

Click on a unit to see the unit content.

Week #	Week of	Units	Due dates	Notes
1	June 5	<ul style="list-style-type: none"> Unit 00: Welcome and setup Unit 01: Computer basics 	Items are due the Sunday END OF DAY of following week	June 5 - First day of classes for summer session
2	June 12	<ul style="list-style-type: none"> Unit 02: Python basics 	Items are due the Sunday END OF DAY of following week	June 12 - Last day to drop an 8-week course and receive a refund
3	June 19	<ul style="list-style-type: none"> Unit 03: Storing data in lists 	Items are due the Sunday END OF DAY of following week	
4	June 26	<ul style="list-style-type: none"> Unit 04: Practicing the basics 	Items are due the Sunday END OF DAY of following week	
5	July 3	<ul style="list-style-type: none"> Unit 05: Control flow - branching and looping 	Items are due the Sunday END OF DAY of following week	
6	July 10	<ul style="list-style-type: none"> Unit 06: The INs and OUTs of functions 	Items are due the Sunday END OF DAY of following week	
7	July 17	<ul style="list-style-type: none"> Unit 07: Object Oriented Programming basics 	Items are due the Sunday END OF DAY of following week	July 17 - Last day to withdraw with a "W"
8	July 24	<ul style="list-style-type: none"> Unit 08: Finishing everything up 	This week's items are due on THURSDAY, JULY 27th.	July 27 - Last day of summer session
9	July 31	<ul style="list-style-type: none"> POST-SEMESTER; grades posted online by Aug 2 at noon 		July 31 - Grades entered online by professors by 5 pm



Unit 00: Welcome and setup (Week 1, June 5)



Welcome to CS 134!





Starting a new class always takes a bit of learning, so hopefully this can help guide you through what to do during this first week. If you have any questions, please post in the discussion board for the unit (e.g., [Unit 0 help - Welcome and setup](#) so that other students can find answers to problems they might have, too! If you have something you'd like to discuss with me privately, please email me via Canvas.

Setup and Syllabus

Starting off, you should make sure to go through the [SYLLABUS](#) and the [Setup Replit](#) assignments. If you have any questions about course policy or the tools we're going to use, please let me know!

There's also an [Introductions!](#) discussion board where you can introduce yourself to your classmates.

First week of class!

- Read through the  [SYLLABUS](#)
- Set up the REPLIT website:  [Setup Replit](#)
- Introduce yourself to your classmates:  [Introductions!](#)
- Need help? Post in:  [Unit 0 help - Welcome and setup](#)
- Don't forget to continue down to [Unit 01: Computer basics](#) and complete that during the first week as well!







Unit 01: Computer basics (Week 1, June 5)



Before we get into coding proper, let's get some background knowledge on computers and software...

Part 1: A bit about computers...

These *Concept Introduction* assignments are meant to introduce you to new concepts in an interactive way. Please work through these first. You can retake these assignments as many times as you'd like.

-  [\[CS134.U01.CI1\]](#)  [What's the point of computer programs?](#)
(Due Sunday, June 11th)
-  [\[CS134.U01.CI2\]](#)  [How do computers store data?](#)
(Due Sunday, June 11th)

Part 2: Watch this!

Why is Python so popular, anyway? Learn a little more about it in this video!

-  [What is Python? Why Python is So Popular? \(Programming with Mosh\)](#)





Part 3: Read this!

More background knowledge to build out your programming foundation!

-  [Why should you learn to write programs? \(Python for Everybody, Chapter 1\)](#)



Part 4: Practicing

In this exercise you will practice stepping through code as if you're the computer. You'll also learn a bit about flowcharts!

-  [\[CS134.U01.EX\]](#)  [Program design basics - Exercise](#)
(Due Sunday, June 11th)
-  [\[CS134.U01.PR\]](#)  [Program design basics - Peer Review](#)
(Due Wednesday, June 14th)

Part 5: Tech literacy

These assignments are additional information about technology in general, not strictly related to programming.

-  [\[CS134.U01.TL\]](#)  [How computers and software work](#)
(Due Sunday, June 11th)



Unit 02: Python basics (Week 2, June 12)



This week we're going to begin working with Python and programming basics: getting input from the user, writing output to the screen, doing basic arithmetic operations, and working with variables. Make sure to go over the Concept Introductions to learn about these items, as well as going over the reading assignment (Chapter 2).

Part 1: Working with variables... (video)

This video is from last year, so the Canvas page looks different but the topics are still the same!

-  [Unit 2 introduction](#)
-  [Unit 2 exercise step-thru](#)
- [REPLIT Example code: Math program](#)
- [REPLIT Example code: Books in stock](#)

Part 2: Review questions

As you go through this unit's content, make sure you can answer the following things!

- [Unit 02 Review Questions](#)



Part 3: Read this!

Learn about variables, values, and expressions in Python!

-  [Chapter 2: Variables, expressions, and statements \(Python for Everybody\)](#)

Part 4: Review concepts

Work through the concept introduction assignments to review what you've learned so far.

-  [\[CS134.U02.CI1\] NEW Python basics - Variables](#)
(Due Sunday, June 18th)
-  [\[CS134.U02.CI2\] NEW Python basics - Input and output](#)
(Due Sunday, June 18th)

Part 5: Practicing coding

Let's get into Replit and get some hands-on practice with variables!

-  [\[CS134.U02.EX\] !\[\]\(6c6f20642b351a420d854c876275f471_img.jpg\) Python basics - Exercise](#)


Unit 03: Storing data in lists (Week 3, June 19)



This week we're talking about storing lots of data grouped together in a list. Please make sure to watch the following video for more information on lists as well. The book talks about strings as well, so we will also highlight that.

Part 1: Working with lists (video)

This video is from last year, so the Canvas page looks different but the topics are still the same!

-  [Unit 3 introduction](#)
- [REPLIT Example code: List basics](#)
- [REPLIT Example code: Contact List](#)
- [REPLIT Example code: Fancy list operation](#)

Part 2: Review questions

As you go through this unit's content, make sure you can answer the following things!

- [Unit 03 Review Questions](#)

Part 3: Read this!

Skip the parts on "parsing lists", "objects and values", "aliasing", "list arguments"

- [Chapter 8: Lists \(Python for Everybody\)](#)

Part 4: Review concepts

Review what you've learned about lists so far...

- [\[CS134.U03.CI\] NEW Lists - Introduction](#)
(Due Sunday, June 25th)

Part 5: Practicing

Programming with lists

- [\[CS134.U03.EX\] 🏆 Storing data in lists - Exercise](#)
(Due Sunday, June 25th)
- [\[CS134.U03.PR\] 🔍 Storing data in lists – Peer Review](#)
(Due Wednesday, June 28th)



Unit 04: Practicing the basics (Week 4, June 26th)



This week is to review the past few units and help you get more practice with variables and lists!

Part 1: Review

This exercise will be a review over print, variables, and lists.

- [\[CS134.U04.EX\] 🏆 Practicing the basics – Exercise](#)
(Due Sunday, July 2)

Part 2: Tech literacy

A bit about the roles of UI and UX designer

- [\[CS134.U04.TL\] 🎯 Careers – UI/UX Designer - Tech Literacy](#)
(Due Sunday, July 2)

Part 3: Check in

Let me know how the unit went for you! Have any questions?

- [\[CheckIn4\] 🏆 Check-in](#)

End of the week!

Remember that if you have any questions you can post it on the discussion board, or join the course Discord server where you can contact other classmates and the instructor.

- [Discussion Boards](#)
- [Discord invite: https://discord.gg/jj7U6HtVeh](#)



Unit 05: Control flow - Branching and looping

(Week 5, July 3rd)



This week we are learning about control flow - how *what* a program executes and how many times depends on some *condition* we set.

Part 1: Read this!

Make sure to read up on how conditionals (if statements) and iterations (while loops, for loops) work!

- [Chapter 3: Conditionals \(Python for Everybody\)](#)
- [Chapter 5: Iterations \(Python for Everybody\)](#)

Part 2: Example coding videos

Watch these videos of example coding to learn a little more about how control flow works.

- [Conditionals](#)
 - [REPLIT Example code: If statement](#)
 - [REPLIT Example code: If/else statement](#)
 - [REPLIT Example code: If/elif statement \(shopping cart v1\)](#)
 - [REPLIT Example code: and, or, not operators](#)
 - [REPLIT Example code: for loop](#)
 - [REPLIT Example code: while loop](#)
 - [REPLIT Example code: list, for loop \(shopping cart v2\)](#)
 - [REPLIT Example code: racecar](#)

Part 3: Review concepts

Review the concepts you're learning about...

- [\[CS134.U05.C11\] NEW Branching basics - Introduction](#)
(Due Sunday, July 9th)
- [\[CS134.U05.C12\] NEW Looping - Introduction](#)
(Due Sunday, July 9th)

Part 4: Review questions

As you go through this unit's content, make sure you can answer the following things!



Unit 06: Functions

(Week asdf, July 10th)



Part 1: Read this

Functions are a way we can delegate tasks elsewhere, under a "named" section of code.

- [Chapter 4: Functions \(Python for Everybody\)](#)
- [Review questions \(FUNCTIONS\)](#)

Part 2: Example coding videos


Watch these videos of example coding to learn more about functions.

- [Functions](#)
 - [REPLIT Example code: no input / no output](#)
 - [REPLIT Example code: no input / yes output](#)

- [REPLIT Example code: yes input / no output](#)
- [REPLIT Example code: yes input / yes output](#)
- [REPLIT Example code: Menu program with multiple functions](#)



Part 3: Review concepts

Review the concepts you're learning about...

-  [\[CS134.U06.CI\] NEW Functions – Introduction](#)
(Due Sunday, July 16th)

Part 4: Practicing

Practice working with functions

-  [\[CS134.U06.EX\] 🏆 Functions - Exercise](#)
(Due Sunday, July 16th)
-  [\[CS134.U06.PR\] 🗣️ Functions – Peer Review](#)
(Due Sunday, July 16th)

Part 5: Tech literacy

Continuing to learn about different roles in tech



Unit 07: Object Oriented Programming basics (Week 7, July 17th)




Part 1: Read this

You only need to read: "Our first Python object", "Object lifecycle", "Multiple instances"

-  [Chapter 14: Objects \(Python for Everybody\)](#)
- [Review questions \(CLASSES/OBJECTS\)](#)


Part 2: Example coding videos

Writing classes in Python



-  [Classes](#)
 - [REPLIT Example code: Basic class](#)
 - [REPLIT Example code: List of Products](#)
 - [REPLIT Example code: Store program](#)

Part 3: Review concepts

Reviewing object oriented programming concepts

-  [\[CS134.U07.CI\] NEW Object oriented programming – Introduction](#)
(Due Sunday, July 23rd)

Part 5: Practicing

-  [\[CS134.U07.EX\] 🏆 Object oriented programming - Exercise](#)
(Due Sunday, July 23rd)
-  [\[CS134.U07.PR\] 🗣️ Object oriented programming - Peer Review](#)
(Due Sunday, July 23rd)

Part 6: Tech literacy

-  [\[CS134.U07.TL\] 🧠 Careers - DevOps, Business Analyst, Product Manager - Tech Literacy](#)

Part 7: Check in

Let me know how the unit went for you! Have any questions?





Unit 08: Finishing everything up (Week 8, July 24th)



asdfasdf

Part 1: Tech literacy

It can be easy to get rusty with programming concepts between semesters. Here you can brainstorm with your classmates on little programming projects you might try out!

-  [\[CS134.U10.TL\]](#)  [Practicing between semesters - Tech Literacy](#)

Part 2: Catch up!

No new topics... just get everything turned in by the course end date (July 27th!)

Part 3: Check in and Course Evaluation

It's the end of the semester! Time to give me a final rating. What did I do well at? What could be improved? :)

You can also leave anonymous feedback in the JCCC Course Evaluation!

-  [\[CheckIn8\]](#)  [Check-in](#)
-  [JCCC Course Evaluation](#)

End of the semester!

We're done!



Unit asdfasdf: asdfasdf (Week asdf, June asdf)



asdfasdf

Part 1: A bit about computers...

asdfasdf

Part 2: Review questions

As you go through this unit's content, make sure you can answer the following things!

- [Unit asdfasdf Review Questions](#)

Part 3: Read this!

asdfasdf

Part 4: Review concepts

asdfasdf

Part 5: Practicing

asdfasdf

Part 6: Tech literacy


asdfasdf

Part 7: Check in

Let me know how the unit went for you! Have any questions?

End of the week!

Remember that if you have any questions you can post it on the discussion board, or join the course Discord server where you can contact other classmates and the instructor.

-  [Discussion Boards](#)
- [Discord invite: https://discord.gg/jj7U6HtVeh](https://discord.gg/jj7U6HtVeh)

Course info

Course

CS 134: Programming Fundamentals

Section

354, CRN 60789, Online only

Semester

Summer 2023

Dates

06/05/2023 - 07/27/2023

Instructor

R.W. Singh (they/them)

Email

rsingh13@jccc.edu

Office

RC 348H (Not on campus during Summer, except by appointment)

Links

Course Discord

<https://discord.gg/jj7U6HtVeh>

Course status

[Grading/prep status](#)[Book & Lectures](#)[Course Policies](#)[Quick Search](#)[Coding Style Guide](#)[JCCC Course Catalog](#)