

SET09107: Advanced Database Systems

2017/2018 Semester 2 – Coursework

Scenario

A bank has several branches in the UK. It needs a database to store information about its local branches. Each branch is identified by a unique branch code, an address (street, city, post code), and a phone number. The customer accounts at each branch are also recorded.

Each customer account is identified by a unique account number, an account type (current or savings), and a balance. Each account has an interest rate (interest rate can be determined by yourself - any reasonable one will be fine). An account is also associated with exactly one branch. The date when the account is opened is recorded as well. An account must be classified as either a current or a savings account (but not both). A current account also has a limit of free overdraft (overdraft can be determined by yourself - any reasonable one will be fine). The free overdraft limit is set at the opening of an account.

Data about customers and employees is also recorded. All customers and employees have an associated National Insurance number (a tax payer's unique identification number), address (street, city, post code) and phone numbers (home number and mobile numbers). An employee cannot be a customer at the same branch where he/she works. An employee has a job position (Head, Manager, Project Leader, Accountant, Cashier) and a salary, and works for exactly one branch. The date that the employee joined the bank is also recorded. Every employee has a supervisor at the same branch, except the head of the branch. The head of the branch is the only person who is not supervised by anyone at the same branch. A customer may have multiple accounts with the bank, and an account may be owned by multiple customers as a joint account.

The database:

This database is designed with the following relations, where the primary keys are underlined and foreign keys are in italic:

Branch(bID, street, city, p_code, bPhone)

Account(accNum, accType, balance, *bID*, inRate, limitOfFreeOD, openDate)

Employee(empID, street, city, postCode, title, firstName, surName, empHomePhone, empMobile1, empMobile2, *supervisorID*, position, salary, niNum, *bID*, joinDate)

SET09107: Advanced Database Systems

Customer(custID, street, city, postCode, title, firstName, surName, custHomePhone, custMobile1, custMobile2, niNum)

CustomerAccount(custID, accNum)

Sample data:

Branch

901	Market	Edinburgh	EH1 5AB	01311235560
908	Bridge	Glasgow	G18 1QQ	01413214556

Account

1001	current	820.50	901	0.005	800	01-May-11
1010	savings	3122.20	901	0.02		08-Mar-10
8002	current	200	908	0.005	100	05-May-09

Employee

101	Dart	Edinburgh	EH105TT	Mrs	Alison	Smith	01312125555	07705623443 07907812345		Head	50000	NI001	901	01-Feb-08
105	New	Edinburgh	EH24AB	Mr	John	William	01312031990	07902314551 07701234567	101	Manager	40000	NI010	901	04-Mar-09
108	Old	Edinburgh	EH94BB	Mr	Mark	Slack	01312102211		105	Accountant	30000	NI120	901	01-Feb-12
804	Adam	Edinburgh	EH16EA	Mr	Jack	Smith	01311112223	0781209890	801	Leader	35000	NI810	908	05-Feb-14

Customer

1002	Adam	Edinburgh	EH1 6EA	Mr	Jack	Smith	01311112223	0781209890 0771234567	NI810
1003	Adam	Edinburgh	EH1 6EA	Ms	Anna	Smith	01311112223	0770111222	NI010
1098	New	Edinburgh	EH2 8XN	Mr	Liam	Bain	01314425567		NI034

CustomerAccount

1002	1001
1002	1010
1003	1010
1098	8002

Your tasks:

- 1 Draw an ER diagram corresponding to the relational database schema and the scenario. **(5 marks)**
- 2 Re-design the database to capture more of the semantics of the application making use of object-relational features as extensively as possible while still retaining the semantics of the application. Provide a **critical** discussion on the rationale for your object-relational design justifying the design you adopted with reasons. The discussion should include alternative possible object-relational representations you considered and why they were rejected (only include those if you think they demonstrate a deeper understanding of the problem involved). **(40 marks)**

- 3 Implement the database according to your design at task 2 and populate the tables with test data, 20 rows at least for each table. The data you inserted should be sufficient to demonstrate your object-relational design decisions and unambiguously answer the queries below. All SQL statements for creating and populating the database should be included. **(10 marks)**
- 4 Provide SQL statements and answers (outputs) to the following queries on the database you re-designed and implemented. Comments are expected. Output should be formatted. Data/information displayed should be in values, not in types, e.g., John Smith, not NAME('John', 'Smith').
- a. Find employees whose first name includes the string “on” and live in Edinburgh, displaying their full names. **(3 marks)**
 - b. Find the number of saving accounts at each branch, displaying the number and branch’s address. **(3 marks)**
 - c. At each branch, find customers who have the highest balance in their savings account, displaying their names, the balance, the branch ID and the free overdraft limit in their current accounts. **(3 marks)**
 - d. Find employees who are supervised by a manager and have accounts in the bank, displaying the branch address that the employee works in and the branch address that the account is opened with. **(3 marks)**
 - e. At each branch, find customers who have the highest free overdraft limit in all current accounts that are joint accounts, displaying the branch’s ID, the customer’s full names, the free overdraft limit in his/her current account. **(5 marks)**
 - f. Find customers who have more than one mobile, and at least one of the numbers starts with 0770, displaying the customer’s full name and mobile numbers. **COLLECTIONS** must be used. **(5 marks)**
 - g. Find the number of employees who are supervised by Mrs Smith, who is supervised by Mr Jones. **REFERENCES** must be used. **(5 marks)**
 - h. Award employees at the end of a year: gold medals for employees who have been working at the bank for more than 12 years and supervised more than 6 staff; silver medals for employees who have been working at the bank for more than 8 years and supervised more than 3 staff; bronze medals for employees who have been working at the bank for more than 4 years, displaying their names and Medal awarded (only displaying those who have been awarded). **METHODS** must be used. **(8 marks)**
- [35 marks in total]**
- 5 Critically discuss the advantages and disadvantages of the object-relational model against the relational model, based on the designs and implementations for the proposed bank database. **(8 marks)**
- 6 Write a sequence of drop statements so that if executed one after the other all of the tables and types implemented for the coursework will be removed. **(2 marks)**

Deliverables:

- A hard copy of answers to tasks **1, 2, 4, 5 and 6**. Answers to task 4 should include all SQL scripts and outputs.
- Your answers to task 3, 4 **and 6**, including all your SQL scripts, test data in tables and outputs, should be saved in a text file first, **well formatted** and then zipped into a file called set09107cw_<your matric number> and uploaded to Moodle **on the same day of your submission**, as per instructions. For example, if your matriculation number is 40012345, your zipped file should be named as set09107cw_40012345.

Demonstration

Your work to task 4 should be demonstrated in the practical sessions in **week 11 and 12**, from 9:00 to 11:00 on Friday, the **23rd March and 13th April**. **No marks to this task will be awarded if you don't do the demonstration.**

Hand in: The hard copies are required to be submitted to the Computing School Office (Room C34 Merchiston), with a coursework coversheet. Your matriculation card is required when you hand in your work, and the matriculation number must be marked on the submitted hard copy.

Deadline: 16:00, 16th March 2018

Notes:

This coursework contributes 60% to the overall module assessment.

Collaboration and Plagiarism

This is an individual assessment. The work submitted should be entirely your own and will be checked against all other submissions by TurnitinUK.

Marking Scheme

Marking (100%)

Item	Mark	Notes
Task1 (5)		
Diagram		
Task2 (40)		
Design (25)		
Rational (10)		
Alternative (5)		
Task3 (10)		
Consistency (5)		
Data (3)		
Format (2)		
Task4(35)		
a (3)		
b (3)		
c (3)		
d(3)		
e(5)		
f(5)		
g(5)		
h(8)		
Task 5 (8)		
General Discuss (4)		
Based on your design and Implementation (4)		
Task 6 (2)		