

Data Analysis and Visualisation for a Online Retail Business

4BUIS014C Business Computing – Coursework (2024/25)	
Module leader	Abarnah Kirupananda
Unit	Coursework
Weighting:	70%
Qualifying mark	30%
Description	The module's second assessment is an individual coursework; students will be given a scenario that will require the student to create a small-scale application that will involve the creation of a number of data stores/repositories and the manipulation of data stored in these repositories; the output will be a technical report that will discuss their methodology and outputs.
Learning Outcomes Covered in this Assignment:	This assignment contributes towards the following Learning Outcomes (LOs): LO1 Produce a detailed design to create small scale business applications to retrieve and manipulate data stored in a data repository (database, spreadsheet, etc); LO2 Utilise a programming language, software packages and tools to create small scale business applications to retrieve and manipulate data stored in a data repository (database, spreadsheet, etc); LO3 Utilise a programming language, software packages and tools to extract from, manipulate and load to data stored in different repositories; LO4 Articulate the benefits of data visualisation and create primitive data visualisations;
Handed Out:	3 rd April 2025
Due Date	5 th May 2025, Submission by 13:00
Expected deliverables	Submit on Blackboard a single file containing the required documentation (either in docx or pdf format). All implemented codes should be included in your documentation together with the results/analysis. You must also submit your supporting files (coding files) in the supporting documents link.
Method of Submission:	Electronic submission on BB via a provided link close to the submission time.
Type of Feedback and Due Date:	Feedback will be provided on BB, within 3 weeks from the due date

Assessment regulations

Refer to section 4 of the "How you study" guide for undergraduate students for a clarification of how you are assessed, penalties and late submissions, what constitutes plagiarism etc.

Penalty for Late Submission

If you submit your coursework late but within 24 hours or one working day of the specified deadline, 10 marks will be deducted from the final mark, as a penalty for late submission, except for work which obtains a mark in the range 40 – 49%, in which case the mark will be capped at the pass mark (40%). If you submit your coursework more than 24 hours or more than one working day after the specified deadline you will be given a mark of zero for the work in question unless a claim of Mitigating Circumstances has been submitted and accepted as valid.

It is recognised that on occasion, illness or a personal crisis can mean that you fail to submit a piece of work on time. In such cases you must inform the Campus Office in writing on a mitigating circumstances form, giving the reason for your late or non-submission. You must provide relevant documentary evidence with the form. This information will be reported to the relevant Assessment Board that will decide whether the mark of zero shall stand. For more detailed information regarding University Assessment Regulations, please refer to the following website:<http://www.westminster.ac.uk/study/current-students/resources/academic-regulations>.

Objectives

1. Understand and analyse a real-world dataset.
2. Create a modular Python application to manage and analyse data.
3. Develop visualisations to communicate business insights effectively.
4. Understand and utilise structured programming techniques through packages and modules.

Purpose

This coursework is designed for you to develop a small-scale application to create a database, manipulate data, and visualize insights. The coursework requires you to create a Python-based package with modules to analyse customer transaction data for an online retail business using the **Online Sales Dataset** available on BB as a zipped folder containing 3 csv files. Outputs will include a technical report discussing methodology and findings.

The aim of this assessment to prepare you to:

- Develop an SQLite database from the given CSV files.
- Perform meaningful data analysis through SQL queries and Python functions.
- Create visualisations to communicate insights effectively.
- Demonstrate the ability to build a Python package with modular code for reusability.

The dataset will help simulate real-world scenarios, such as identifying sales trends, product popularity, and customer behaviours, essential for business decision-making.

Dataset Description

You are provided with three CSV files:

1. **customers.csv:** Contains customer IDs and countries.
Columns: CustomerID, Country.
2. **products.csv:** Contains product codes, descriptions, and unit prices.
Columns: StockCode, Description, UnitPrice.
3. **transactions.csv:** Contains online transaction details, including total purchase amounts in GBP.
Columns: InvoiceNo, CustomerID, StockCode, Quantity, InvoiceDate, UnitPrice, TotalAmount_GBP.

Tasks

Task 1: Create a Module for Database Setup and Population

[25 Marks]

- **Module Name:** “**OnlineDB**” (contained in the package **OnlineSales**).
- The module “**OnlineDB**” should include the followings:
 1. Define functions to:
 - Create an SQLite database (OnlineDB.db) with three tables: Customers, Products, Transactions.
 - Populate the tables in the database using data from the CSV files.
 - Handle missing or invalid data by excluding incomplete rows.
 2. Ensure data integrity and proper indexing in the database schema.
- **Deliverables:**
 - Create **OnlineDB** module.
 - Justify and document the functions in the module.
 - Ensure correct implementation and execution of the module.

Task 2: Create a Module for Data Queries

[25 Marks]

- **Module Name:** “**QueryData**” (contained in the package **OnlineSales**).
- The module “**QueryData**” should include the followings:
 1. Define functions to execute SQL queries answering the following questions:
 - What is the total quantity sold for each product monthly?
 - What are the sales trends for each product?
 - Who are the top 5 customers by total sales?
 - What is the revenue contribution by each product?
 2. Implement the functions using SQL queries and return the results as pandas DataFrames.
- **Deliverables:**
 - Create **QueryData** module.
 - Justify and document the query functions.
 - Ensure correct implementation and execution of the queries.

Task 3: Create a Module for Data Visualisation

[25 Marks]

- **Module Name:** “**VizData**” (contained in the package **OnlineSales**).
- The module “**VizData**” should include the followings:
 1. Define functions to generate visualisations for the queries:
 - **Bar chart:** Total quantity sold for each product.
 - **Line chart:** Product sales trends monthly.
 - **Horizontal bar chart:** Top 5 customers by sales.
 - **Pie chart:** Revenue contribution by products.
 2. Use libraries like Matplotlib and Seaborn for visualisation.
- **Deliverables:**
 - Create **VizData** module.
 - Justify and document visualisation functions.
 - Ensure correct implementation and execution of the visualisations.

Task 4: Integrate and Execute Modules in a Main File

[25 Marks]

- **File Name:** **Main.ipynb** (contained in the package **OnlineSales**).
- The file “**Main.ipynb**” should include the following:
 1. Import and use the created modules from the package **OnlineSales** (from **task 1, task 2, and task 3**).
 2. Execute functions to:
 - Create and populate the database.
 - Perform data queries.
 - Generate and display results in the form of visualisations to develop relevant insights.
 3. Provide a brief explanation on the insights derived.
- **Deliverables:**
 - Integration and execution in **Main.ipynb**.
 - Presentation of results and insights.

Submission Instructions:

- 1.** A technical report (**PDF or DOCX**), including:
 - Copied and pasted codes in a text format, followed by the relevant justification/purpose of the required functions to populate the database, execute queries, and create visualisations.
 - Results and insights derived from the analysis.
 - Visualisations with interpretations.

- 2.** Submit a single zipped folder as a **Supporting Document** containing Python package **OnlineSales** with modules:
 - **OnlineDB.py**
 - **QueryData.py**
 - **VizData.py**
 - **Main.ipynb** Jupyter notebook file.

Conditions:

- The report must express your own conclusions and findings.
- The overall size of your report is restricted to 20 pages in total.
- Your report should be referenced if necessary.

Assessment Criteria:

- **Database Creation (20%):**
 - Completeness and correctness of **OnlineDB** implementation in the package **OnlineSales**.

- **Data Queries (20%):**
 - Completeness and correctness of **QueryData** in the package **OnlineSales**.

- **Visualisations (20%):**
 - Quality and relevance of **VizData** visualisations in the package **OnlineSales**.

- **Integration and Execution (20%):**
 - Overall code quality and seamless integration in the **Main.ipynb** file.

- **Technical Report (20%):**
 - Clarity, justification, and interpretation of results.