

# CSCI 1300/5001: Introduction to Object-Oriented Programming

(Fall 2020, Dr. Michael Goldwasser)

## Catalog Course Description

An introduction to computer programming based upon early coverage of object-oriented principles such as classes, methods, inheritance and polymorphism, together with treatment of traditional flow of control structures. Good software development practices will also be established, including issues of design, documentation, and testing.

## Student Learning Outcomes

After successfully completing this course, students will be able to:

1. Accurately predict the behavior of small pieces of code authored by others, including use of control structures and interacting objects.
2. Make use of data types and control structures in order to implement high-level behaviors.
3. Write, debug, and document a well-structured program, of at least 100 lines of code, that functions in accordance with desired specifications.
4. Make use of objects from a class defined by someone else (such as built-in string and list classes, or from other language APIs).
5. Implement a user-defined class based upon given specifications, and make use of inheritance to design a subclass of another.
6. Demonstrate an emergent knowledge of recursion through simulation of existing code or implementation of simple recursive functions.

## Outline of Topics

- Variables and Expressions
- User Input/Output
- Basics of Object-orientation
- Interacting with Objects
- Conditional Statements
- Loops
- User-defined Functions
- User-defined Classes
- Good Software Practices
- Inheritance
- Container Classes
- Recursion

## Contents

Course Goals	1
Outline of Topics	1
Course Information	2
Flipped Classroom	3
Technologies	4–5
Grading Policies	6–7
Academic Integrity	8
COVID Considerations	9–11
Student Success	12
Disability Accommodations	12
Title IX	12

## Course Information

### Prerequisites

MATH 1200 (College Algebra) or equivalent, and a C- or better in one of CSCI 1010 through 1090, or equivalent programming experience.

### Class Meetings

The class has four weekly meetings:

Time: Mon/Tue/Wed/Fri, 1:10–2:00pm

Place: Combination of ISE 230 and <https://slu.zoom.us/j/97770602782> (passcode 13005001)  
(Please carefully review page 9 regarding which meetings will be in-person)

We will employ a flipped classroom model for most of the semester (see page 3) and attendance and class participation are expected. The grading policy (see page 6) provides relief for a limited number of absences, and we will provide flexibility given current COVID-19 circumstances (see page 9).

### Staff

**Instructor:** Dr. Michael Goldwasser  
**Email:** [michael.goldwasser@slu.edu](mailto:michael.goldwasser@slu.edu)  
**Web:** [cs.slu.edu/~goldwasser/](http://cs.slu.edu/~goldwasser/)  
**Office:** Ritter Hall 335  
**Phone:** (314) 977-7039  
**Office hours:** Exclusively via Zoom <https://slu.zoom.us/my/goldwasser>  
Mondays 9:00–10:00am  
Tuesdays 3:00–4:00pm  
Fridays 2:10–3:00pm  
or by appointment

**Communication:** There is opportunity for individual attention and mentorship during class meetings and in office hours. Email is also a convenient form of communication. I try to respond to email promptly, including at least once each evening. If you have a question involving progress on a current programming assignment, please include a copy of all relevant source code to allow for more specific feedback.

**Peer Assistants:** In addition to the instructor, we will employ peer assistants, both to provide support for in-class activities and to hold drop-in tutoring hours at various times during the week (see page 12).

### Textbook

We will use a draft revision of *Object-Oriented Programming in Python* by Michael Goldwasser and David Letscher. This draft is freely available through our online course technologies (see page 4).

# Flipped Classroom

## Organization

Throughout the semester, we will employ a “flipped classroom” approach, constructed around a series of lessons from the textbook. Typically, two class meetings will be devoted to an exploration of a lesson, structured as follows. (Note: for some topics, we will choose to vary this structure to use one or three days.)

## Before “Day 1”

Reading assignments will be available well in advance of the introduction of a new topic in class. **By 11:00am of the designated “day 1” for a topic**, students are responsible for having:

- Engaged in the online reading (including community discussion embedded within)
- Successfully completed an online quiz with reinforcement questions
- Completed an online learning survey

## “Day 1” — Initial Exploration

During the first class meeting for a lesson, the instructor will employ a variety of techniques that may include a combination of:

- Providing a mini-lecture on the most meaningful (of confusing) portions of the chapter
- Answering common questions raised by students
- Exploration of source code or live Python demos
- Leading group learning activities

## Between “Day 1” and “Day 2”

There will typically be an individual homework that consists of several questions that require deeper engagement in the new material. The homework will be released alongside the original reading, so that students can consider (and perhaps complete) the questions while doing the reading. However, student will have an opportunity to ask question during “day 1” before completing the homework due at **1:10pm of “day 2”** (i.e., just before class starts).

## “Day 2” — Hands-on Day

For most topics, the goal of the second class meeting will be to provide hands-on experience with new material in a programming environment, and to assess students’ mastery. We will (loosely) employ the technique of “pair programming” having students work together in pairs each day (with those pairs randomly re-assigned for each such class period).

Early in the semester, the 50-minutes will be divided into 40 minutes of pure practice (with an ample supply of challenges/solutions), followed by a 10-minute quiz to be **completed collaboratively by the pair of students**. In the latter part of the semester, larger challenges will serve as 50-minute quizzes.

## Technologies

### Canvas Learning Management System

This semester, we will be piloting **Canvas** as the Learning Management System. You should find an entry for Canvas among the mySLU Tools. This will serve as the primary location of course material and a calendar of meetings and deadlines.

**Warning:** There is currently an incompatibility between Canvas and the Safari browser on OSX, causing embedded images to fail to display. We ask OSX users to use a different browser with Canvas.

### Computer Science Department Computing Systems

Every student in the class will be granted credentials for a collection of computing resources maintained by the Department of Computer Science. Your username will be the same as your SLU Net ID, but the password for our systems is not tied into the university authentication.

Credentials on our department systems will provide you access to:

- Logging into the classroom Linux machines in ISE 230
- Logging into machines in our 24-hour Linux Lab (ISE 216)
- Remote access to `hopper.slu.edu` (our primary student computing server)
- Access to `git.cs.slu.edu` (discussed below)
- Access to `jupyter.cs.slu.edu` (discussed below)

### Python 3 Programming Language

Python is an open-source programming language that is freely-available for all major computing platforms. It is installed on all our department systems described above, but most students prefer to download and install on their personal machines and we encourage you to do.

To install on your own computer, visit [www.python.org/download](http://www.python.org/download); **make sure to download the Python 3.x line** (As of August 2020, Python 3.8.5 is the most recent version). Note that OSX comes pre-installed with Python, but only with the Python 2 distribution.

The standard Python deployment includes a simple Integrated Development Environment named IDLE that we will demonstrate in Lesson 01; most industrial IDEs (e.g. Eclipse, Xcode, VisualStudio) also support Python development.

Finally, there are several web-based environments that support Python programming:

- Our department's `jupyter.cs.slu.edu` server will allow you to develop interactive worksheets that can be saved and retrieved within your department account.
- There is a wonderful web site named `pythontutor.com` that allows you to walk through the execution of a Python program, step-by-step, while providing a helpful visualization of your current workspace.
- The site `repl.it/site/languages/python3` provides a full IDE for developing and executing Python code in the browser.

## The cs1graphics module

The graphics support for this class is provided by a module named `cs1graphics` developed here at SLU. It is not an official part of the Python distribution, so you will need to make sure to have it available on your system as well. If working on our department system, it is already available. If working on your own system, you will need to download a single file named `cs1graphics.py`. The most recent public release of this software is available at [www.cs1graphics.org](http://www.cs1graphics.org).

There are two ways to use the file on your own computer. One is to make sure to place the file in the same directory in which you are working when you start the Python interpreter. A better approach is to install the package by placing the file on your system in a folder typically named `site-packages` that is part of the Python distribution. If you have trouble finding the location of that package, please type the following command while within the Python interpreter.

```
>>> import sys; print([p for p in sys.path if 'site-packages' in p])
```

## Perusall Reading Platform

All of our course readings will be through a website named Perusall. direct links to those readings are already provided through Canvas without need for additional credentials.

A Perusal reading assignment will **remain active until 11:00am on the indicated due date** (typically the morning of “day 1” in our flipped model). You will be able to read the text and may highlight portions of a reading and participate in group discussions about the material. Once a thread is started on a selection, other students and the instructor are able to participate by making further comments, answering questions, or upvoting other people’s questions or answers.

The goal of a reading is to gain initial exposure to fundamental concepts and techniques. We suggest you read each section and carefully work through all of the examples (ideally by trying things out yourself in Python). This initial exposure to the material is a critical step in the learning process and should not be undervalued. You might also do an initial reading a day in advance, and come back to view and respond to other students’ comments.

The Perusall system performs an automated evaluation of your reading engagement (time spent, activity in providing comments/questions, or responding to others). We ask that you keep in mind that the true goal is your learning and not to get too caught up in its scoring system. The system is calibrated to score a student’s engagement on a scale of zero to three, and we will consider a grade of two or higher as successful completion of the assignment. As discussed on page 6, we will also offer other ways to demonstrate your readiness for class.

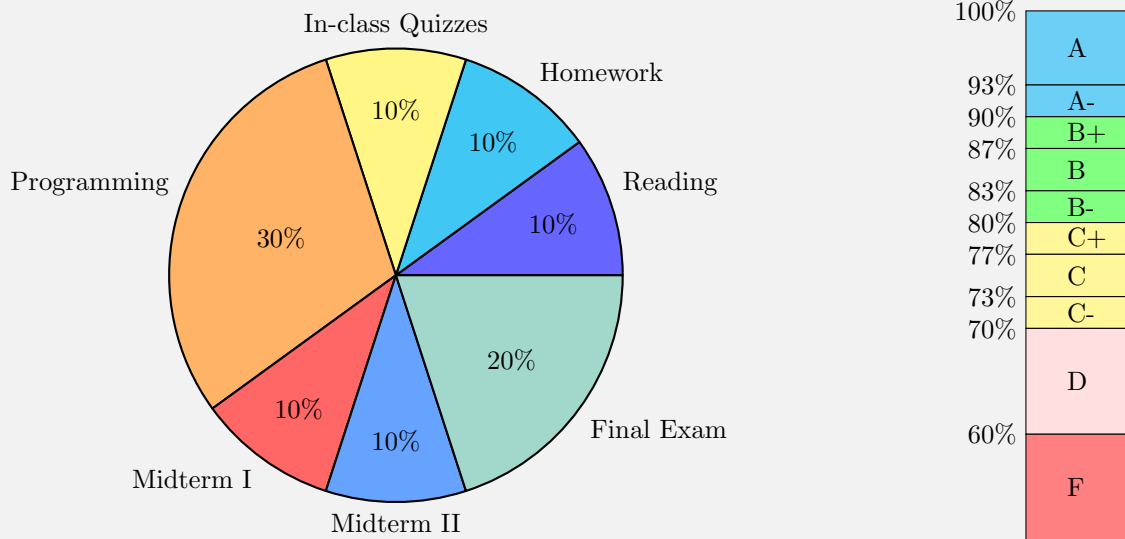
## Git Repositories for Submission of Software and Feedback

To allow the student and instructor to exchange electronic files for programming assignments, we will rely on a version control system known as git. More specifically, our department manages a web-based system available to you at [git.cs.slu.edu](http://git.cs.slu.edu).

## codepost.io for Assignment Feedback

While we will collect many materials through git, we use another system, [codepost.io](http://codepost.io), that allows us to test and annotate your code. While the numeric grades will be included in our gradebook, you are encouraged to view your graded assignments within the codepost system to receive our feedback.

## Grading Policies



**Reading (10%)** A student’s preparation before each “day 1” in our flipped model will be evaluated on a two-point scale, with *three* opportunities to achieve one of those two points:

1 point	Complete reading with score of 2 or higher in Perusall’s system
1 point	Score 75% or higher on the reading quiz
1 point	Complete the learning survey

At the end of the semester, the lowest 5 of these grades will be dropped.

**Homework (10%)** For each class day that is designated as a hands-on day, there will be a pen-and-paper homework set containing exercises from the relevant reading. This homework will be due at the *beginning* of the class meeting and we will not accept late homeworks. At the end of the semester, the lowest 4 of these grades will be dropped.

All of the homeworks must be done independently, and in accordance with the policy on Academic Integrity as given on page 8.

**In-class Quizzes (10%)** Most hands-on days will end with a ten-minute quiz, to be completed and submitted by a pair of students who have been working together during that class period. At the end of the semester, the lowest 4 of these grades will be dropped.

**Programming (30%)** Beyond the in-class work, we expect there to be about 10 more significant programming assignments during the semester. At the end of the semester, the lowest of these grades will be dropped.

These assignments will be submitted electronically through git (details on page 5), and due at 11:59pm on the assigned due date. See page 7 for the policy on late submission. On certain assignments, you will be required to work individually; on others you will be allowed to work in pairs. Please respect the policy on Academic Integrity as given on page 8.

**Midterms (10% each) and Final exam (20%)** Two in-class midterm exams are tentatively scheduled for **September 16** and **October 19** and a remote final exam on **Dec 2, 12:00–1:50pm**.

Final grades will be assigned using the standard scale above. At the discretion of the instructor, this scale might be adjusted, yet only to the benefit of the students.

## Extra Credit

Homework and programming assignments will sometimes include a small extra credit challenge. Please notice, however, that the actual extra credit given for these challenges is relatively insignificant. Students who are concerned about improving their overall grade would be best advised to focus efforts on doing as well as possible on the required work and in preparing for exams.

Our real reason for including these opportunities is to provide some fun and encouragement for students who wish to dig a bit deeper than was required in an assignment. For those students, the chosen extra credit challenges provide a good next step.

## Late Policy

The timing of the various assignments, quizzes, and exams are orchestrated based around the classroom activities and so we will not allow any late submissions of reading assignments or homework assignments.

In-class quizzes and exams must be taken at the regularly schedule time unless advanced arrangements have been made for unavoidable conflicts or subsequently due to emergency situations with appropriate documentation

For the out-of-class programming assignments, we wish to allow students to continue to work comfortably beyond the official deadline when a little more time will result in more progress, while at the same time discourage students from falling significantly behind pace and jeopardizing their success on future assignments. Our solution is the following exponentially decaying late formula (some have suggested that we should offer extra credit to anyone who fully understands this formula).

We will consider an assignment submission “complete” when any part of the assignment is last submitted or modified. Any assignment which is not complete promptly by its due date and time will be assessed a penalty based on the formula  $S = R \cdot e^{-h/173}$ , where  $S$  is the grade given,  $R$  is the grade the work would have received had it been turned in on time, and  $h$  is the amount of time (in hours or fractions thereof) that the work was late. Examples:

- work turned in 1 hour late receives over 99.6% of its original credit
- work turned in 5 hours late receives over 97% credit
- work turned in one full day late receives less than 88%
- work turned in two full days late receives less than 76%
- work turned in five days late receives less than 50%

## Academic Integrity

*Academic integrity is honest, truthful and responsible conduct in all academic endeavors.* The mission of Saint Louis University is “the pursuit of truth for the greater glory of God and for the service of humanity.” Accordingly, all acts of falsehood demean and compromise the corporate endeavors of teaching, research, health care, and community service through which SLU fulfills its mission. The University strives to prepare students for lives of personal and professional integrity, and therefore regards all breaches of academic integrity as matters of serious concern. The full University-level Academic Integrity Policy can be found on the [website of the Provost’s Office](#).

Additionally, each SLU College, School, and Center has its own academic integrity policies, available on their respective websites. All SLU students are expected to know and abide by these policies, which detail definitions of violations, processes for reporting violations, sanctions, and appeals. Please direct questions about any facet of academic integrity to your faculty, the chair of the department of your academic program, or the Dean/Director of the College, School or Center in which your program is housed. Specific College of Arts and Sciences Academic Honesty Policies and Procedures may be found at: [www.slu.edu/arts-and-sciences/student-resources/academic-honesty.php](http://www.slu.edu/arts-and-sciences/student-resources/academic-honesty.php)

In addition to those general statements, we wish to discuss our policy in the context of this course. When it comes to learning and understanding the **general course material**, you may certainly use other reference materials and you may have discussions with other students in this class or other people from outside of this class. This openness pertains to material from the text and practice problems.

However, for **work that is submitted for this course**, you are not to use or search for any direct assistance from unauthorized sources, including but not limited to:

- other texts, books, or solution manuals
- online information other than that referenced by course materials
- other students in this class (except when collaboration is explicitly allowed, as described below)
- students or acquaintances who are not in this course

Acceptable sources of information include consultations with the instructor, teaching assistants, or members of organized tutoring centers on campus, as well as any materials explicitly authorized in an assignment. Even in these cases, if you receive significant help you should make sure to document both the source of the help as well as the extent.

Any violations of these policies will be dealt with seriously. Penalties will apply as well to a student who is aiding another student. Any such violations will result in a minimum penalty of a zero on the given assignment which cannot be dropped, and severe or repeated violations will result in an immediate failing grade in the course. Furthermore all incidents will be reported in writing to the Department and/or the Dean, as per the College procedure.

## Collaboration Policy

On selected portions of assignments for this course, we will explicitly allow students to work in pairs. In this case, conversations between partners is both permissible and required. Furthermore, all students are expected to contribute significantly to the development of the submitted work. It is unethical to allow a partner to “sign on” to a submission if that partner did not significantly contribute to the work.



## COVID-19 Considerations

### Active Learning with Physical Distancing

As discussed on page 3, we are employing a “flipped classroom approach.” In traditional semesters, this involves having you do some preparatory readings and exercises outside of class, and then using our meetings times for active learning to reinforce new lessons. Typically this would include small groups (2–4 people) participating in guided explorations that involve discussion, and on some days pairs of students working side-by-side to complete a challenge together, and with the instructor or teaching assistants often coming to join a discussion or look over your shoulders when answering questions.

The asynchronous preparation outside of class quite easily adapts to our current circumstances. The challenge we face is how we can carry forward with the sort of in-person collaborations among students in the class.

#### **“Day 1” Modality – In-person meetings (with Zoom simulcast for remote students)**

We will make effective use of the “Day 1” meetings in-person. Some of those will be more lecture, or question/answer, and the groupwork on those days will be ungraded and something we can do with 4 students seated in a square with six-foot separation having a discussion.

Remote students, or students who typically attend in-person but are not well enough to attend (see attendance policy on next page), may participate synchronously via Zoom (see link on page 2).

For active-learning activities on such days, we will group remote students with each other through zoom breakout rooms.

We will record all “Day 1” sessions for those who wish to watch asynchronously, though these recordings will not include the active-learning breakout groups.

#### **“Day 2” Modality – Everyone on Zoom**

The “Day 2” framework is entirely built upon close interactions between two students working on practice problems and/or a graded quiz, and typically involving Python code on a computer. There is no effective way to recreate that sharing of a screen and open discussion among partners, as well as coaching from the instructor or teaching assistants, while still maintaining six-foot separation in the classroom.

Instead, we will rely on Zoom for all students on days designated on the schedule as “Day 2” (i.e. those with in-class quiz). We will rely on Breakout Rooms with randomly assigned partners each of those days. Breakout rooms are very well suited for this collaboration as they allow for one person to screen-share with the other (and even offer remote control of the desktop), and the instructor and assistants can easily bounce from room to room, and the room can alert the instructor when they have a question.

#### **Exam Modality TBD**

Our plan is to hold the two midterm exams in-person (with accommodation for remote students), and the final exam will be remote for all students. Further details about the exam formats will be forthcoming.

## Possible Transition to Fully Online Learning

Were SLU to have to pivot away from in-person meetings, or were the instructor to have any short-term or long-term absence, we will switch entirely to synchronous zoom meetings and breakout rooms for all scheduled class meetings.

## Attendance Policy

The health and well-being of SLU's students, staff, and faculty are critical concerns. Accordingly, the following University policy statements on in-person class attendance are designed to preserve and advance the collective health and well-being of our institutional constituencies.

1. Students who exhibit any **potential COVID symptoms** (those that cannot be attributed to some other medical condition the students are known to have, such as allergies, asthma, etc.) shall absent themselves from any in-person class attendance or in-person participation in any class-related activity until they have been evaluated by a qualified medical official. Students should contact the **University Student Health Center** for immediate assistance.
2. Students who exhibit any **potential COVID symptoms** (those that cannot be attributed to some other medical condition the students are known to have, such as allergies, asthma, etc.) but who feel well enough to a) attend the course synchronously in an online class session or b) participate in asynchronous online class activities, are expected to do so. Those who do not feel well enough to do so should absent themselves accordingly.
3. Students (whether exhibiting any of potential COVID symptoms or not, and regardless of how they feel) who are under either an isolation or quarantine directive issued by a qualified health official must absent themselves from all in-person course activity per the stipulations of the isolation or quarantine directive. They are expected to participate in synchronous or asynchronous online class activities as they feel able to do so, or absent themselves accordingly.
4. Students are responsible for notifying each instructor of an absence as far in advance as possible; when advance notification is not possible, students are responsible for notifying each instructor as soon after the absence as possible.
5. As a temporary amendment to the current **University Attendance Policy**, all absences due to illness or an isolation/quarantine directive issued by a qualified health official shall be considered "Authorized" absences (effective August 2020 through May 2021).

## Mask Policy

The University's [Interim Policy on Face Masks](#) governs all students, faculty, staff, and campus visitors in all University-owned, leased, or operated facilities. All persons physically present in any such University facility associated with this course shall comply fully with this policy at all times. Masks must be worn before entry to all such University facilities (as well as outdoors on all University property when six feet of distance is unpredictable or cannot be maintained).

Saint Louis University is committed to maintaining an inclusive and accessible environment. Individuals who are unable to wear a face mask due to medical reasons should contact the Office of Disability Services or Human Resources to initiate the accommodation process identified in the University's ADA Policy. Inquires or concerns may also be directed to the Office of Institutional Equity and Diversity. Notification to instructors of SLU-approved ADA accommodations should be made in writing prior to the first class session in any term (or as soon thereafter as possible).

As the instructor of this course, I shall comply fully with SLU's policy and all related ADA regulations.

Students who attempt to enter a classroom without wearing masks will be asked by the instructor to wear masks prior to entry. Students who remove their masks at any time during a class session will be asked by the instructor to resume wearing their masks.

**Note: Accordingly, no consumption of any food will be allowed in class.**

Students who do not comply with a request by a SLU instructor to wear a mask in accordance with the University's Interim Policy on Face Masks may be subject to disciplinary actions per the rules, regulations, and policies of Saint Louis University, including but not limited to the Student Handbook. Non-compliance with this policy may result in disciplinary action, up to and including any of the following:

- dismissal from the course(s)
- removal from campus housing (if applicable)
- dismissal from the University

To immediately protect the health and well-being of all students, instructors, and staff, instructors reserve the right to cancel or terminate any class session at which any student fails to comply with faculty or staff request to wear a mask in accordance with University policy.

Students are strongly encouraged to identify to their instructor any student or instructor not in compliance. Non-compliance may be anonymously reported via the SLU Integrity Hotline at 1-877-525-5669 (or confidentially via the Integrity Hotline's website at [www.lighthouse-services.com/slu](http://www.lighthouse-services.com/slu)).

## Additional Information

### Supporting Student Success

In recognition that people learn in a variety of ways and that learning is influenced by multiple factors (e.g., prior experience, study skills, learning disability), resources to support student success are available on campus. The Student Success Center assists students with academic-related services and is located in the Busch Student Center (Suite, 331). Students can visit [www.slu.edu/life-at-slu/student-success-center/](http://www.slu.edu/life-at-slu/student-success-center/) to learn more about tutoring services, university writing services, disability services, and academic coaching.

Furthermore, the Department of Computer Science provides drop-in tutoring support for our introductory classes as well as use of our department's Linux computing systems. **For Fall 2020, tutoring hours will be via Zoom.** The schedule for available hours can be found at [cs.slu.edu/resources/tutoring](http://cs.slu.edu/resources/tutoring).

### Disability Accommodations

Students with a documented disability who wish to request academic accommodations must formally register their disability with the University. Once successfully registered, students also must notify their course instructor that they wish to use their approved accommodations in the course.

Please contact Disability Services to schedule an appointment to discuss accommodation requests and eligibility requirements. Most students on the St. Louis campus will contact Disability Services, located in the Student Success Center and available by email at [Disability\\_services@slu.edu](mailto:Disability_services@slu.edu) or by phone at 314-977-3484. Once approved, information about a student's eligibility for academic accommodations will be shared with course instructors by email from Disability Services and within the instructor's official course roster. Students who do not have a documented disability but who think they may have one also are encouraged to contact Disability Services. Confidentiality will be observed in all inquiries.

### Title IX

Saint Louis University and its faculty are committed to supporting our students and seeking an environment that is free of bias, discrimination, and harassment. If you have encountered any form of sexual misconduct (e.g., sexual assault, sexual harassment, stalking, domestic or dating violence), we encourage you to report this to the University. If you speak with a faculty member about an incident that involves a Title IX matter, **that faculty member must notify SLU's Title IX coordinator (or that person's equivalent on your campus) and share the basic facts of your experience.** This is true even if you ask the faculty member not to disclose the incident. The Title IX contact will then be available to assist you in understanding all of your options and in connecting you with all possible resources on and off campus.

Anna Kratky is the Title IX Coordinator at Saint Louis University (DuBourg Hall, room 36; [anna.kratky@slu.edu](mailto:anna.kratky@slu.edu); 314-977-3886). If you wish to speak with a confidential source, you may contact the counselors at the University Counseling Center at 314-977-TALK or make an anonymous report through SLU's Integrity Hotline by calling 1-877-525-5669 or online at [www.lighthouse-services.com/slu](http://www.lighthouse-services.com/slu). To view SLU's policies, and for resources, please visit the following web addresses: [www.slu.edu/about/safety/sexual-assault-resources/index.php](http://www.slu.edu/about/safety/sexual-assault-resources/index.php) and [www.slu.edu/general-counsel](http://www.slu.edu/general-counsel).

IMPORTANT UPDATE: SLU's Title IX Policy (formerly called the Sexual Misconduct Policy) has been significantly revised to adhere to a new federal law governing Title IX that was released on May 6, 2020. Please take a moment to review the new policy and information on the following web address: [www.slu.edu/about/safety/sexual-assault-resources/index.php](http://www.slu.edu/about/safety/sexual-assault-resources/index.php). Please contact the Anna Kratky, the Title IX Coordinator, with any questions or concerns.