College of Public Health
DEPARTMENT OF HEALTH ADMINISTRATION AND POLICY
George Mason University

Syllabus – HI 618 – Spring 2026

Course Information	HI 618: Computational Tools in Health Informatics Location: Fairfax Campus | Aquia Building | Room 219 Monday 4:30 PM – 7:10 PM ET
Course Placement	(X) Core ( ) Concentration (X) Elective ( ) Pre-requisite(s)
Instructor	Naren Durbha (sdurbha@gmu.edu) Office Hours: By Appointment
Course Description	This course introduces computational tools used in health informatics. Reviews hardware and software needs and uses. Topics covered include operating systems, virtualization and high-performance computing, basic programming in a scripting language, basic data analysis and data integration skills, and use of specialized software. All topics are covered in context of specific solutions used in health information systems.
Course Objectives	**Upon completion of the course, students will be able to:** • Describe hardware and software requirements needed for health informatics applications • Select appropriate tools for solving computational problems in health informatics • Understand issues related to processing large data files • Prepare computer scripts for automating simple tasks • Be able to use command-line tools in Windows and Unix based operating systems
Course Methodology	The class format will combine reading, lectures, screencasts, discussions, presentations, and other learning tools. The class will be interactive and require every student to be engaged in the classroom discussion and assignments. In addition to the lectures, screencasts and timely completion of assignments, every student will be expected to be an avid consumer of health informatic industry trends, an active participant and a dedicated individual applying what you learn to every element of the course work.
Required textbook(s) and/or materials	The course uses a variety of materials freely available on the web.
Course Grading	

Grade	Percentage
A	96-100
A-	90-95
B+	86-89

<table>
<tr><td></td><td></td><td><table>
<tr><td>B</td><td>80-85</td></tr>
<tr><td>B-</td><td>76-79</td></tr>
<tr><td>C</td><td>70-75</td></tr>
<tr><td>F</td><td>70 and below</td></tr>
</table></td><td></td></tr>
<tr><td>Computer Requirements</td><td colspan="3">This is a computing intensive course, and all students are required to complete assignments and projects using computer software. Health informatics professionals should know their computers well.

Minimum computer (laptop or desktop) system requirements: Multicore (preferably Intel VT/AMD-V), 12GB RAM, at least 512 GB storage (and 200 GB+ free), webcam, speakers, good internet connection.

Mac computers are allowed, but students need to do additional configuration – some assignments require windows. Mac users should be able to use Windows through virtualization software. Students must be able to install software and configure their computers, configure security settings, firewall, etc.

Students are strongly encouraged to backup all contents of their computers on regular basis. Loss of data cannot be used as excuse for late or not submitted assignments/projects.

The class does not require students to purchase any specialized software.</td></tr>
<tr><td>Expectations</td><td colspan="3">Students are responsible for assigned readings, class content and material. Students are also responsible for finding right computer equipment that allows accessing the course materials online and checking email/canvas on daily basis.</td></tr>
<tr><td>Mason Honor Code</td><td colspan="3">The complete Honor Code is as follows:

To promote a stronger sense of mutual responsibility, respect, trust, and fairness among all members of the George Mason University community and with the desire for greater academic and personal achievement, we, the student members of the university community, have set forth this honor code: Student members of the George Mason University community pledge not to cheat, plagiarize, steal, or lie in matters related to academic work.

Use of AI:
It is also normal practice to use AI-based tools to help with answers. However, the submitted work needs to be yours. If any of the code or response is copied or generated, you must clearly indicate which portions of the code and what sources are. When AI tools are used, clearly state what tools and what prompts you use to arrive at answers. Failing to do so is considered honor code violation.

(From catalog.gmu.edu)</td></tr>
</table>

Individuals with Disabilities	**The university is committed to providing equal access to employment and educational opportunities for people with disabilities.** Mason recognizes that individuals with disabilities may need reasonable accommodations to have equally effective opportunities to participate in or benefit from the university educational programs, services, and activities, and have equal employment opportunities. The university will adhere to all applicable federal and state laws, regulations, and guidelines with respect to providing reasonable accommodation as necessary to afford equal employment opportunity and equal access to programs for qualified people with disabilities. Applicants for admission and students requesting reasonable accommodation for a disability should call the Office of Disability Services at 703-993-2474. Employees and applicants for employment should call the Office of Equity and Diversity Services at 703-993-8730. Questions regarding reasonable accommodations and discrimination on the basis of disability should be directed to the Americans with Disabilities Act (ADA) coordinator in the Office of Equity and Diversity Services. *(catalog.gmu.edu)*
E-Mail Policy	**Web address: https://mail.gmu.edu** Mason uses electronic mail to provide official information to students. Examples include notices from the library, notices about academic standing, financial aid information, class materials, assignments, questions, and instructor feedback. Students are responsible for the content of university communication sent to their Mason e-mail account and are required to activate that account and check it regularly. Students are also expected to maintain an active and accurate mailing address to receive communications sent through the United States Postal Service. *(catalog.gmu.edu)*

Evaluation

Grading Components	Percentage of Course Grade
Assignments	25%
Final Presentation/Project	40%
Final Exam	25%
Participation	10%
Total	**100%**

Assignments – **25%**	**Instructions** Each week assignments are required to be uploaded to Canvas. Assignments are due **by Sunday, 11:59 PM, ET** unless otherwise stated. Refer to the course schedule and weekly overviews for details. Each week assignments are posted to Canvas. - Late assignments will be subject to a 20% deduction. - Correction and one-time resubmission of assignments within a week is allowed. However, to have the resubmission considered for grade update, you must have posted your initial answer by the initial assignment due date. Also, your initial answer should show how you tried to solve the specific assignment, that you make a real attempt at solving it, not just general concept. - Do not copy-and-paste from external sources or each other when submitting your answer. -For programming exercises, you are required to include working python code. Do not copy python code directly into Word but submit as a separate .py (or .txt) file. The Python code needs to run to get credit. No partial credit for non-working code. In addition, you need to submit one MS Word document that shows the question, screenshot of the Python code answering the question and screenshot of the code output/results.
Participation – **10%**	Many of you face similar challenges and can help each other in finding solutions. The course instructor can also help many of you to face the same issues with class content. The Course Q and A discussion board is a forum for everyone to discuss the challenges you face in the class. It is loosely moderated by instructor, but the idea is that you learn how to help each other in the online environment. Once you face a methodological or technical challenge, search the Course Q and A discussion board to see if others face similar issues and if there are solutions available. If not, create a new thread by clearly stating the challenge you are facing. Please do not post questions that explicitly ask for solutions of homework assignments. If you see those, do not respond. However, it is fine to ask about specifics of certain portions of the work. While the Course Q and A forum is ungraded, participating is important, as you all need to help each other. However, helping others in the forum affects grade for participation portion of the class. In addition to course Q&A, participation in weekly class meetings counts towards the class participation. It is not required, but those who do not participate should be active in Q&A and other class discussions.

Final Exam – **25%**	Final exam questions will reflect ALL course material. You may use books and online resources; however, the work submitted must be your own. To complete this exam, you must include the question prompts as well as your responses. For Python code, include .py files that can be executed without any modifications. No collaboration is allowed. Each student must submit his own work. All answers must be submitted through Canvas to get graded. No late submission or resubmission is allowed. Provide appropriate citations. Each student will upload their word document and .py files.
Final Project – **40%**	Projects are the most important part of the class and contribute to 40% of class grade. There is no limitation on what the projects are about, as long as they are based on technologies we cover in class and demonstrate Python and computing skills. **Project deliverables include:** • Topic Selection Submitted to Instructor • Mid-Point Check-In, Presentation of Initial Findings to Instructor • Report that describes problems, technical solutions, implementation, and obtained results • The actual working prototype of the solution you create (i.e. working Python code) • Presentation Once you have completed work on your project you are responsible for posting a final presentation (file format: mp4 or similar video) on the designated discussion board You will create an initial post with your presentation and then watch a minimum of three other students' presentations and provide them with comments and feedback **The following are some examples of topics that have been previously presented. You can use the samples as a point of reference, or define your own:** • Analyze data in Python. Write code that loads data, calculates descriptive statistics, does some plots and visualizations, etc. Use it for some meaningful data and results. Write an interactive website in Python. All communications with your python program should be through a web browser. The program may ask about some information and store it, so it can be viewed later. For example, it may be a simple blog" application in which you can submit new entries, and it is posted on top of other entries with date. • Write a backup program in python. The program should look for all files in a specified folder, check if they have changed since the last backup, and copy them to a new location if they change. • Write a survey program that interviews users about a specific topic. The program may ask a series of questions. Depending on the answers to previous questions it will ask more questions. Results should be stored

	in a database. The database should include all responses by multiple people (when program is run multiple times). The program should present various statistics of the submitted responses.
	• Write a simple hospital waiting room simulation. The program will randomly generate how many patients come every hour, and can include parameters in how many staff members are available, how many patients can be cared for within an hour based on severity, etc. Then you can run many simulations so check how many staff are needed, what the average waiting times, etc.
	• Write a scheduling program for university. The program should read a list of courses and a list of available classrooms. The basic scheduling rules are:
	  ○ Two courses cannot be offered in the same classroom at the same time
	  ○ A faculty member cannot teach two courses at the same time.
	  ○ Additionally, you can include information about which classes are typically taken together, so they are not scheduled at the same time
	  ○ You may also include additional constraints such as type of classroom (lab, etc.), size of classroom, etc.
	• Create an online health tracker (html+Python). The tracker will allow users to enter daily information about exercise, calories, weight, etc. The data will is recorded. Users can access historical data and see how the values changed over time.
	
	The mid-point check-in, presentation of initial findings should include the following information:
	• Provide a 2-3 paragraph description of your progress on the final project.
	• The first paragraph should describe your idea for the project. What you plan to achieve and what will be the final product.
	• The remainder of the report should focus on what you accomplished so far and what your progress is.
	• At this point you are expected to have a completely finalized idea what you plan to create and already started to design the solution.
	
	The final project presentation should include the following information:
	• 7-10 minutes in length. While the format for your presentation is ultimately your decision, the key is to make it engaging and explanatory for your audience. Some suggestions would include (but aren't limited to) basic information presented on slides with a screencast on the functionality, showing of source code, execution of program, etc.

Project Report Guidelines - Final project deliverables consist of three elements:

1. Final report
2. 7–10-minute presentation to be shared with your class (via the discussion forum)
3. Working Python program as described in (1) and (2)

The final report should include the following information:

- Background and problem description (2-3 paragraphs or more)
  - Describe what problem you are solving, why it is important, what you want to accomplish by programming it in Python, etc.
- Design of the solution (as needed)
  - This section includes all information about how the Python program is designed, how intersection with users work, what libraries you are using and why, should the code be executed on server or at client computer, etc.
- Implementation
  - Describe all the details of implementation of the program. Include the most important parts of source code and describe how the code works.
- Testing (as needed)
  - Present results of testing, use of the program you wrote. Include example user sessions, input/output form the program, all plots' charts, and other things you prepare.
- Conclusions & Future Work (2-3 paragraphs or more)
  - Present conclusions from your project. Include information about what you learned by doing it. Discuss future work and possible extensions.
- References
  - Remember to include references to all materials that you used, including source code, tutorials, libraries, etc.

***Please note* While it is normal to look up solutions on the web and look up example solutions to specific tasks, it is unacceptable to copy someone else's code and use it as your project. The code needs to be yours and only small parts can be based on work you found. Moreover, you need to know how it works.**

Tentative Weekly Schedule		
Week & Date	Computational (C) Topic	Programming (P) Topic
1. 01/26/2026	Introductions	Introductions
2. 02/02/2026	Introduction to Computing: - Introduction and History - Basics – Hardware Basics – Software	- Overview of Programming - Why Python and how Python? - Python Setup - Install Anaconda Integrated Development Environment (IDE) - Python Data Types - Variable Assignment in Python Your First Python Code: "Hello World"
3. 02/09/2026	Networks and Internet	Conditions, Lists, Loops
4. 02/16/2026	Cybersecurity	Data Structures — Basics, Lists, Tuples, Dictionary
5. 02/23/2026	Microsoft Windows	Functions, Libraries, Numpy and Pandas
6. 03/02/2026		Object Oriented Programming
7. 03/09/2026	Spring Break (NO CLASS)	
8. 03/16/2026	Linux	Python In Linux
9. 03/23/2026		Basic Data Analysis in Python
10. 03/30/2026	Virtualization & Cloud Computing	Data Visualization
11. 04/06/2026		Data Analysis in Python and SQL
12. 04/13/2026	HTML & Web	Web Programming 1 - Intro
13. 04/20/2026	Topics in Computing	Web Programming 2 & 3
14. 04/27/2026	Programming Exercises	
15. 05/04/2026	Collaboration Tools, Versioning Code with Git/Github, IDEs	
16. 05/11/2026	**Final Exam & Final Project**	