

Module Code	Examiner	Email of Examiner	Tel
CPT103	Jianjun Chen		

**2<sup>st</sup> SEMESTER 2021/22 Individual Coursework (Part 1)**

**Undergraduate – Year 2**

**Introduction to Databases**

---

**INSTRUCTIONS TO CANDIDATES**

- 1、 This coursework (course work 3) accounts for 70% of the final marks and contains two parts. This is part 1.
- 2、 This part worth 60% of the total marks of the coursework 3.
- 3、 The report and the summary should be written in English.
- 4、 Please remember to check the updates and common Q&A on the LearningMall about this coursework. This document might be updated one or two times if certain specifications are not clear enough. All updates will appear here or be highlighted in this document. Remember to check emails about any updates.

<b>Release Date</b>	<b>Announced on LearningMall</b>
<b>Due date</b>	Follow the LearningMall assignment deadline
<b>Deliverables</b>	D1. A written report. D2. A SQL script file.
<b>Format</b>	Please submit 2 files to the LearningMall (do not compress these files into zip files): <ul style="list-style-type: none"> <li>- A written report, no more than 10 pages, called “report.pdf”, Containing your ER diagram and necessary explanations to your design.</li> <li>- A script called “script.sql”, which contains all statements for creating the tables.</li> </ul>
<b>Submission</b>	LearningMall
<b>Mark</b>	This part worth 60% of the total marks of the coursework 3

## Lukewarm Live Steaming Platform (Part 1)

### Introduction

Cockatiel Entertainment is a gaming platform company located at the east coast of Lukewarm Kingdom. Recently, the company wants to extend its business to support video game streaming. They plan to build a platform for this purpose and has already given it a name: CockatielTV. You have been appointed to design a database based on the requirements of the CockatielTV and provide detailed documentation about its design and usage. You will need to apply all the knowledge throughout the course. Only database design is involved, no C/Java programming is needed for this project. There are two parts of this coursework, this assignment is the first part.



Image from <https://petwiseaquatics.co.uk/cockatiel/>

### Task 1: Normalisation

To help you develop the CockatielTV database, the manager provided you with an example live streaming record (see the next page). You need to identify functional dependencies of the attributes in this list and normalise it to the third normal form (3NF). Please follow the format provided in the report template and write down the steps. This part is to test your understanding of normalisation, you are not allowed to add any extra attributes or remove any existing attributes. Once you have done the normalisation, please work out all foreign keys and draw the corresponding ER diagram. Then, proceed to finish the rest part of the coursework and complete the report.

The example live stream sessions record is shown below. Assume that the current time is 2022-03-21 13:21.

stream_id	streamer_id	streamer_name	Stream_title	start_time	end_time	game_id	game_title	developer_name
900	6511123, 8421921	Taylor, Wendy	Dungeon exploration time with Taylor&Wendy	2022-2-10 17:00	2022-2-10 19:00	7701	Dungeon Fight XD	Voodoo Gaming
201	6511123	Taylor	Friday Cyberprank Time!	2021-11-26 19:00	2021-11-26 22:00	6602	Cyberprank	Projekt Troll
392	8421921	Wendy	Casual afternoon with Wendy	2022-2-26 13:00	2022-2-26 17:00	6603	The Watcher 3: Mild Hunt	Projekt Troll
851	6511123, 8421921	Taylor, Wendy	Holiday Party! Yay!	2021-1-21 11:30	2021-1-21 15:00	6602, 7701	Cyberprank, Dungeon Fight XD	Projekt Troll, Voodoo Gaming
1001	7577187	George	Let's explore the nether today	2022-03-21 12:40		5501	Ninecraft	Mahjong Games

## Task 2: Full Database Design Using ER Modelling

This part is based on the design you obtained from task 1. Please read the requirements below carefully and add new tables or modify existing tables accordingly. You are required to produce an ER diagram for your design and write detailed descriptions for your tables in your report. Details are given in the report template.

According to the company's plan, all games that will appear in live stream sessions must be recorded in the database first. Each game has a game title, its release date, age rating and its developer. Games will also be given unique IDs in case game names collide. The age rating refers to the minimum age that can play this game. In this coursework, we will use PEGI age labels for the games stored in the database. Please learn about PEGI age labels by yourselves and design your database according to it.

Live stream sessions can start and finish at any time. CockatielTV allows multiple streamers to share a same live stream session and play several games in the session. Past live stream recordings are also available to users, so make sure the information of all stream sessions is recorded correctly in the database. A streamer can plan a stream before it even happens. All live streams also obey the age ratings of games. For example, an 8-years-old user cannot watch any live streams or stream records that involve games with age ratings higher than 8.

A user account is needed to watch live streams or streaming any games. To register one account, a person needs to provide his name, id card/passport number, date of birth, email address and then decide a unique nickname and a password. A user can do his own live streaming only after owning his user account for more than 180 days.

The income of streamers comes from 2 sources: direct donations from audiences and the total amount of time (in minutes) people spent on his live streams. Direct donations come in the form of virtual gifts. There are currently 6 virtual gifts available in CockatielTV: Light Feather, Silver Feather, Gold Feather, Cockatiel Wink, Cockatiel Song and Cockatiel Crown. The company plans to add more gifts in the future, so your database design for this part should also support future extensions.

A special virtual currency called Parrocoin is used to purchase gifts throughout the platform. Parrocoins are obtained by paying real-world money to the platform. After paying the money, Parrocoins will be added to the user's balance. The table below lists the prices of the 6 virtual gifts. These prices may also be adjusted in the future by the company.

Light Feather	100 Parrocoins
Silver Feather	150 Parrocoins
Gold Feather	200 Parrocoins
Cockatiel Wink	400 Parrocoins
Cockatiel Song	500 Parrocoins
Cockatiel Crown	1000 Parrocoins

The second source of incomes for streamers is calculated based on how much time people spend on his live streams. Note that if a user watch the recordings of a live stream, his watching time should not contribute to the total amount of time of this live session. If an user join and leave

multiple times during the live stream, the total number of minutes he has watched the video should be the sum of these multiple watch periods. Your database design must support this way of time calculation precisely.

## Example Database Operations

This section lists some additional operations that should be supported by your database. Please try to work out the SQL queries for these tasks. You are not required to include them in the report. But failing to achieve these operations indicates some major flaws with your database design. As a result, if you find out that your design cannot support these operations, go back and modify the database design:

1. List all live stream sessions that a user nicknamed “Jason” can watch, the result should satisfy the age rating restrictions.
2. List all future live streams about the game “Ninecraft”. Assume that the current time is 2022-2-20 19:30.
3. List all donations that happened during live streaming sessions with title “Peekaboo!”.
4. Find out the total number of hours users have spent on watching the live streaming session called “Peekaboo!”, assume that there’s only one stream with this title.
5. Find out the average number of minutes “Jason” spend on game streams (including recordings).
6. List all stream sessions that involves more than one streamer.

## Your Tasks

### D1. Report

You are required to write a detailed report, called “report.pdf”, explaining your database design for the CGS. **Please do not submit word documents directly**. Export the word document to PDF first (Word, WPS and Libreoffice all support this). **5 marks** will be deducted if the file format is not PDF. **The file should be a valid PDF file that can be opened by Acrobat Reader or similar software.**

For the database design, you need to make an ER diagram that fits **one single page**. Please do not split the diagram into several pages. **5 marks** will be deducted if the ER diagram doesn't fit one page.

Please strictly follow the report template and answer all the questions in it. The ER diagram only constitutes a small part of your marks. The majority of your marks come from the quality of your discussions and ideas.

The final design must be in the third normal form (3NF) and has no M:M and 1:1 relationships. You can make some assumptions about the data if they are not specified in the requirements. For example, you can assume address occupies up to 200 characters, or people's names can be up to 30 characters long. You need to indicate these decisions in the report.

### D2. Script

A script called “script.sql”, which contains all CREATE TABLE statements that create the tables. **Your SQL script and your ER diagram must match (attributes = columns, entities = tables, 1:m relationships = foreign keys).**

## University Policy on Late Submissions

If you submit coursework after the deadline, you will be penalized:

- **5%** of the total marks available for the assessment will be deducted from the assessment mark for **each working day** after the submission deadline, up to a maximum of 25%;
- Coursework received **more than five working days** after the submission deadline will receive a mark of **zero**.

## University Policy on Academic Integrity

The University aims to foster a learning environment which produces students who embrace academic integrity, understand that they must produce their own work, are able to acknowledge explicitly any material that has been included from other sources or legitimate collaboration, and to present their own findings, conclusions or data based on appropriate and ethical practice.

The University will support you to understand the standards of academic integrity, while you are responsible for learning and upholding professional standards of research, writing, assessment, and ethics in your area of study. Violation of academic integrity comes in many forms, including but not limited to the following:

- improper citation or referencing;
- unauthorised collaboration with another person in the preparation and production of a submitted work;
- copying directly from other persons without their knowledge as your own work;
- submitting all or part of the same academic work for two or more modules without permission;
- consciously representing another's work or concept as your own without proper acknowledgment and citation of the sources;
- altering data obtained by legitimate means or making up a portion or whole set of data and reporting them in your own assignment;
- requesting another party to prepare all or part of an assignment (with or without payment) on your behalf.

Any violation of academic integrity is a serious offence and is therefore subject to an appropriate penalty. According to the individual case and the seriousness of the offence, penalties applied will vary and may include one or a combination of the following:

- a written warning;
- a mark penalty or a zero mark for the assessment;
- a zero mark for the module;
- a note on student's records;
- suspension of studies;
- termination of studies.

In addition to the respective penalty imposed, you may also be given feedback on how to avoid further offence in future work.