

CS148 Homework 0

Homework Due: June 25th at 11:59 PM PST (Recommended)
Quiz Date: See HW1 for details

Welcome to CS148! This is homework 0, meant to make sure you have everything installed and ready to go when the actual assignments roll around.

This setup should only take 1-2 hours at most. **We strongly recommend finishing this assignment before June 25 (Saturday)**, but taking into account late enrollment and some flexibility with start-of-quarter hecticness, it's fine to submit this anytime before the homework 1 deadline on July 2 (Saturday).

1 Introduction

1.1 Homework overview

In this course, weekly assignments will be split into a **technical component** and a **practical component**. Our goal is to have you understand the technology and then learn how to utilize off-the-shelf implementations of it to create nice images.

- For the technical component, you will be mostly working in python / jupyter notebooks.
- For the practical component, you will learn your way around Blender.
- To submit your assignments, you will run a generate a pdf (via a script or jupyter) and upload to gradescope.

1.2 Piazza and Honor Code

If you have any questions about any of these installation/setup steps, please ask in a Piazza post. Additionally, we encourage you to make the posts public if it is a question that may benefit other students, but please do not share code snippets and/or any solutions to the assignments in public posts or on your own public github repositories, as sharing code is in direct violation of the Stanford Honor Code. (Please refer to [this CS department article](#) for more details!)

1.3 Collaboration Policy

You are allowed to (and encouraged to) discuss the homework with other students, but please do not share explicit code. We encourage you to find a partner to work on the homeworks and subsequently on the final project with together – see the pinned Piazza post for more details. If you work with a partner, the code that you submit must be your own, but feel free to share assets and to work on the practical component together. The intention is to build upon the practical components of the assignments for your final image.

1.4 Late Assignments

Requests for extensions will only be approved for extenuating circumstances or as required by university policy. Note that with only 8 weeks to complete 6 assignments and a final project, we

really don't recommend delaying submissions. If you're having trouble catching up, reach out to us via a private Piazza post and we can try to figure something out!

2 Assignment Outline

This assignment is simply a setup step – no coding or active Blender work is required. The assignment is broken down into several sections that we recommend you complete in order:

- Homework download
- Blender installation and intro to Blender
- Python environment setup and intro to notebooks
- Gradescope Submission

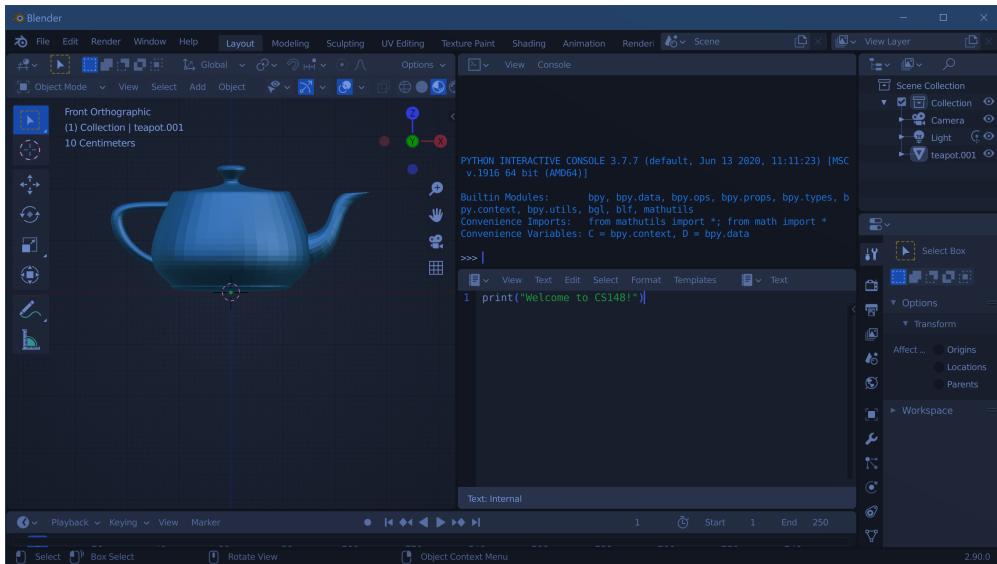
3 Homework download

Before we start, please download **HW0.zip**, save it in some course directory **\$CS148_DIR** on your machine, and unzip the file in this directory. We'll be releasing one such zip file every week. After extracting **HW0.zip**, take a brief look at the contents. We assume for the following sections that **\$CS148_DIR/HW0/assets**, **\$CS148_DIR/HW0/src**, **\$CS148_DIR/HW0/images** are directories that now exist on your machine.

4 Blender

What is Blender? Blender is a free and open-source 3D creation suite. It supports the entirety of the 3D pipeline—modeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation.

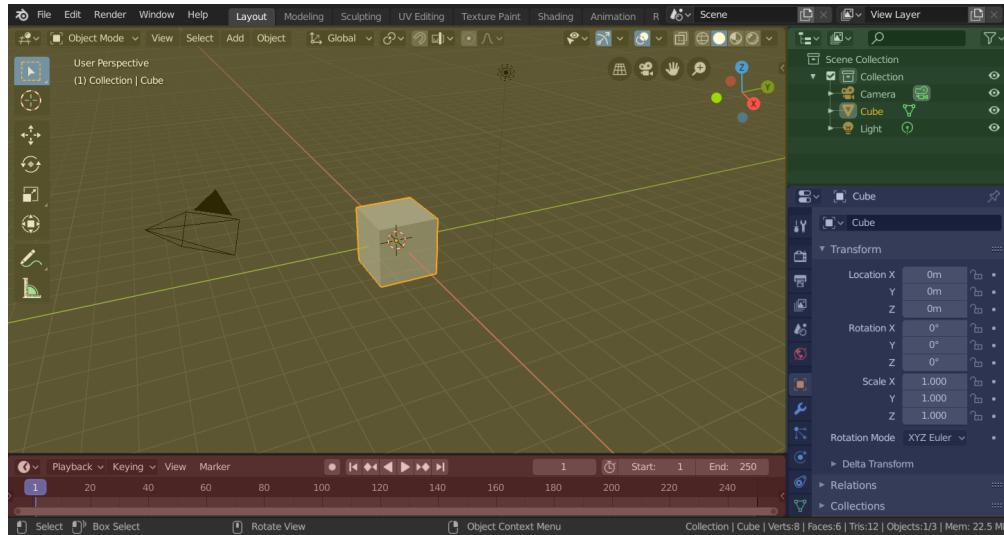
- Follow the steps on [the blender website](https://www.blender.org/download/releases/2-93) (<https://www.blender.org/download/releases/2-93>) to download Blender 2.93 LTS for your machine. Pay close attention to the version (this is not the latest release!)
- Launch the program, and open up **\$CS148_DIR/HW0/assets/hw0.blend** – you should see a scene that looks somewhat like this:



- **Action:** Take a screenshot of this scene, and save it as `$CS148_DIR/HW0/images/blender_screenshot.png`

4.1 Intro To Blender

Close the project that you just opened, and create a new project by clicking **File → New → General** on the upper left of the screen where the top bar is. You should now see a default scene with a cube, a light, and a camera:



This is the default workspace for blender (see more at [blender documentation](#)), with:

- 3D Viewport on the top left (yellow)
- Outliner on the top right (green)
- Properties editor on the bottom right (blue)
- Timeline on the bottom left (red)

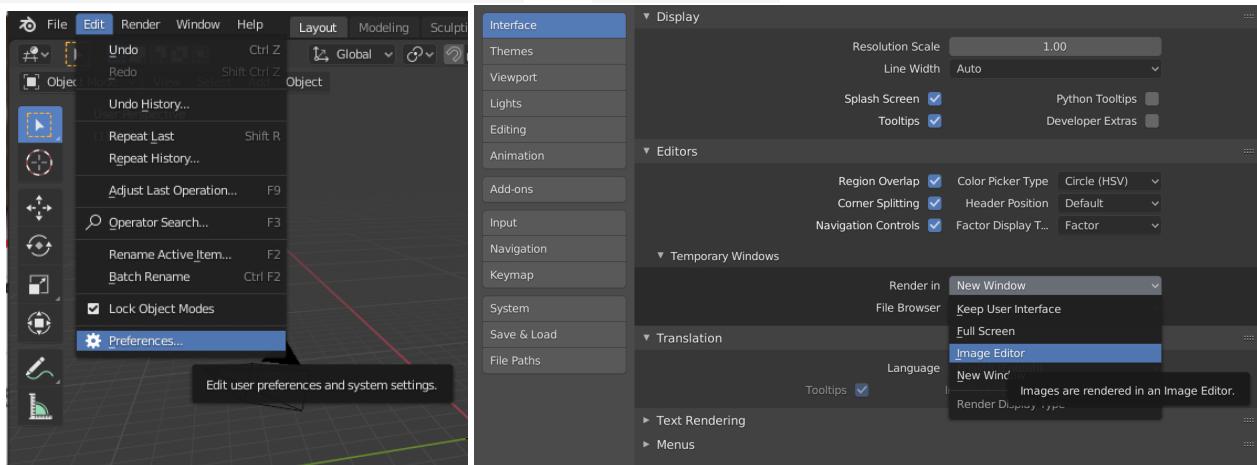
Blender has a nice [official introduction playlist](#) for 2.8 on their youtube channel which is compatible with 2.93. If you haven't used Blender before, we suggest watching the first two videos to learn the basics of how to get around inside Blender. But don't worry, we will still provide detailed instructions below to help you get started. We will also include some links to Blender documentation along the way if you want to further investigate the tool.

4.1.1 Editor Setup

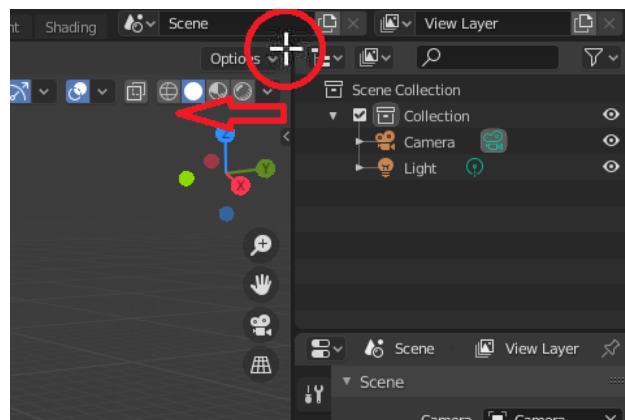
We will first add an Image Editor to our workspace. This will be where we see our rendered results inside Blender. It will be more convenient than saving your image each time you make changes to the scene, going to your file explorer, then opening the rendered image. Setting up the editor will keep the render in the same window, without popping out a new window every time.

Go to the top bar, and click on **Edit → Preferences**. A new window should pop up. On the sidebar of this window, select **Interface**, then change the option for

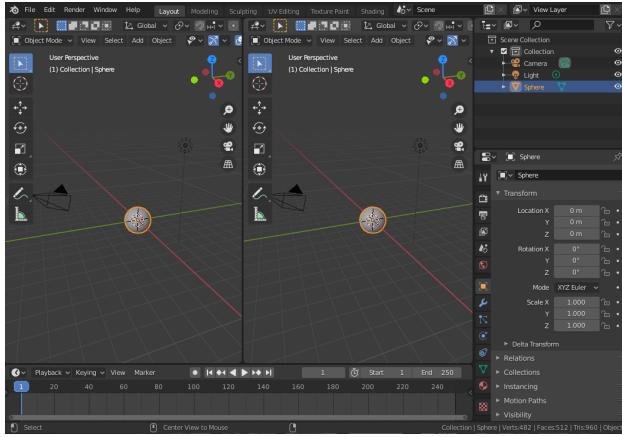
Editors → Temporary Window → Render into **Image Editor**.



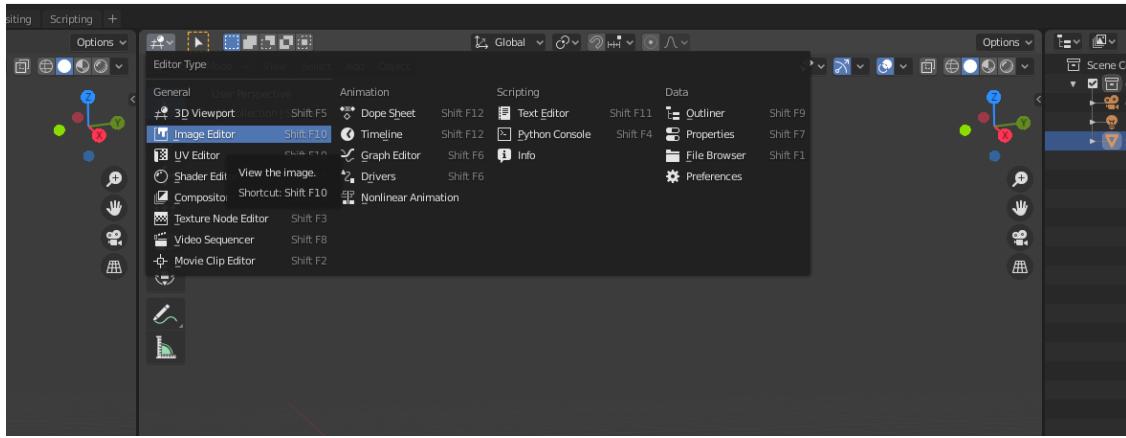
Hover the mouse over the top right corner of the 3D viewer. The cursor will change to a cross.



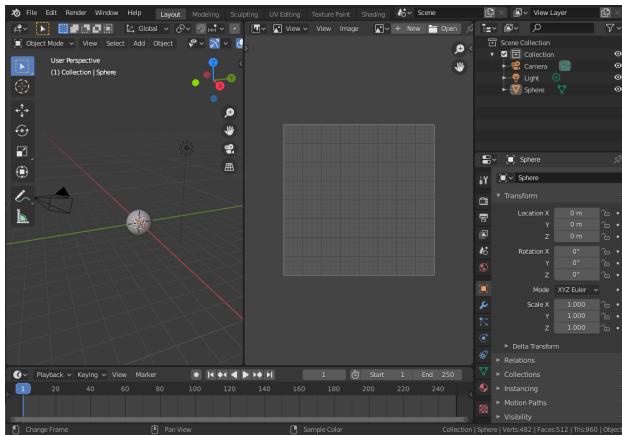
Drag left to create a new area.



In the top left of the new area, change the editor type to Image Editor.



Your screen should now look like this:

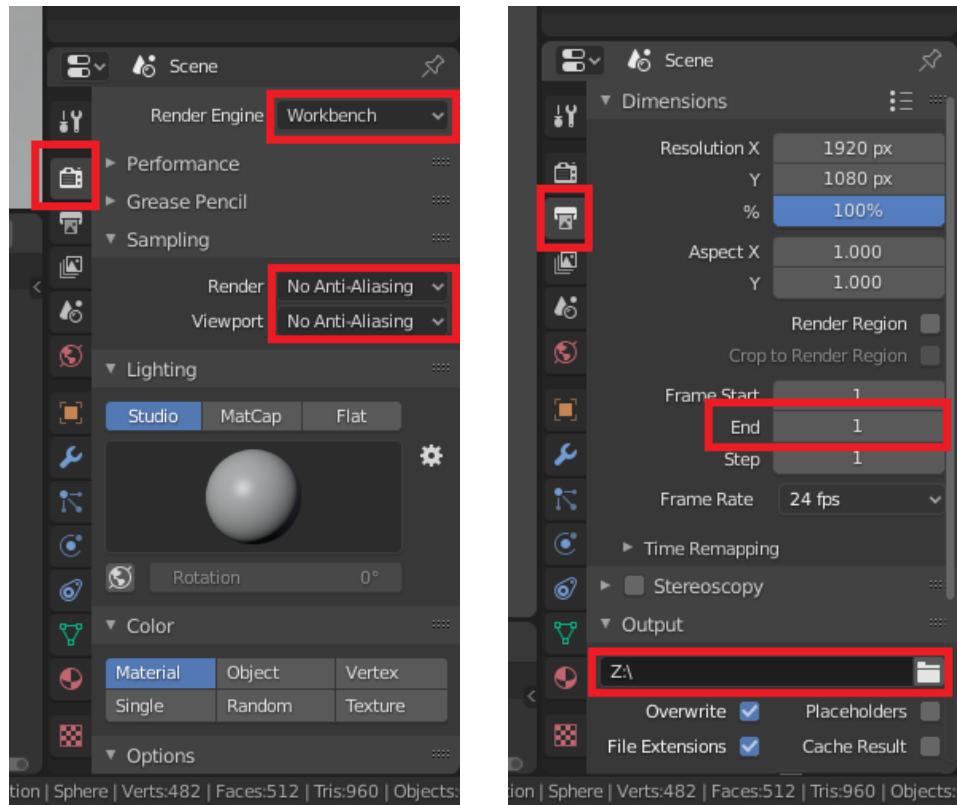


For more detail about areas and how to manipulate them, we optionally recommend [the tutorial video](#).

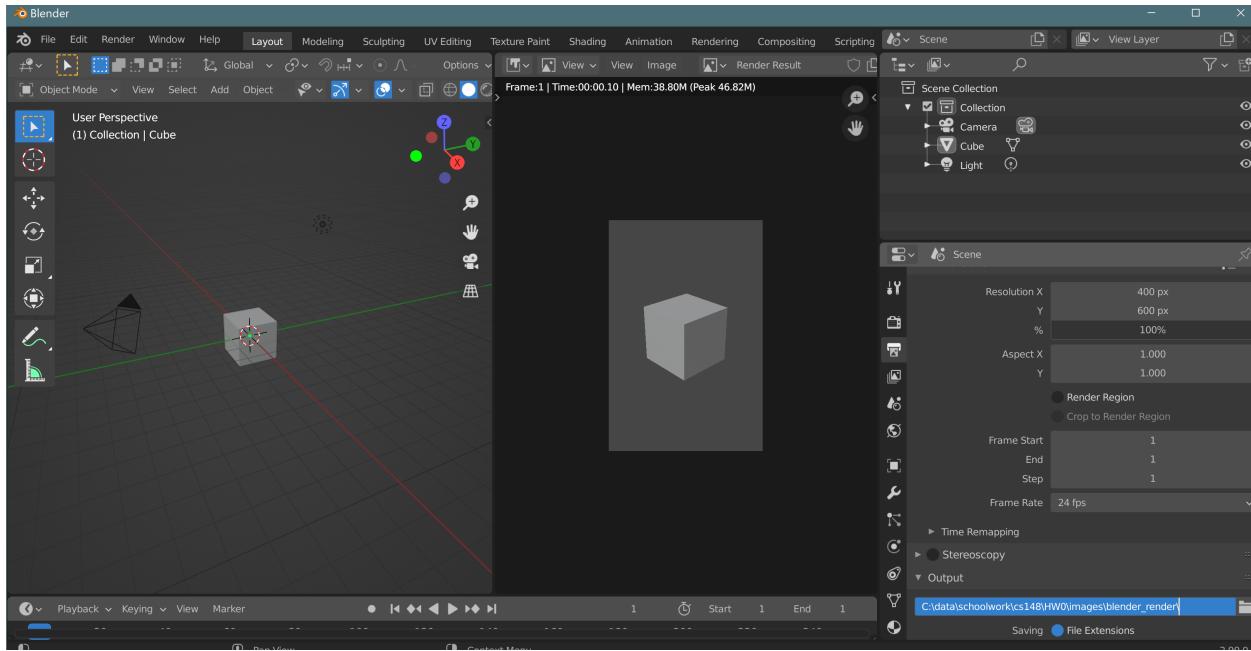
4.1.2 Render and Save Image

We can now change the render settings and render the image. By default Blender only saves animations, so here we change the length of the animation to 1. Also, we are playing with the viewport scanline renderer for this homework, so we will use the Workbench render engine.

Go to the Properties Editor (bottom right), select the camera icon (Render Properties) on the sidebar. Change the **Render Engine** to **Workbench**, and the **Sampling** for Render and Viewport to **No Anti-Aliasing**. (We will talk about what this all means later!) Then click on the printer icon (Output Properties) on the sidebar. Change **Frame End** to **1**, the Output Path to **\$CS148_DIR/HW0/images/blender_renders/**, and the resolution to **X=400px,Y=600px**.



Now render the image by going to **Render → Render Animation** on the top bar (or shortcut Ctrl/Cmd F12) to render the image, and save it to the specified path automatically. The rendered image will also show up as "Render Result" in the Image Editor we created in the previous step.



Now you have rendered your first image in Blender! Note that this image is rendered from the default camera's perspective.

Action: Check that the rendered image of the cube is saved correctly to

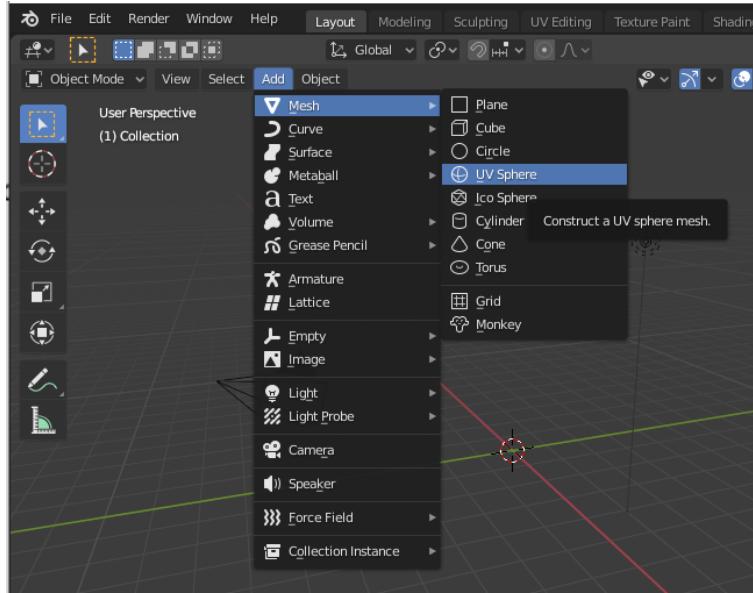
\$CS148_DIR/HW0/images/blender_renders/0001.png

4.1.3 Scene manipulation

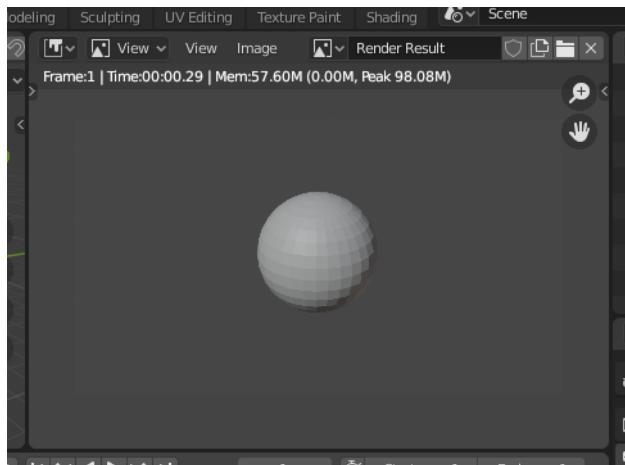
The default scene comes with a cube. We can add/delete objects in the scene. Blender stores some basic objects that we can directly add to the scene without constructing them from scratch. Here we will delete the cube and add a UV sphere.

Move your mouse over the cube, left click to select. An orange outline will appear around the cube, indicating it's being selected. Hit the Delete key to delete the cube.

Then go to the menu bar on top, and navigate to **Add → Mesh → UV sphere**. A sphere should appear in the center of the scene.



Action: Press **Ctrl/Cmd F12** (or **Render → Render Animation**) again to update the render. You should see a sphere instead of a cube now. Check that the rendered image of the sphere is saved correctly to **\$CS148_DIR/HW0/images/blender_renders/0001.png**, having overwritten the render of the cube.



4.2 Python Environment Setup

- Follow the steps on [the anaconda website](https://www.anaconda.com/products/distribution) (<https://www.anaconda.com/products/distribution>) to setup anaconda on your machine.
- If you've never used anaconda before, read through [this short starter guide](#) to understand what conda environments are.
- Now, in your terminal of choice and/or the anaconda prompt, create a cs148 conda environment by running the following in order:

```
conda create -n cs148_env python=3.9
```

```
conda activate cs148_env
```

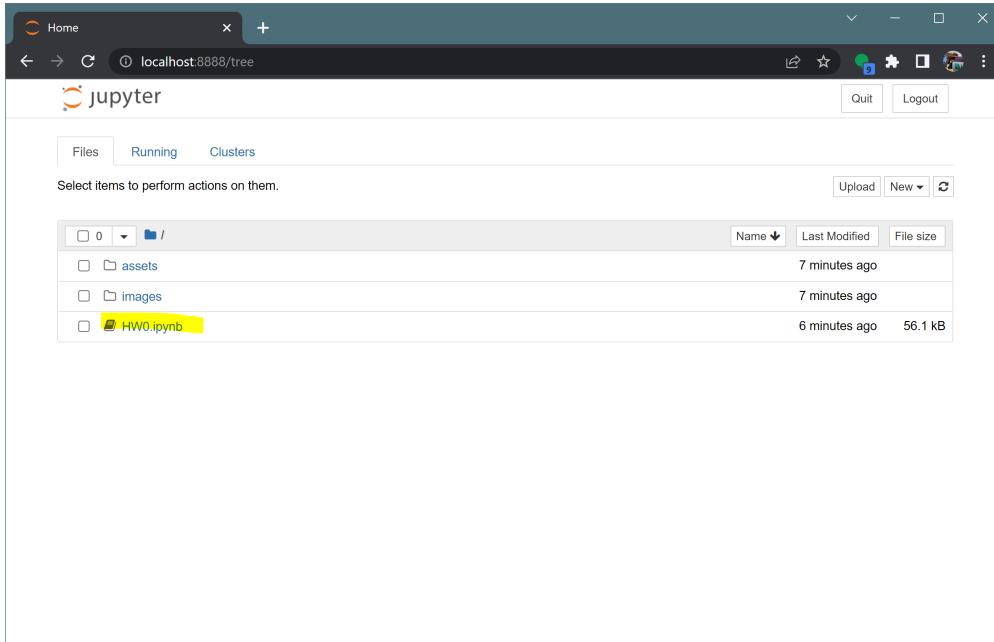
- In the terminal/prompt, make the HW0 directory your working directory, install the python package requirements, and launch a local jupyter notebook server with the following 3 steps in order:

```
cd $CS148_DIR/HW0 (Or, dir $CS148_DIR\HW0 if you're on Windows)
```

```
pip install -r src/requirements.txt
```

```
jupyter notebook --browser chrome --port 8888 .
```

- Now, open up your Chrome browser and type `localhost:8888` in the URL field, and you should see the following:



Note that you may need to follow prompts to setup a password/security token depending on your machine's security settings. Reach out to us on Piazza if you need help here.

4.3 Jupyter Notebooks

- Open up `HW0.ipynb`. You should see something like this:

The screenshot shows a Jupyter Notebook interface with the title "Jupyter HW0 Last Checkpoint: 03/14/2022 (unsaved changes)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The toolbar has buttons for back, forward, search, and run. A Python 3 (ipykernel) kernel is selected. The notebook content starts with assignment instructions:

Assignment 0 is meant to ensure you have your development environment setup for this course. The deliverables are to submit a pdf of this notebook onto Gradescope. The To-Dos of this assignment are as follows:

1. Install Blender and generate a render of a sphere
2. Install Anaconda and setup a python environment
3. Launch jupyter and see this notebook
4. Run this notebook and make sure no errors occur
5. Answer the 3 questions at the end of this notebook
6. Generate a pdf (link to instructions) and submit the pdf to Gradescope

Note: Do not edit any of the cells that have "Do Not Edit" on them!

In [1]:

```
#DO NOT EDIT
import os, sys
import cv2
from matplotlib import pyplot as plt

%load_ext autoreload
%autoreload 2

def show_image(filename:str, height:float=10, width:float=10):
    assert os.path.isfile(filename)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    plt.gcf().set_size_inches((width,height))
    plt.gca().grid(None)
    plt.gca().axis('off')
    plt.imshow(img)
    plt.show()
```

Before you execute/run the 2 next cells, make sure you have saved the blender screenshot to `images/blender_screenshot.png` and your first render to `images/blender_render/0001.png`.

In []: #DO NOT EDIT
show_image("images/blender_screenshot.png",height=6,width=10)

In []: #DO NOT EDIT
show_image("images/blender_render/0001.png",height=6,width=4)

Short Responses

Please fill out the answers to these questions -- 1-2 sentences for each should suffice.

- **Action:** Run the code cells in order, using either the Run button or CTRL+Enter after clicking on and highlighting the code cell. After you execute a cell, numbers should appear on the left side of each cell in order of execution. Make sure the screenshot you took earlier of Blender, and the render of the sphere, appears correctly.

The screenshot shows a Jupyter Notebook interface with the title "Jupyter HW0 Last Checkpoint: 03/14/2022 (unsaved changes)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The toolbar has buttons for back, forward, search, and run. A Python 3 (ipykernel) kernel is selected. The notebook content starts with assignment instructions:

Assignment 0 is meant to ensure you have your development environment setup for this course. The deliverables are to submit a pdf of this notebook onto Gradescope. The To-Dos of this assignment are as follows:

1. Install Blender and generate a render of a sphere
2. Install Anaconda and setup a python environment
3. Launch jupyter and see this notebook
4. Run this notebook and make sure no errors occur
5. Answer the 3 questions at the end of this notebook
6. Generate a pdf (link to instructions) and submit the pdf to Gradescope

Note: Do not edit any of the cells that have "Do Not Edit" on them!

In [1]:

```
#DO NOT EDIT
import os, sys
import cv2
from matplotlib import pyplot as plt

%load_ext autoreload
%autoreload 2

def show_image(filename:str, height:float=10, width:float=10):
    assert os.path.isfile(filename)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    plt.gcf().set_size_inches((width,height))
    plt.gca().grid(None)
    plt.gca().axis('off')
    plt.imshow(img)
    plt.show()
```

Before you execute/run the 2 next cells, make sure you have saved the blender screenshot to `images/blender_screenshot.png` and your first render to `images/blender_render/0001.png`.

In [2]: #DO NOT EDIT
show_image("images/blender_screenshot.png",height=6,width=10)

A screenshot of the Blender interface is shown, displaying a 3D model of a teapot.

In []: #DO NOT EDIT
show_image("images/blender_render/0001.png",height=6,width=4)

- Action:** Fill in the text questions by double clicking on the text cell.



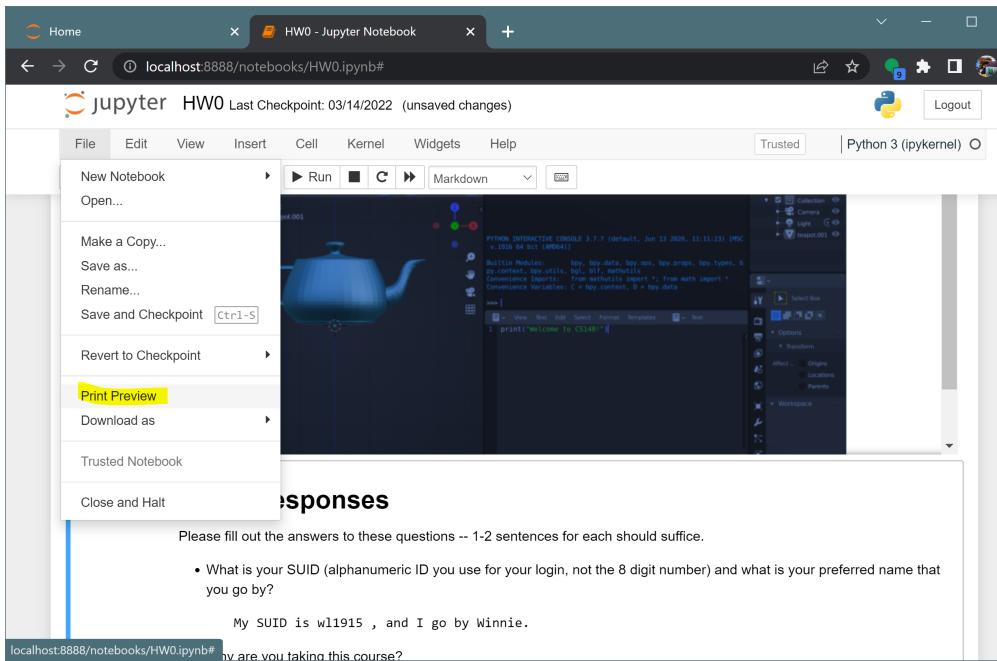
```
In [3]: #DO NOT EDIT
show_image("images/blender_render/0001.png",height=6,width=4)
```

Short Responses

Please fill out the answers to these questions -- 1-2 sentences for each should suffice.

- * What is your SUID (alphanumeric ID you use for your login, not the 8 digit number) and what is your preferred name that you go by?
My SUID is wl1915 and I go by Winnie
- * Why are you taking this course?
TODO: fill in your answer here
- * What do you hope to learn from this course?
TODO: fill in your answer here

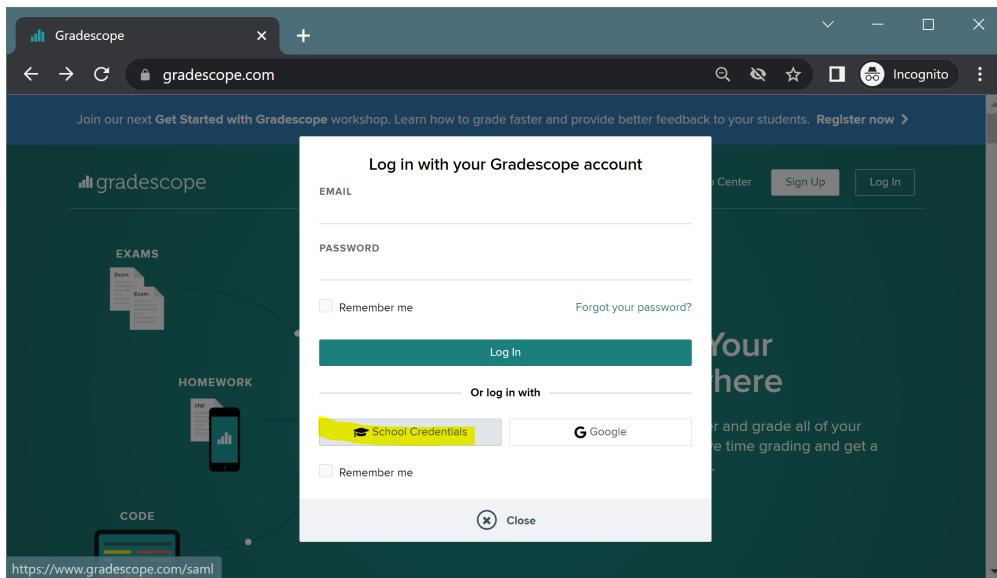
- Action:** Save the notebook as a pdf: Use the print preview button to generate a clean HTML page, then use CTRL+P to print and save the page as a PDF.



If you've never used jupyter notebooks before or find these instructions confusing, we recommend taking a look at the “Running Code” and “Markdown Cells” sections of [the official documentation](#) first, or watching a tutorial [such as this one](#). Feel free to stop by office hours if you still have questions.

4.4 Gradescope Account

Log into [Gradescope](#) using the “school credentials” option. Find “Stanford University Network ID” in the list of schools, and login using your SUNet ID. (If you’ve been enrolled in the course for more than 24 hours, CS148 should show up in your dashboard. If you don’t see the course in Gradescope and you’ve been officially enrolled, reach out to us via a private Piazza post.)



Action: Submit your pdf to the HW0 assignment.

5 Submission Checklist

By the end, you should have uploaded a pdf of the jupyter notebook to gradescope. The pdf should contain the following items:

- A screenshot of the pre-existing blender project (with a teapot)
- A render of a grey sphere that you generated
- Short answers to the free-form questions