DEREE COLLEGE SYLLABUS FOR: ITC 1004 Computational Thinking with Python	**3/0/3**
New course: Fall 2026	
PREREQUISITES:	None
CATALOG DESCRIPTION:	Foundational computational thinking; structured programming using Python; logic, control flow, text manipulation, and data processing; fundamentals of data handling; searching and parsing; introductory data retrieval across networks and web services; basic exposure to structured data formats and databases; real-world applications of Python for data analysis; critical and responsible use of AI-assisted programming tools; ethical computing and responsible data use.
RATIONALE:	This course introduces students to the foundations of programming and data handling through practical work in Python. Students are introduced to how data is collected, processed, and used in contemporary contexts. Also, they explore basic techniques for working with data from files and networks and gain introductory exposure to web services and databases. Students also learn to use AI-assisted programming tools critically and responsibly to support coding, debugging, and problem-solving. Ethical computing, responsible data use, and sustainability considerations are integrated throughout the course. Through collaborative activities and a final project, students apply their knowledge to realistic data-oriented scenarios. No prior experience is required.
LEARNING OUTCOMES:	Upon completion of this course, students will be able to: 1. **Demonstrate understanding** of ethical, social, and sustainability considerations in the use of data, AI, and digital technologies. 2. **Develop solutions** by applying computational thinking principles to interpret and address user requirements. 3. **Critically engage** with AI-assisted programming tools to support coding, debugging, and problem-solving, while evaluating the accuracy, limitations, biases, ethical implications, and originality of AI-generated outputs. 4. **Use** Python scripts to work with structured and textual data from files and web sources.
METHOD OF TEACHING AND LEARNING:	In congruence with the teaching and learning strategy of the college, the following teaching and learning methods are used: • Classroom lectures and guided discussions to introduce key concepts related to computational thinking, data, AI, social impact, and sustainability. • Laboratory practical sessions using Python, data files, web-based data sources, and AI-assisted programming tools to support the development of coding, debugging, data processing, analysis, and problem-solving skills. • Problem-based learning activities where students interpret user needs, refine unclear requirements, and develop appropriate computational solutions. • Progress meetings and formative feedback opportunities to support students' development throughout the course and guide improvement in practical work. • Office hours held by the instructor to provide additional academic support and individual guidance. • Use of the Blackboard Learning platform, where instructors post lecture notes, laboratory materials, assignment instructions, tutorials, and additional resources.

<table>
<tr>
<td>ASSESSMENT:</td>
<td>
Summative:
<table>
<tr><td>1st Assessment: Midterm Exam
Short answer and/or multiple-choice questions, and coding problems.</td><td>40%</td></tr>
<tr><td>2nd Assessment: Portfolio
Collection of in-class activities, quizzes, short tasks completed throughout the semester.</td><td>10%</td></tr>
<tr><td>Final Assessment: Project
Coding project and presentation</td><td>50%</td></tr>
</table>
Formative:
<table>
<tr><td>Various in class exercises and case problems</td><td>0</td></tr>
<tr><td>Coursework</td><td>0</td></tr>
</table>
The formative assessments aim to prepare students for the summative assessments and expose them to teamwork.

The 1st summative assessment tests the LOs 1 and 2.
The 2nd summative assessment tests the LOs 1-4.
The final summative assessment tests the LOs 1-4.

The final grade for this module will be determined by averaging all summative assessment grades, based on the predetermined weights for each assessment. Students are not required to resit failed assessments in this module. Failure to pass the module results in module repeat. Students must attempt all assessments.
</td>
</tr>
<tr>
<td>INDICATIVE READING:</td>
<td>
REQUIRED READING:
1. Guleria, P. (2024). Basics of Python Programming. BPB Publications. [Perlego]
2. Sario, A. H. (2025). Python. Azhar Sario Hungary. [Perlego]
3. Instructor's notes

RECOMMENDED READING:
1. Porter, L., & Zingaro, D. (2024). Learn AI-Assisted Python Programming. (2nd Edition). Manning [Perlego]
2. Taherdoost, H. (2025). Information Technology Ethics. CRC Press. [Perlego]
</td>
</tr>
<tr>
<td>INDICATIVE MATERIAL:
(e.g. audiovisual, digital material, etc.)</td>
<td>
REQUIRED MATERIAL: N/A
RECOMMENDED MATERIAL:
1. Ryan Mitchell. (2024) Python Essential Training. LinkedIn. https://www.linkedin.com/learning/python-essential-training-18764650/
2. Walter, N. (2024) Python for Non-Programmers. LinkedIn Learning. https://www.linkedin.com/learning/python-for-non-programmers/
3. Emanuele Bartolesi. (2025) Learning GitHub. LinkedIn. https://www.linkedin.com/learning/learning-github-26923460/
</td>
</tr>
<tr>
<td>COMMUNICATION REQUIREMENTS:</td>
<td>Daily access to the course's site on the College's Blackboard CMS.

Effective communication using proper written and oral English.</td>
</tr>
<tr>
<td>SOFTWARE REQUIREMENTS:</td>
<td>MS OFFICE, Web Browser, Blackboard CMS, IDE such as Visual Studio Code, Python latest version</td>
</tr>
</table>

WWW RESOURCES:	https://learn.github.com/ https://www.onlineide.pro/ https://pythontutor.com/ https://www.python.org
INDICATIVE CONTENT:	1. **Foundations of programming and computational thinking** 1.1. Computational thinking concepts: decomposition, abstraction, pattern recognition, and algorithms 1.2. Variables, data types, expressions, input and output 1.3. Conditional statements, iteration, and control flow 1.4. Basic error handling, testing, and debugging 1.5. Reading, explaining, adapting, and evaluating simple Python scripts 2. **Working with text, data, and web sources** 2.1. String manipulation and basic text processing 2.2. Lists, tuples, dictionaries, and other core built-in data structures 2.3. File input and output 2.4. Common structured data formats such as CSV, JSON, and XML 2.5. Retrieving, parsing, cleaning, and processing data from files and web sources 2.6. Introductory interaction with web services and APIs 3. **Data storage, analysis, and applications** 3.1. Basic principles of data storage and querying 3.2. Introduction to data analysis libraries such as pandas 3.3. Simple exploratory data analysis and interpretation 3.4. Real-world applications of Python across disciplines 4. **AI-assisted programming** 4.1. Using AI tools to support coding, debugging, explanation, and documentation 4.2. Critically evaluating, testing, and refining AI-generated code and responses 4.3. Responsible use of AI, including authorship, originality, and academic integrity 5. **Ethics, society, and sustainability** 5.1. Ethical and responsible use of data and digital technologies 5.2. Privacy, fairness, transparency, and accountability 5.3. Sustainability considerations in data-driven systems and AI