R'Course Syllabus

TITLE: A Practical Introduction to the Python Programming Language

DEPARTMENT: BCOE Computer Science

COURSE FACILITATOR: John Pham

FACULTY MENTOR: Heng Yin

FACULTY MENTOR ROLE:
The Faculty Mentor will be the instructor of record. This includes but is not limited to: assigning the final grades for students and meeting with the course facilitator on a bi-weekly basis to assess the progress of the course.

COURSE DESCRIPTION:
Python is a programming language that is applicable to anyone independent of their field of study. For computing majors, this class will provide a breadth to the core curriculum showing the applications for Python. For non-computing majors, this class will provide a strong foundation on programming which will allow the student to use software in their respective fields. This course will cover the basics of programming in Python, and consist of 2 projects: a data visualization website and a browser automation tool.

KEY LEARNING OBJECTIVES:
At the end of the term, participants will be able to . . .

1. Describe how Python can be used
2. Use Python as a general-purpose programming language
3. Create visualizations with Python
4. Create browser automation scripts

ASSESSMENT:
You will be assessed with the learning objectives through completion of the homework assignments, your data visualization project, and your open-ended project.

EVALUATION OF STUDENT PERFORMANCE

Activity	**Points possible (or percentage)**
Quarter Long: Homework	50%
Data Visualizations Project	20%
Open-ended Project	30%

METHODS OF INSTRUCTION:
Class will be taught through an hour lecture that will introduce basic concepts. Office hours will be available during ACM's open office hours or by appointment. Logistics of these are listed below:

- **Lecture:** Wednesdays, 11 AM - 12PM, Chung 127
- **Office Hours:** Scheduled through email; jpham035@ucr.edu

- **Appointment:** Contact John Pham at jpham035@ucr.edu

READINGS AND RECOMMENDED TEXTS:
These are optional material that will give you a greater depth into the concepts covered in lecture. They will certainly help you with your mini projects.

- Reference Materials
  - Python Language
    - https://learnxinyminutes.com/docs/python3/
    - https://www.codecademy.com/learn/python
- Text Book
  - Automate the Boring Stuff

WEEKLY SCHEDULE: *Show topics, readings, and assignments for each week*

Date	Workshop Topics	Required Readings and Resources	Assignments Due
	Week 1: Logistics, Software, Intro the Programming		
	Week 2: Math Operators, variables, comments, loops, and conditionals		HW1
	Week 3: User I/O, functions and modules		HW2
	Week 4: Modules, GitHub, CLI		HW3
	Week 5: Matplotlib		
	Week 6: Matplotlib animations		HW5
	Week 7: Open-ended project introduction, Email automation		Data Visualization Project
	Week 8: Web scraping, accessing APIs		HW6
	Week 9: Automating your mouse and keyboard		HW7
	Week 10: Open-ended Project Demos		Open-ended Project Due

ACADEMIC ACCOMMODATIONS
If you have a disability or believe you may have a disability, you can arrange for accommodations by contacting Services for Students with Disabilities (SSD) at 951-827-4538 (voice) or specserv@ucr.edu (email). Students needing academic accommodations must first register with SSD and provide required disability-related documentation. If you already have approved accommodation(s), you are advised to notify the faculty instructor of record for this course privately.

ACADEMIC RIGHTS AND RESPONSIBILITIES
All students, faculty, and staff are responsible for understanding and complying with the University's stated academic requirements. Students should feel free to express their thoughts and opinions in an academic forum. Assignments must be completed by the student for whom the work is assigned and without unauthorized aid of any kind.

TECHNOLOGY POLICY

You will need access to a computer for this course. Access during lecture and during office hours will help you follow along.