




[newtFire {dh}](#)

**Maintained by:** Elisa E. Beshero-Bondar (eeb4 at psu.edu)  **Last modified:** Monday, 24-Aug-2020 09:38:29 UTC. [Powered by firebellies](#).

---

# DIGIT 400: Digital Project Design

## Course Policies

**Fall 2020:** Classes meet M W F 10:10 - 11am over Zoom and in Kochel 77. Zoom attendance is required for all students. The classroom is optional, but even in the classroom you will be connecting to Zoom to team up with your classmates. We may need to divide into groups to rotate attendance in the physical classroom. Group assignments will be made in [Canvas](#). For in-class meetings, we must all wear masks and maintain a safe social distance. Remember: *Your mask protects me, my mask protects you.*

## Schedule: Fall 2020

DIGIT 400: Lionpath class number: 5985. This course fulfills a 400-level requirement for the DIGIT major at Penn State Erie, The Behrend College. You should have taken DIGIT 110 and/or DIGIT 210 before this class, but these pre-requisites are not strictly necessary to your success in this class.

## Instructor

[Dr. Elisa Beshero-Bondar](#) ("Dr. B"), Professor of Digital Humanities and Program Chair of DIGIT.

- E-mail: eeb4 at psu.edu
- Office Hours by Zoom: Tuesdays 11am - 1pm, Thursdays 1pm - 3pm, and by appointment.

## Digital Project Design: Course Description

This course is all about working with computers and digital technology to build cultural resources on the public web. In taking this course you will gain experience with textual scholarship, editing, and digital production, and you will learn a variety of coding designed for systematic building and sharing of information resources.

Students who complete this course will gain skills in practical hands-on programming, digital project management, and web development. Your digital projects will distinguish you as investigators and makers, able to wield computers creatively and effectively for human interests. Your success will require patience, dedication, and regular communication and interaction with us, working through assignments on a daily basis. Your success will **not** require perfection, but rather your regular efforts throughout the course and your documenting of problems when your coding doesn't yield the results you want. Homework exercises are a back-and-forth, intensive dialogue between you and your instructors, and we plan to spend a great deal of time with you individually over these as we work together. Our guiding principle in developing assignments and working with you is that the best way for you to learn and succeed is through regular practice as you hone your skills. Our goal is not to make you expert programmers (as I am far from that myself). Instead, I want you to learn how to apply coding technologies for your own purposes, how to track down answers to questions, how to ask for help, and how to find answers to questions by searching and experimenting. In designing a digital project, you will gain skills in thinking algorithmically (step-by-step) through problems to find good solutions.

## Learning to Code: Our Context

You do not need any background at all with computer programming or web development to succeed in this course. We teach practical programming as a foundational skill (like reading, writing, and arithmetic) that all students should experience regardless of major or background.

You will be learning to write, test, and share code to build a digital project. You will be working in teams to investigate a research topic in the humanities. You students will propose and select the project ideas we will work on in this class, with guidance from me. You will be learning to build projects with eXtensible Markup Language (XML) and languages connected with it, because XML is a powerful tool for modelling texts that we can adapt creatively to our interests and questions. You will learn how to work with regular expressions to match patterns in plain text and "up-convert" them to an XML document. You'll gain lots of experience with writing XPath expressions: a formal language for searching and extracting information from XML code which serves as the basis for transforming XML into many publishable forms. You'll learn to write XSLT: a programming "stylesheet" transforming language designed to convert XML to publishable formats, as well as to extract information and plot it in charts in graphs in Scalable Vector Graphics (SVG). You will learn how to design your own systematic coding methods to work on projects, and how to write your own rules in schema languages (like Schematron and Relax-NG) to keep your projects organized and prevent errors. Depending on the project you work on, you may gain experience with an international XML language called TEI (after the Text Encoding Initiative) which serves as an international standard for coding digital archives of cultural materials. Since one of the best and most widely accessible ways to publish XML is on the worldwide web, you will definitely gain working experience with HTML code (another markup language related to XML). You will be styling HTML with Cascading Stylesheets (CSS), and adding dynamic features to your website with JavaScript. You will also, all along, be working with Git and GitHub for collaborating with the class and your project team, and gaining command line (or shell) experience.

While we are using XML in an academic research context, what you learn here is also important in the tech industry, where XML is the internal format for many general applications (used for bank and hospital records, and the basis of the entire Microsoft Office and LibreOffice software packages). XML is important, too, for web developers, where the HTML (hypertext markup language) of web pages to be viewed in browsers is often expressed as a form of XML (as we will be applying it). Developers of relational databases work widely

with XML, too, as a universal data interchange format. Finally, gaining fluency in the XML "family of languages" in a humanities context may give you a strong foundation in practical computing experience and I hope what you learn here will help you build a distinctive portfolio of skills and projects.

## Learning Objectives:

- Work with Texts as Artifacts—As Physical and Virtual Objects:
  - Generate "digital surrogates": digitally represent facsimiles of rare manuscripts and other kinds of documents, and make their content digitally searchable.
  - Reflect and write on the issues and problems with digital representation, as well as the capacity of the digital medium to enhance or add dimensions to a physical text.
  - Learn and practice coding in eXtensible Markup Language (XML) and related coding technologies: to "mark up", process, and extract information about the structure, physical condition, and cultural contexts of textual artifacts.
- Gain experience with information retrieval, distant reading, and autotagging techniques:
  - Write code to apply searching and data extraction methods through multiple kinds of pattern-matching algorithms, including forms of regular expression matching. Take conventional boolean searches and library database searches to new levels.
  - Apply "mining" and "drilling" methods to interact with texts and visualizations differently than we could do "manually" or with unassisted eyes and brains.
  - Learn how to "autotag" enormous texts or collections of texts, for practical results: to code the structure of enormous texts from a distance, in order to navigate them and make them accessible through distant reading.
  - Reflect on the strengths and limitations of data processing and visualization.
- Gain Project Design and Editing Experience:
  - Gain digital editing experience with proposing, designing, and contributing to a digital research project
  - Gain experience with collaborating and sharing code using a version control system (GitHub) in a team repository
  - Transform XML code into publishable web formats, to build or contribute to a project website.
  - Design navigation elements, and build visual aids and models (such as timelines and tree diagrams) from texts: to generate charts and images from extracted data
  - Gain experience with plotting digital graphs and charts
- Last but not least: Discover that you read and write with "new eyes," with greater precision and agility, thanks to your adventures with digital projects!

## Optional Textbook and Other Class Resources

- Michael Kay, *XSLT 2.0 and XPath 2.0: Programmer's Reference*, 4th edition (Wiley Publishing, 2008) ISBN-13: 978-0-470-19274-0 This book is optional, and I have not requested it at the bookstore. I have two copies and it is available in the Penn State Library as an e-book. This is really **the** authoritative word on XSLT and XPath, written by a designer of the official W3C specifications of XSLT 2.0 that we're using. We are learning from this book ourselves and consult it frequently! We're not requiring that you buy it,

but we recommend it to have a powerful reference at your fingertips and for learning more on your own. There's a kindle edition available but poorly designed for searching, so we (actually) prefer the hardcover print edition. If you're going to purchase it, be sure you pick up the latest edition (from 2008), and not the earlier versions.

## Other resources: Project Guidelines and Past Student Projects

- [Guidelines for Projects Developed in This Course](#)
- [Student Course Projects](#)

## Explanatory Guides and Exercises: Complete List

### Class Web Resources:

- [Course Home Website: https://newtfire.org/courses/digitProjectDesign/](https://newtfire.org/courses/digitProjectDesign/) Home of our syllabus and schedule.
- [digitProjectDesign-Hub: https://github.com/newtfire/digitProjectDesign-Hub](https://github.com/newtfire/digitProjectDesign-Hub) Class GitHub Repository and Issues Board
- [Canvas: https://courseweb.pitt.edu](https://courseweb.pitt.edu) To submit homework assignments and exams and read private course announcements
- [File Conventions for Canvas Assignments](#)

## Grading:

### Homework Exercises (30%):

To keep up with this class, you must work on exercises regularly. Each day will involve some small assignment, to prepare you for the next of class, and to help you to build your course project. **90% Rule:** If students do not submit at least 90% of the regular homework assignments, the grade for the homework portion is based on the percentage of homework they completed. Students should therefore aim to submit at least 90% of the regular homework assignments, and complete at least 90% of the work in each component of the course.

**About homework assignments:** Coding and project review exercises in this course are about your active learning, and not—as in other courses—a way of testing whether you have already learned something we covered in class or in an assigned reading. You may often need to look up how to do something that you don't already know how to do. Often, there will be multiple ways of accomplishing the task and we are not simply looking for you to do things perfectly in just one way. We are instead looking for a record of your learning process as you take on a challenge. Documenting problems is key to learning, and sometimes just writing out what you are trying to do helps lead you to a solution! There may be times when you don't get the result you want in the homework, and that is to be expected! In those cases you can still get full credit for the assignment if you've made a serious attempt and if you submit, along with your code, a description of what else you tried, what results you expected, what results you got, and what you think went wrong. Getting stuck is part of the learning process. You will see me get stuck sometimes, and I will need *your eyes* to help me fix something! As long as you've described your

understanding of the problem and your attempts to resolve it on your own, you will do well: documentation of how you get stuck is key. One of our goals is to form a supportive coding community in this class, so we are comfortable with *unsticking* each other,

I will read and evaluate all student homework, and will post assessments on Canvas. Coding assignments are assessed as "**check plus**", "**check**", and "**check minus**", or "**redo**". Don't think of these as grades, since, if you resubmit a "redo" to correct a serious problem, you will receive full credit for the assignment. My comments on homework are feedback for learning purposes. If you have not engaged with the assignment adequately (whether that means solving the tasks or discussing the coding obstacles you encountered and how you dealt with them), we will ask you to meet with us to review the issues and then complete a followup (redo) task in order to receive credit. For assignments with posted solutions, I will invite you to review the posted solution on GitHub and comment on it (we will show you how to do this) to address something you learned from the solution or did in a different way. For some assignments where we review posted solutions and line-comments together in person or in class, we will write back to you with individual comments only if your specific submission raises an issue that we don't address elsewhere. If we don't return your assignment, that means that we have nothing to add to our posted solution, but should you have any specific questions after you've read our posted solution, please ask the instructors. And we will go over assignments together to get the class unstuck on things in our regular class meetings.

**Issue posts:** Throughout the course, we'll assign discussion posts on [our class GitHub site](#) in which you will respond to online readings or evaluate web resources. Your posting should do more than summarize the article or site (which you could just do by skimming or reading the first paragraph), but should demonstrate a thoughtful reflection on specific ideas and issues. When evaluating a web resource, don't simply praise or condemn it without going into details about why a key component is effective or poorly designed. Good posts demonstrate care and reflection, and you may choose to respond to the overarching ideas of a piece, or to selected details of specific interest. These posts are scored as "check plus", "check", and "check minus".

## **Participation: In Class and on GitHub (15%):**

Coding and programming in real life is a social activity, and professionals in the real world aren't "know-it-all" experts who work alone, but rather are tuned into discussion boards and regularly ask and answer questions to stay sharp and to learn from their community. In this class, we want you to work together and talk to each other and your instructors as your community resource, so we have built this into our course participation grade as a formal expectation. **Beginning by week two, we'll expect each student to post at least once per week on [our course GitHub repo](#)**, and we strongly encourage you to do more than this minimum. Earn an "A" in participation by asking questions, making suggestions, and sharing helpful resources you've found. Help each other out by trying to answer questions on GitHub (and read the instructor posts too as we wade in to help). Your instructors will likely be dominating the class time as we model concepts and methods, so the GitHub Issues board gives the students a good space to form into a coding community to help each other and reflect together. Also, if you have a question about an assignment, **always think of our GitHub Issues board as your first resource** to check for helpful hints and to post your questions, because others may have the same question and answers are best shared! Of course you may e-mail us, but we really prefer you go the discussion board first, and doing so is, after all, worth course credit as your participation grade.

## **Tests (15%):**

As scheduled throughout the course there will be several (probably about five or six) tests on the various kinds of coding we are learning in the course, and we will drop the lowest grade. All will be take-home or taken online in between classes. They are open-book, open notes, but they must be completed individually and are designed to demonstrate that you have learned from the class material, coding assignments, and posted solutions. Tests may resemble homework assignments, but unlike homework exercises, these are given letter grades. These are given grades because they are evaluative and involve demonstrating what you have learned after we have finished a coding unit.

## **Project (40 %):**

Throughout the semester you will be working as part of a team on a course project. Early in the semester each student posts a proposal for a semester project to work with a text (or collection of texts) in the public domain and a set of research questions to explore in a coding project. Teams will form around a selection of these projects in mid-September and begin work, performing document analysis, developing and implementing a system of markup and project rules, marking up text following that system, writing programs to conduct research and create a resource to share on a public website you will develop together that represents your investigation and your conclusions. Each project team must meet regularly together and check in with me outside of class for project planning and discussion. Each of the project components described below adds up to 40% of your grade for the course.

**Project Checkpoints** There will be a series of project checkpoints to complete, by set due dates throughout the semester. Each is worth 5% of the final course grade (a total of 20%) and a letter grade on the following scale: "exceeds target" (A+), "meets target" (A), "some progress" (B), "negligible progress" (C), "no progress" (F). Each checkpoint will expect you to complete a stage of serious work on the course project with your project team. Project Checkpoints are met using the Issues and/or Projects tabs on your project GitHub repository and by posting files on the project website on newtFire.

The course project develops throughout the semester, but is fully assembled in the final weeks of the course and submitted in two places, through code and documentation shared in your GitHub repository and on your project website due in Finals Week. Projects are evaluated as a team effort, but if unequal effort is observed, project members may receive different project grades accordingly. The Final Project grade is worth 20% of the course grade.

## **Grading Scale:**

Grades for the course are calculated and posted on Canvas, and follow this standard scale: A: 93-100%, A-: 90-92%, B+: 87-89%, B: 83-86%, B-: 80-82%, C+: 77-79%, C: 70-76%, D: 60-69%, F: 59% and below. In taking the course on a S / NC (pass-fail) basis, students must earn a C to receive Satisfactory credit.

## **Course Policies:**

Each day we are covering material that builds on earlier material and assignments, so your success depends upon regular attendance and completing each assignment on time.

### **Due dates and why we need them:**

Your daily homework for this course is time-sensitive! Coding assignments, response papers, and other homework exercises must be uploaded to Canvas (or GitHub or our web server as specified), by the due date and time indicated on the class schedule. Homework assignments will be posted online to our class website and linked from our schedule, so students who miss class are nevertheless expected to consult the schedule and submit assignments on time. Because we post and share answers to homework exercises after submission deadlines, we will usually not accept late homework submissions.

### **Exam Policy:**

All exams will be take-home, to do on your own time, with submissions due in Canvas or by web submission. Because I will be posting answers and sharing them in class, I do not allow people to write exams after the solutions are posted. However, I will drop your lowest exam score for the class, so that you may miss one exam without penalty.

### **Attendance and Classroom Courtesy:**

I am not calculating your attendance, but I will expect your active presence and interaction with me and your classmates this semester, as we need to rely on each other to learn and develop projects.

Our class is fast paced and requires that we all be making the best use we can of our in-person class sessions. Arriving late and leaving early (physically or remotely!) disrupts the important collective mental activity of class. So does in-class texting and checking your cell phone. Whether you are joining over Zoom or in the physical classroom, I ask that you put mobile devices in Do Not Disturb mode. While class is in progress, talking disruptively, leaving the classroom, texting or using a cell phone or computer, reading a newspaper, or other distracting behavior will be actively discouraged.

### **Covid and Attendance**

When present on campus together this semester, whether in the classroom or elsewhere, *always* wear a mask and wear it appropriately (over your mouth and nose). We are being provided with some personal protective equipment (PPE) in the form of plastic face shields to be worn *in addition to masks* if we need to look at each other's computer screens. It is a good idea to bring hand sanitizer with you to campus and sponge off your work area in class. I will ask you to put on your mask if I see you without one, or to fix your mask if you are not wearing it correctly. Non-compliance by any of us means we have to leave the classroom immediately for our own safety, and I have to report anyone not masking or distancing as a violation of the Student Code of Conduct. Remember, *My mask protects you; your mask protects me.*

Please do not attend our physical class if you are not feeling healthy! This is not the semester to suffer through a fever or chills heroically to attend class in person. Stay home, report symptoms, get tested. This applies to me as your professor as well as to you!

If you need to miss classes for health reasons, make arrangements with me and your peers to catch up. We will always be meeting on line (via chat and GitHub asynchronously and via Zoom for class meetings) and we will find ways to keep you looped in.

**Our Covid Mixed Mode:** I am planning for the class to be meeting in mixed ways, with some of you attending in Zoom and some of you in the classroom. I hope we can maintain class meetings so we see each other in person once per week before Thanksgiving, but we are ready to be fully remote when needed. I am preparing much course material to be recorded and watched and read on your own time. In-class meetings are paired with synchronized Zoom sessions so you all, wherever you are, can talk to me live. Think of the in-class sessions as live conversations where we share screens and discuss homework, tests, and projects.

Finally, if I feel as if we are risking our health with in-class meetings, I will simply move us all to Zoom to protect our group. Our health comes first!

### **Student (and Faculty) Health and Wellness Services**

If any of us, you students or me, are feeling sick, with COVID or flu-like, or other serious ailments this semester, please contact Behrend Student Health & Wellness Services at **814-898-6217**. None of us can be sure what will happen with the COVID pandemic, and we are taking on a great risk this semester. Reporting in when you do not feel well is not shameful; it is responsible and important to protect yourself and our community.

Also, this semester may be more stressful than usual with so much uncertainty! Many students at Penn State face personal challenges or have psychological needs that may interfere with their academic progress, social development, or emotional wellbeing. Seek help! The university offers a variety of confidential services to help you through difficult times, including individual and group counseling, crisis intervention, consultations, online chats, and mental health screenings. These services are provided by staff who welcome all students and embrace a philosophy respectful of clients' cultural and religious backgrounds, and sensitive to differences in race, ability, gender identity and sexual orientation. Counseling and Psychological services are available through the Personal Counseling Office in Reed Union Bldg. Rm 1: **814-898-6504**.

### **Equity**

Penn State takes great pride to foster a diverse and inclusive environment for students, faculty, and staff. Acts of intolerance, discrimination, or harassment due to age, ancestry, color, disability, gender, gender identity, national origin, race, religious belief, sexual orientation, or veteran status are not tolerated and can be reported through Educational Equity via the Report Bias webpage (<http://equity.psu.edu/reportbias/>).

### **E-mail:**



Each student is issued a University email address (username@psu.edu) upon admission. This email address may be used by the University for official communication with students. Students are expected to read email sent to this account on a regular basis. Failure to read and react to University communications in a timely manner does not absolve the student from knowing and complying with the content of the communications. The University provides an email forwarding service that allows students to read their email via other service providers (e.g., Hotmail, AOL, Yahoo). Students who choose to forward their email from their psu.edu address to another address do so at their own risk. If email is lost as a result of forwarding, it does not absolve the student from responding to official communications sent to their University email address. To forward email sent to your University account, go to <http://accounts.psu.edu>, log into your account, click on Edit Forwarding Addresses, and follow the instructions on the page. Be sure to log out of your account when you have finished.

## Academic Integrity

Penn State Erie, The Behrend College, puts a very high value on academic integrity, and violations are not tolerated. Academic integrity is the pursuit of scholarly activity in an open, honest and responsible manner. Academic integrity is a basic guiding principle for all academic activity at The Pennsylvania State University, and all members of the University community are expected to act in accordance with this principle. Consistent with this expectation, the University's Code of Conduct states that all students should act with personal integrity; respect other students' dignity, rights and property; and help create and maintain an environment in which all can succeed through the fruits of their efforts. Academic integrity includes a commitment by all members of the University community not to engage in or tolerate acts of falsification, misrepresentation or deception. Such acts of dishonesty violate the fundamental ethical principles of the University community and compromise the worth of work completed by others." (Senate Policy 49-20 and G-9 Procedures. Any violation of academic integrity will receive academic and possibly disciplinary sanctions, including the possible awarding of an XF grade which is recorded on the transcript and states that failure of the course was due to an act of academic dishonesty. All acts of academic dishonesty are recorded so repeat offenders can be sanctioned accordingly. More information on academic integrity can be found at: <http://psbehrend.psu.edu/intranet/faculty-resources/academic-integrity/academic-integrity>.

**Source Citation and Plagiarism:** One goal of our course is to reflect on how best to cite sources in digital contexts. We will consider how and why such citations differ from documenting printed texts. We will also consider the ease and frequency with which digital texts and graphics are plagiarized on the worldwide web, and discuss how the omission of source citations detracts from the authority of a digital information resource. We expect you to practice mindful source citation, and plagiarism on your part will have very serious consequences.

Representing the voice of another individual as your own voice constitutes plagiarism, however generous that person may be in "helping" you with an assignment. Turning in an assignment generated collectively under the name of a single individual is considered plagiarism. **When instructed to collaborate on a project, project collaborators share collective authorship and should identify themselves directly as a team.** To avoid plagiarism, cite your sources whenever you quote, paraphrase, or summarize material, or use digital images from any outside source (including websites, articles, books, course readings, Courseweb or GitHub postings, or someone else's notes). When using the "copy" and "paste" features as you read and research, be sure that you are carefully marking that these passages are unprocessed from their source, so that you know to process it later. Forgetting to do so not only produces sloppy work but (whether you intended it or not) results in a false representation. As long as you make a good faith and clear effort to cite your sources, you will not be faulted for plagiarism, but your work will be penalized if citations are inaccurate, unclear, or lack important information.

That said, the coding and digital development we do encourages collaboration, and for that reason we adopt our colleague David Birnbaum's [Collaboration policy](#), since his course is very similar to ours. This policy specifies that students identify collaborators in a comment on submitted assignments and take care on projects that all students contribute equally (and no student is contributing excessively more than what everyone else has done). When joining a group homework session, always work on the assignment by yourself first so you can be an equal participant, and write up the assignment *by yourself, after* the session is over so you take care not to copy from the other students. While we want you to consult with each other, you are responsible for doing all your writing and coding by yourself, using your own words.

## Disability Services:

Penn State welcomes students with disabilities into the University's educational programs. Every Penn State campus has an office for students with disabilities. Student Disability Resources (SDR) website provides contact information for every Penn State campus (<http://equity.psu.edu/sdr/disability-coordinator>). For further information, please visit Student Disability Resources website (<http://equity.psu.edu/sdr/>). In order to receive consideration for reasonable accommodations, you must contact the appropriate disability services office at the campus where you are officially enrolled, participate in an intake interview, and provide documentation: See documentation guidelines (<http://equity.psu.edu/sdr/guidelines>). If the documentation supports your request for reasonable accommodations, your campus disability services office will provide you with an accommodation letter. Please share this letter with your instructors and discuss the accommodations with them as early as possible. You must follow this process for every semester that you request accommodations. Penn State Behrend's Disability Services Coordinator is Stacey Walbridge ([slw44@psu.edu](mailto:slw44@psu.edu))

## Inspiration

We gratefully acknowledge [David Birnbaum's Digital Humanities course](#) at the University of Pittsburgh as our starting point and supporting resource for much of our development. Other inspirational resources include:

- [The Programming Historian \(full collection of tutorials\)](#)

## Projects that inspire us:


- [Obdurodon](#): where we learned what we can teach, and where we're still learning.
- [Venice Time Machine](#): very ambitious, enormous project team of faculty and students to study and model a thousand years of Venice, digitizing "kilometers of archives."
- [Map of Early Modern London](#)
- [Lord Byron and His Times](#): The very thoughtful stylistic design of this important project reproduces the style of nineteenth-century print and layout. The content makes many rare materials about Lord Byron's social network searchable and connected to the web of linked open data.
- [The Shelley-Godwin Archive](#): digitizes the manuscripts of Percy and Mary Shelley, and Mary Shelley's parents, William Godwin and Mary Wollstonecraft—manuscripts often written in multiple hands. Provides an important study of the Frankenstein notebooks

to demonstrate how much of a role Percy Shelley played in the writing of Frankenstein. The archive provides a good model of the use of TEI for manuscript encoding and of complex and multiple visualizations of manuscript texts.

- [TokenX: a text visualization, analysis, and play tool](#)
  - [A Tour Through the Visualization Zoo](#)
  - [Clay Shirky on Love, Internet Style](#) (9 minutes of Youtube inspiration: on what lasts, and why community matters in our digital worlds.)
-



[newtFire {dh}](#)

Maintained by: Elisa E. Beshero-Bondar (eeb4 at psu.edu)  Last modified: Monday, 24-Aug-2020 09:28:21 UTC. [Powered by firebellies](#).

---

# DIGIT 400: Digital Project Design

**Fall 2020 Syllabus (Schedule)** Classes meet M W F 10:10 - 11am over Zoom and in Kochel 77. Zoom attendance is required for all students. The classroom is optional, but even in the classroom you will be connecting to Zoom to team up with your classmates. We may need to divide into groups to rotate attendance in the physical classroom. Group assignments will be made in [Canvas](#). For in-class meetings, we must all wear masks and maintain a safe social distance. Remember: *Your mask protects me, my mask protects you*.

## Read the [Course Description](#)

This contains a detailed explanation of course policies and the basis for grades.

## [Jump Down to the Schedule](#)

This link jumps to the closest day to today's date. Review the schedule as we get started to get a sense of how this course will work on a daily basis.

## All the Tools You Need As We Begin:

Download and install the following software on your own personal computer(s) on or before the first day of class. These software tools are available in our campus computing labs, too.

1. [oXygen/](#). The DIGIT program has purchased a site license for this software, which is installed in Kochel 77 and the Lilley Library computers. The license also permits students enrolled in the course to install the software on their home computers (for course-related use only). When installing this on your own computers, **you will need the license key**, which we have posted on our course Announcements section of [Canvas](#).
2. Zoom: [Make sure your Zoom installation is up-to-date](#), and you are ready to connect. The Zoom link for our regularly scheduled class meetings is posted in Canvas: Look for the Zoom menu option.

3. All students require a good means of secure file transfer (SFTP) for homework assignments and projects (also available in the campus computer labs). There are several good options available. We recommend you download and install on your own computers one (or more) of the following, depending on your platform: (Feel free to experiment with these and others!)
  - **Windows users:** one of the following FTP clients—the functionality is similar:
    - [FileZilla](#) (This is our favorite client because it behaves the same way across platforms.)
    - [WinSCP](#) (This is one we used for a long time, since the 1990s, but we now use SSH and Filezilla more frequently.)
    - [SSH Secure Shell Client](#)
  - **Mac users:**
    - [FileZilla](#) (This is our favorite client because it behaves the same way across platforms.)
    - or [Fetch](#) (students may obtain free licenses at <http://fetchsoftworks.com/fetch/free>)
  - **Linux users:** You probably don't need to install anything, but look at how your system handles secure file transfer (SFTP). (FileZilla or other clients designed for Linux environments.)
4. Rusty with coding? Don't remember anything from DIGIT 110? Don't worry! Past students in this course who never saw anything like markup or XML code have designed projects ([like these](#)) and even spoken about them at undergraduate conferences! You will learn to develop your own digital tools and how to manage digital projects as teamwork.

#### Class Web Resources:

- [newtFire](#): Our project development site, where you will be publishing your projects.
- [digitProjectDesign-Hub](https://github.com/newtfire/digitProjectDesign-Hub): <https://github.com/newtfire/digitProjectDesign-Hub> Class GitHub Repository and Issues Board
- [Canvas](http://canvas.psu.edu): <http://canvas.psu.edu> To submit homework assignments and exams, read private course announcements, access Zoom class meetings and video recordings.
- [File Conventions for Canvas Homework Assignments](#)
- [Explanatory Guides and Exercises: Complete List](#)

Week 1	Topics	Do before class	Group A	Group B
<b>M 08-24</b>	Welcome! Introduction to the course. <a href="#">Clay Shirky on Love, Internet Style</a> and <a href="#">the Ise Shrine</a> (9 minutes) What does an ancient wooden shrine have to do with digital projects? Intro to XML in oXygen and some things we can build with it.	Respond to Dr. B's Poll (see Canvas / Penn State email). Find the Zoom link for class (on Canvas / Penn State email).	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>W 08-26</b>	XML coding: a poem; a recipe. Elements, attributes, comments, escape characters, and "pretty-printing" in <oXygen/>.	Install oXygen XML Editor and add our license key so we can all use this during our meeting today. Read my <a href="#">Introduction to XML</a> and experiment with the code in the tutorial: Can you tell what makes markup <b>well-formed</b> or not?	Kochel 77 + Zoom (audio muted)	Zoom remotely

<b>F 08-28</b>	Discussion of homework, and XML: Well-formedness vs. Validity. XML projects in digital humanities. Introduce <a href="#">past student projects</a> .	Complete <a href="#">XML Exercise 1</a>	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>Week 2</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 08-31</b>	Discussion of homework (XML Exercise 2). Introduce the Bash shell and GitHub.	Complete XML Exercise 2.	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>W 09-02</b>	Getting started with command line, git, and GitHub. Mindful file management. Markdown vs. markup in GitHub issues: Hands-on work with our class <a href="#">GitHub repo</a> .	Git Exercise 1 (setting up your GitHub account). Watch my video introducing the Bash shell and "home" on your computers. Practice some shell commands	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>F 09-04</b>	Learning git commands and working at command line (hands-on exercise).	Review feedback on your coding exercises so far and submit revisions if I asked you to. Complete XML Exercise 3: Mark up a text of your choice (any genre, manageable but reasonable size, any language). Work on applying attributes with your elements in a careful and systematic way.	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>Week 3</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 09-07 Labor Day, PSU classes in session anyway!</b>	Overlapping Hierarchies: Group Exercise with "Ozymandias."	Complete Git Exercise 2.	Zoom remotely	Kochel 77 + Zoom (audio muted)

<b>W 09-09</b>	Building a code repository as part of a digital project. User Experience (UX): Launch discussion.	Git Shell Practice (starting today for seven days): Make sure your personal repo and the class repo are properly set up and cloned to your computer. Using your Bash shell, practice some basic git commands to build a habit. On alternating days, push to the other repo (if you pushed to your personal repo today, tomorrow you will pull and push a file to our class repo).	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>F 09-11</b>	Hands-on introduction to schemas with Relax NG: How to write the rules for an XML project	Post in UX Discussion exercise: Choose a digital project to explore (from a list to be posted), and write a post addressing: 1) something interesting the site is inviting us to explore, and 2) the effectiveness of the user experience. Read <a href="#">Intro to Relax NG</a> to get a sense of what it is (we'll go over it in class.)	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>Week 4</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 09-14</b>	Relax NG: data types, mixed content. Document data modeling.	Watch my video on setting up oXygen to do Relax NG. Read and work with the and complete Relax NG Exercise 1. Review XML syntax and consult the <a href="#">Intro to Relax NG</a> as you work on this, and ask for help on GitHub if you get stuck. Continue GitHub practice!	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>W 09-16</b>	Troubleshooting and debugging Relax NG issues. Tour of course projects with strong research questions. Start Class GitHub Project Proposals	Complete Relax NG Exercise 2	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>F 09-18</b>	DH projects with visual dimensions; research questions to explore patterns and concepts. Workshopping project ideas.	Complete Relax NG Exercise 3. Post / respond to project proposal (now through M 9/21). Each student must respond to at least one proposed idea from another student and indicate suggestions or further ideas.	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>Week 5</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>

<b>M 09-21</b>	Form project teams!	Post a project proposal / respond to another. Each student must respond to at least one proposed idea from another student and indicate suggestions or further ideas.	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>W 09-23</b>	Review Relax NG: Common issues in homework. Project discussion / initiation time in class.	<ul style="list-style-type: none"> <li>• <b>Project Checkpoint 1:</b> Launch the project GitHub repo and post an issue. Post in the Issues board your available meeting times to help determine a regular meeting time for your group.</li> <li>• Revise Relax NG exercises if you were asked to do that.</li> </ul>	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>F 09-25</b>	Initiate first take-home test: Relax NG.	Install FileZilla (or other SFTP client) if you have not done so already: we'll use it next class! Follow instructions posted on <a href="#">Canvas</a> for setting up SSH keys to access your personal webspace on Newtfire.	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>Week 6</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 09-28</b>	Building on the Web: Introducing HTML and CSS: Make a simple index.html page. Make a simple CSS page. CSS resemblance and difference from Relax NG (those curly braces). Test / troubleshoot using SSH Keys to access our web server.	<ul style="list-style-type: none"> <li>• <b>Complete test: Relax NG</b></li> <li>• Read / watch our <a href="#">Introduction to HTML</a> in preparation for next class.</li> </ul>	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>W 09-30</b>	What is "index.html" to a web server? Website addresses and file directories on a remote web server Hands-on: Work with FileZilla (or other SFTP client) to connect to the Apache Server for newtFire. SSH keys. File directories and their association with web URLs. How to customize SFTP (Filezilla) to work with your GitHub repo.	Consult our <a href="#">Introduction to Cascading Stylesheets (CSS)</a> to help complete HTML/CSS Exercise 1 (Do not submit this on Canvas, but instead use SFTP on your Newtfire webspace.)	Zoom remotely	Kochel 77 + Zoom (audio muted)



<b>F 10-02</b>	HTML and CSS <ul style="list-style-type: none"> <li>• Mindful file management: mirroring directory structures on GitHub and the web server</li> <li>• Web browsers and display variations</li> <li>• CSS Box Model</li> <li>• Introduce Flexboxes (see <a href="#">tutorial</a>)</li> </ul>	Complete HTML / CSS Exercise 2 (submit this to personal / project web space). Consult Learn CSS Layout, Flexbox tutorial, and w3 Schools CSS Reference as you code. (Also, check out Paletton (or hunt for other color scheme generators on the web) to help think about choosing a balanced color scheme for your website. Experiment with writing CSS to control font, layout, color, backgrounds.	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>Week 7</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 10-05</b>	Slack chat and GitHub project management tools for improving communication, completing tasks. Project web work. Project customizations and boilerplate with Server Side Includes (examples). Decide on project website directory names and URLs.	<b>Project Checkpoint 2:</b> <ul style="list-style-type: none"> <li>• Locate all sources for project XML markup</li> <li>• Establish a clear file directory structure on your project GitHub, including a distinct directory for website files</li> <li>• Make sure all project team members work consistently with the GitHub directory structure (all agree on it, no one changes it without notice)</li> <li>• Strong start on a project schema (Relax NG)</li> <li>• A good quantity of documentation and/or markup is present in the GitHub repo.</li> </ul>	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>W 10-07</b>	Introducing Regular Expressions and autotagging. Regular patterns in documents. How to start? inside-out or outside in. "close-open" strategy.	Read <a href="#">Intro to Regular Expressions</a>	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>F 10-09</b>	Regular Expressions: thinking algorithmically. Greedy matching. Work on Regex together.	Regex Exercise 1	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>Week 8</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>

<b>M 10-12</b>	Regex: simplifying over-complicated expressions. Selecting for what's not there. Searching over highlighted portions of a document.	Regex Exercise 2	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>W 10-14</b>	Project review	<b>Project Checkpoint 3:</b> <ul style="list-style-type: none"> <li>Some web development for site; navigation menu, page organization</li> <li>Refined schema</li> <li>XML markup: significantly more progress than the last checkpoint. Everyone's files are working with the project schema</li> </ul>	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>F 10-16</b>	Navigating XML with XPath: Introducing the XPath window in <oXygen/>, functions, axes, path steps /, and predicate filters [ ] Hands-on XPath navigation with Hamlet.	Read our <a href="#">Introduction to XPath: Follow the XPath!</a> As you read, try experimenting with the XPath expressions on our page, by downloading the explainXPath.html file, opening it in oXygen, and experimenting in the XPath window with some of our expressions.	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>Week 9</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 10-19</b>	Work on XPath 1 and 2 together. Using XPath axes. Predicate expressions [ ], (grouping).	XPath Exercise 1	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>W 10-21</b>	Go over XPath 2, start XPath Exercise 3 together. XPath Functions, simple map vs. arrow operator. Using functions inside predicates.	Complete XPath Exercise 2	Kochel 77 + Zoom (audio muted)	Zoom remotely

<b>F 10-23</b>	Go over XPath Exercise 3. Introduce string functions	Complete XPath Exercise 3, or get as far as you can	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>Week 10</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 10-26</b>	Combining regular expressions with XPath. String functions. Getting ready for XSLT	Get as far as you can with XPath Exercise 4: String functions	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>W 10-28</b>	Introducing XSLT (eXtensible Stylesheets Language Transformations). XML to XML, XML to HTML. Namespaces Setting up oXygen to write XSLT.	Read Introduction to XSLT / orientation video	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>F 10-30</b>	How to write XSLT to change XML into HTML	XSLT Exercise 1: An Identity Transformation	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>Week 11</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 11-02</b>	Go over XSLT to HTML. XPath in XSLT. Initiate XPath take-home test.	XSLT Exercise 2 (HTML list)	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>W 11-04</b>	XPath from test, worked into XSLT	Complete <b>XPath test</b>	Kochel 77 + Zoom (audio muted)	Zoom remotely

<b>F 11-06</b>	XSLT: How templates work. Push and pull processing. Write XSLT to make an HTML table together in class.	XSLT Exercise 3	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>Week 12</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 11-09</b>	XSLT for editions: push processing. Styling: working with XSLT and CSS together.	XSLT Exercise 4 (review <a href="#">Attribute Value Templates</a> with this). Read about <a href="#">Modal XSLT</a>	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>W 11-11</b>	Review/ work on Modal XSLT in XSLT Ex 5. Introduce xsl:sort	XSLT Exercise 5	Zoom remotely	Kochel 77 + Zoom (audio muted)
<b>F 11-13</b>	XML to HTML with CSS. Apply to projects	XSLT Exercise 6	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>Week 13</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 11-16</b>	Work toward Project Checkpoint 5	XSLT Exercise 7: Applied to your projects.	Zoom remotely	Kochel 77 + Zoom (audio muted)

<b>W 11-18</b>	XML that makes graphics: SVG (Scalable Vector Graphics: Drawing elements and screen grid coordinates <ul style="list-style-type: none"> <li>• <a href="#">introductory slideshow</a></li> <li>• Play with <a href="#">w3schools SVG Tutorial</a></li> </ul>	<b>Project Checkpoint 5:</b> <ul style="list-style-type: none"> <li>• Goal: XML markup is mostly complete and ready for active processing with XSLT</li> <li>• Work on XSLT to HTML transformations to build content for the project website</li> <li>• Improve the project website, update its organization and navigation</li> <li>• Project intros: introduce the team members</li> <li>• Other goals specific each team</li> </ul>	Kochel 77 + Zoom (audio muted)	Zoom remotely
<b>F 11-20</b>	XSLT to SVG: working with variables to plot coordinate space	SVG Exercise 1	Zoom remotely	Kochel 77 + Zoom (audio muted)
	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 11-23 - F 11-27</b>	<b>Thanksgiving Holiday</b>	Have a peaceful and productive week! See you online. Work on SVG Exercise 2 and project development		
<b>Week 14</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>
<b>M 11-30</b>	XSLT to SVG to make a graph	SVG Exercise 2 (XSLT to SVG)	Fully Remote (Zoom)	Fully Remote (Zoom)
<b>W 12-02</b>	XSLT to SVG work together. Introducing simple JavaScript.	SVG Exercise 3 (options: a timeline?, graph from project data?)	Fully Remote (Zoom)	Fully Remote (Zoom)
<b>F 12-04</b>	Associating JavaScript files with HTML, and coordinating with CSS	JavaScript Exercise 1	Fully Remote (Zoom)	Fully Remote (Zoom)
<b>Week 15</b>	<b>Topics</b>	<b>Do before class</b>	<b>Group A</b>	<b>Group B</b>

<b>M 12-07</b>	JavaScript for projects	JavaScript Exercise 2 (options: Toggling @class attributes, or working on SVG: show/hide). Project sprint	Fully Remote (Zoom)	Fully Remote (Zoom)
<b>W 12-09</b>	Putting it all together: JavaScript with CSS to interact with SVG.	Project sprint. Prepare for Project Checkpoint 6	Fully Remote (Zoom)	Fully Remote (Zoom)
<b>F 12-11</b>	Last day: <b>Project Checkpoint 6:</b> Teams present their projects to the class, invite comments and feedback. Classmates from other teams ask questions, offer commentary on project GitHub repos through early next week.	Prepare to share your project (nearing completion) with the class	Fully Remote (Zoom)	Fully Remote (Zoom)
<b>Finals Week: M 12/14 - F 12/18</b>	<i>Due</i>			
<b>H 12-17</b>	<b>Projects due by 11:59pm</b>  Finish developing projects on newtfire, and send a post to me on GitHub to indicate your team is finished.			