL query to display the combined customer name, donation purchased, donation price and the reason why the donation was returned.

Sample Output:

```
CUSTOMER:                 Jack, Smith
DONATION PURCHASED:       JVC Surround Sound System
PRICE:                    179
RETURN REASON:            Customer not satisfied with product
-----------------------------------
CUSTOMER:                 Andre, Clark
DONATION PURCHASED:       6 Seat Dining room table
PRICE:                    799
RETURN REASON:            Product had broken section
-----------------------------------


PL/SQL procedure successfully completed.
```

Question 5 **(Marks: 15)**

Create a PL/SQL query to display the customer's name, employee name, donation, dispatch date and the delivery date. In your query, also display the number of days between the dispatch date and the delivery date in days for customer 11011.

Sample Output:

```
CUSTOMER:        J.Smith
EMPLOYEE:        A.Andrews
DONATION:        KIC Fridge
DISPATCH DATE:   15/MAY/24
DELIVERY DATE:   10/MAY/24
DAYS TO DELIVERY 5
-----------------------------------
CUSTOMER:        J.Smith
EMPLOYEE:        K.Marks
DONATION:        Lazyboy Sofa
DISPATCH DATE:   19/MAY/24
DELIVERY DATE:   18/MAY/24
DAYS TO DELIVERY 1
-----------------------------------
```

Question 6 **(Marks: 15)**

Create a report to display the combined customer name and total amount spent by each customer on the purchases. In your solution determine the customer rating as follows: If the total amount spend is greater than or equal to R 1 500, the customer receives a 3-star rating.

Sample Output:

```
PL/SQL procedure successfully completed.

FIRST NAME:  Jack
SURNAME:     Smith
AMOUNT:      R 1798 (***)
--------------------------------------------------
FIRST NAME:  Pat
SURNAME:     Hendricks
AMOUNT:      R 1299
--------------------------------------------------
FIRST NAME:  Lucy
SURNAME:     Williams
AMOUNT:      R 1778 (***)
--------------------------------------------------
FIRST NAME:  Andre
SURNAME:     Clark
AMOUNT:      R 799
--------------------------------------------------
```

Question 7 **(Marks: 20)**

Provide relevant code examples, with code comments based on this case study, of how to implement the following:

Note: Provide your own code examples, with comments, and the achieved output for each question. The reuse of codes from the previous questions (Q.1 -Q.6) will be awarded a 0.

Q.7.1	%TYPE attribute;	(5)
Q.7.2	%ROWTYPE attribute	(5)
Q.7.3	User defined exception	(10)

Question 8 **(Marks: 10)**

In Question 6, you created a PL/SQL report to determine customer ratings. Using this background, you are required to create a similar report using SQL and CASE statements only, to display the combined customer name and total amount spent by each customer on the purchases. In your solution, determine the customer ratings as follows:

- If the total amount spent is greater than or equal to R 1 500, the customer receives a 3-star rating.
- If the total amount spent is between R1000 and R1400, then the customer receives a 2-star rating.
- Otherwise, the customer receives 1 star rating.

Sample Output:

FIRST_NAME	SURNAME	AMOUNT	CUSTOMER_RATING
Jack	Smith	1798	***
Pat	Hendricks	1299	**
Lucy	Williams	1778	***
Andre	Clark	799	*

Appendix A

Assessment Sheet (Marking Rubric)

MODULE NAME:	**MODULE CODE:**
INFORMATION SYSTEMS 2C	**INSY7213**
STUDENT NAME:	
STUDENT NUMBER:	

Question 1 Mark Allocation	**Levels of Achievement**				**Feedback**
	Excellent	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
Database and Tables	**3**	**2**	**1**	**0**	
	New database schema created; Tables created correctly.	New database schema created; At least half of the tables created correctly.	New database schema created; Less than half of required tables created correctly.	**Not provided.**	
Data Inserts	**6 – 7**	**4 – 5**	**1 – 3**	**0**	
	Insert statements per table inserted correctly.	Insert statements for at least half of the tables inserted correctly.	Insert statements provided; Less than half of tables inserted correctly.	**Not provided.**	

Question 2 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
SQL Query	**8-10**	**6**	**1-4**	**0**	
	Correct select used: Correct tables, attributes and conditions used. Correct output achieved.	Minor changes required.	Major changes required.	Not provided.	

Question 3 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
Table	**3 -4**	**2**	**1**	**0**	
	Funding table created correctly.	Minor changes required.	Major changes required.	Not provided.	
Solution & Output	**5 – 6**	**3 - 4**	**1 – 2**	**0**	
	Correct implementation of a solution to add a unique funding id with every new insert (4). Justification comment correctly provided (1). Output achieved (1)	Minor changes required.	Major changes required.	Not provided.	

Question 4 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
PL/SQL Statement	**4 – 5**	**3**	**1 – 2**	**0**	
	Correct PL/SQL statements and structure implemented	PL/SQL statements and structure implemented; minor changes required.	PL/SQL statements and structure implemented; major changes required.	**Not provided.**	
SQL Statements	**3**	**2**	**1**	**0**	
	Correct SQL statements and tables used.	SQL statements and tables used; minor changes required.	SQL statements and tables used; major changes required.		
Output	**2**	**1**	**½**	**0**	
	Correct output achieved.	Minor changes required	major changes required.		

Question 5 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
Declarations	**5 – 6**	**3 –4**	**1 – 2**	**0**	
	Correct declare statements with variables used.	Minor changes required.	Major changes required.	Not provided.	
Tables & PL/SQL Statement	**6 -7**	**4 - 5**	**1 - 3**	**0**	
	Correct tables used; joined correctly; Correct PL/SQL statements and structure used	minor changes required.	major changes required.	Not provided	
Output	**2**	**1**	**½**	**0**	
	Output achieved.	minor changes required.	major changes required.	Not provided	

Question 6 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
Declarations	**4 -5**	**3**	**1 - 2**	**0**	
	Correct declare statements with variables used.	**Minor changes required.**	**Major changes required.**	**Not provided.**	
Tables & PL/SQL Statement	**6 -8**	**4 -5**	**1 - 3**	**0**	
	Correct tables used; joined correctly; Correct PL/SQL statements and structure used	minor changes required.	major changes required.	Not provided	
Decision Statements	4 -5	2	1 - 2	**0**	
	Correct decision statement used; Customer rating implemented correctly.	Minor changes required	Major changes required	Not provided	
Output	**2**	**1**	**½**	**0**	
	Output achieved.	minor changes required.	major changes required.	Not provided	

Question 7 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
Q.7.1: %TYPE attribute	**4 -5**	**3**	**1 - 2**	**0**	
	Correct implementation (3); Relevant comments to explain implementation provided correctly (1); Code runs successfully (1).	**Minor changes required.**	Major changes required.	Not provided.	
Q.7.2: %TYPE attribute	**4 -5**	**2**	**1 - 2**	**0**	
	Correct implementation (3); Relevant comments to explain implementation provided correctly (1); Code runs successfully (1).	minor changes required.	major changes required.	Not provided	
Q.7.3: User defined exceptions	**8 – 10**	**5 - 7**	**1 - 4**	**0**	
	Correct implementation (6); Relevant comments to explain implementation provided correctly (2); Code runs successfully (2)	Minor changes required	Major changes required	Not provided	

Question 8 Mark Allocation	Levels of Achievement				Feedback
	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
SQL Query	**4**	**3**	**1-2**	**0**	
	Correct use of select statements (2): Correct tables, attributes and conditions used (2).	Minor changes required.	Major changes required.	Not provided.	
CASE statement	**4**	**3**	**1-2**	**0**	
	CASE statement correctly applied.	Minor changes required.	Major changes required.	Not provided.	
Output	**2**	**1**	**½**	**0**	
	Correct output achieved.	Minor changes required.	Major changes required.	Not provided.	