INTERNATIONAL QUALIFICATIONS

AND ASSESSMENT CENTRE (IQAC)

«Approved»

2025 y. «16»

Programme	**LEVEL 4 EXTENDED DIPLOMA IN DATA SCIENCE**		
Unit Number/ Unit Title	UNIT 1. PROGRAMMING WITH PYTHON		
Cohort Code:	L04PWP-U1		
Unit Level	Level 4		
Total GLH	Total qualification time 200/ Total Guided learning hours 90/ Self-guided learning hours 110		
Credits	20 CATS/ 10 ECTS		
Lecturer			
Start Date		**End Date**	

Unit Aims	This module aims to provide students with hands-on programming experience using Python, the leading language in the data science ecosystem. The module develops practical skills in data handling, analysis, visualization, and automation — enabling students to build foundational tools and workflows essential in data science projects
Differentiation Strategies *(e.g. planned activities or support for individual learners according to their needs)*	The total number of students to be in the lesson is approximately 20. This is a multicultural group of students predominantly between the ages of 24 – 45, with numerous ethnic, gender, and creed background. These are UK academic level 5 students; hence it is assumed that they have practical, theoretical, or technological knowledge and understanding of a subject or field of work to find ways forward in broadly defined, complex contexts. These students must be able to generate information, evaluate, synthesise the use information from a variety of sources. Various approaches to addressing the various identified students needs will be adopted throughout the lesson. Such will include:- 1. Progressive tasks 2. Digital resources 3. Verbal support 4. Variable outcomes

Learning Outcome		Assessment Criteria
LO1. 1. Understand core programming concepts in Python.	7. Flexible pace learning	1.1.Define and apply data types, variables, loops, and conditional statements.
Equality & Diversity	Variety of teaching techniques will be employed to ensure that the needs of each individual learner are met.	1.2 Implement functions, error handling, and modular code structures.
Safeguarding & Prevent	Safeguarding policies and the Prevent duty are strictly observed to ensure the safety, well-being, and inclusivity of all students and staff.	
LO2. 2. Manipulate and analyse datasets using Python.		2.1 Load, clean, and transform datasets using libraries such as Pandas.
Health & Safety	SIRM H&S policies will be maintained.	
	Teaching and Learning Materials	**Virtual Learning Environment**
LO3. 3. Create data visualizations using Python tools.	• VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media. • McKinney, W. (2022). Python for Data Analysis. O'Reilly Media.	2.2 Perform exploratory data analysis and summary statistics. 3.1 Develop charts and plots using Matplotlib and Seaborn. 3.2 Interpret and present visual data effectively.
Learning Resources		College Moodle/Zoom
LO4. 4. Build and document Python scripts for reproducible data workflows.	• Downey, A. (2015). Think Python: How to Think Like a Computer Scientist. O'Reilly Media. • Lutz, M. (2013). Learning Python. O'Reilly Media.	4.1 Write well-structured and reusable code. 4.2 Use commenting and markdown documentation to enhance clarity.
LO5. 5. Apply Python programming in a mini data science project.		5.1 Solve a real-world data problem using Python. 5.2 Present and explain the solution with code and visualization.
Enrichment Opportunities	Virtual learning platform.	

No	Learning Outcome / Topic	Learning and Teaching Activities	Which assessment criteria does the session relate to?	Day/month/year/ signature
1.	**Python Basics & Jupyter Notebooks**	**Python Basics & Jupyter Notebooks** Variables, data types (`int`, `str`, `bool`), and operators	LO1: Core Python Programming	
2.	**Control Flow**	**Control Flow** `if-elif-else` statements, logical operators	LO1: Core Python Programming	
3.	**Loops & Iterations**	**Loops & Iterations** `for`/`while` loops, `break`/`continue`	LO1: Core Python Programming	
4.	**Functions & Modularity**	**Functions & Modularity** Defining functions, `return` vs `print`, scope (local/global)	LO1: Core Python Programming	
5.	**Error Handling**	**Error Handling** `try-except-finally`, custom exceptions	LO1: Core Python Programming	
6.	**Introduction to Pandas**	**Introduction to Pandas** Series vs. DataFrames, data loading (`read_csv`)	LO2: Data Manipulation with Pandas	
7.	**Data Cleaning**	**Data Cleaning** Handling missing values (`dropna`, `fillna`), duplicates	LO2: Data Manipulation with Pandas	

8.	Half-Term Exam	- Review of LO1 topics - Practice questions and mock assessment - **Half-term assessment** based on LO1 (theory)	LO1 LO2	
9.	**Data Transformation**	**Data Transformation** `apply`, `map`, `groupby`, `pivot_table`	LO2: Data Manipulation with Pandas	
10.	**Advanced Filtering**	**Advanced Filtering** Boolean indexing, `query`, `loc`/`iloc`	LO2: Data Manipulation with Pandas	
11.	**Exploratory Data Analysis (EDA)**	**Exploratory Data Analysis (EDA)** `describe`, `value_counts`, correlation matrices	LO2: Data Manipulation with Pandas	
12.	**Matplotlib Fundamentals**	**Matplotlib Fundamentals** Line plots, bar charts, histograms (`plt.plot`, `plt.bar`)	LO3: Data Visualization	
13.	**Seaborn for Statistical Visualization**	**Seaborn for Statistical Visualization** Box plots, violin plots, pair plots	LO3: Data Visualization	
14.	Final Exam Preparation & Review	- Comprehensive review of all learning outcomes - Practice questions and revision of key topics		
15.	Final Exam	- **Final-term assessment** covering all learning outcomes (theory and practical elements)		
16.	Feedback & Reflection	- Review of final exam - Individual feedback on performance - Reflective discussion on key learning points		
17.	**Customizing Visualizations**	**Customizing Visualizations** Titles, labels, legends, color palettes	LO3: Data Visualization	

18.	**Advanced Plots**	**Advanced Plots** Heatmaps, scatter plots with regression lines	LO3: Data Visualization	
19.	**Interactive Visualization**	**Interactive Visualization** Introduction to Plotly (`plotly.express`)	LO3: Data Visualization	
20.	**Writing Readable Code**	**Writing Readable Code** PEP 8 guidelines, meaningful variable names	LO4: Code Quality & Documentation	
21.	**Modular Programming**	**Modular Programming** Organizing code into scripts/modules (`import`, `__init__.py`)	LO4: Code Quality & Documentation	
22.	**Documentation Best Practices**	**Documentation Best Practices** Docstrings, Markdown in Jupyter (`# %%`), `help()`	LO4: Code Quality & Documentation	
23.	Half-Term Exam	**Problem Scoping** Defining a real-world data problem (e.g., sales analysis)		
24.	**Version Control with Git**	**Version Control with Git** Basic Git commands, GitHub for collaboration	LO4: Code Quality & Documentation	
25.	**Debugging & Profiling**	**Debugging & Profiling** `pdb`, `%timeit`, optimizing Pandas operations	LO4: Code Quality & Documentation	
26.	**Data Collection & Cleaning**	**Data Collection & Cleaning** APIs, web scraping (intro to `requests`, `BeautifulSoup`)	LO5: Capstone Project	
27.	**Analysis & Visualization**	**Analysis & Visualization** Answering key questions with Pandas/Seaborn	LO5: Capstone Project	
28.	**Report Generation & Project Presentation**	**Report Generation & Project Presentation** Code walkthrough, visualization showcase	LO5: Capstone Project	
29.	Final Exam Preparation & Review	LO1, LO2, LO3, LO4	LO1, LO2, LO3, LO4	
30.	Final Exam		LO1, LO2, LO3, LO4	