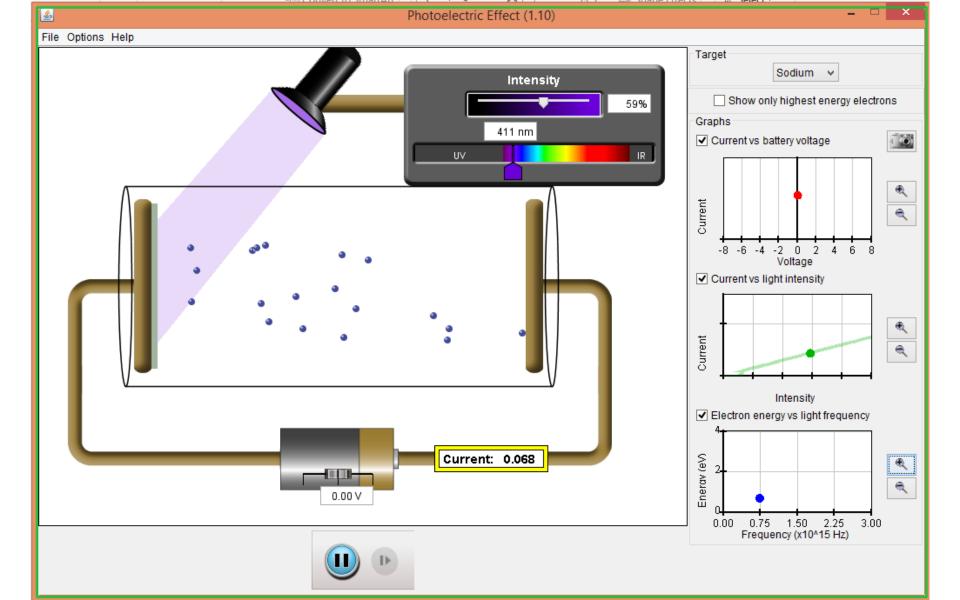
Mini-Lab: Photoelectric Effect

1. Use the PhET program Photoelectric Effect (Java).

- Experiment with the controls and the three graphs. Note that you may need to use the +/- buttons to zoom in or out to see meaningful data the graphs.
 Use sodium. Set voltage to zero. Use a purple color of light. Adjust the intensity of the light what is the relation
- shown in the current vs. intensity graph?

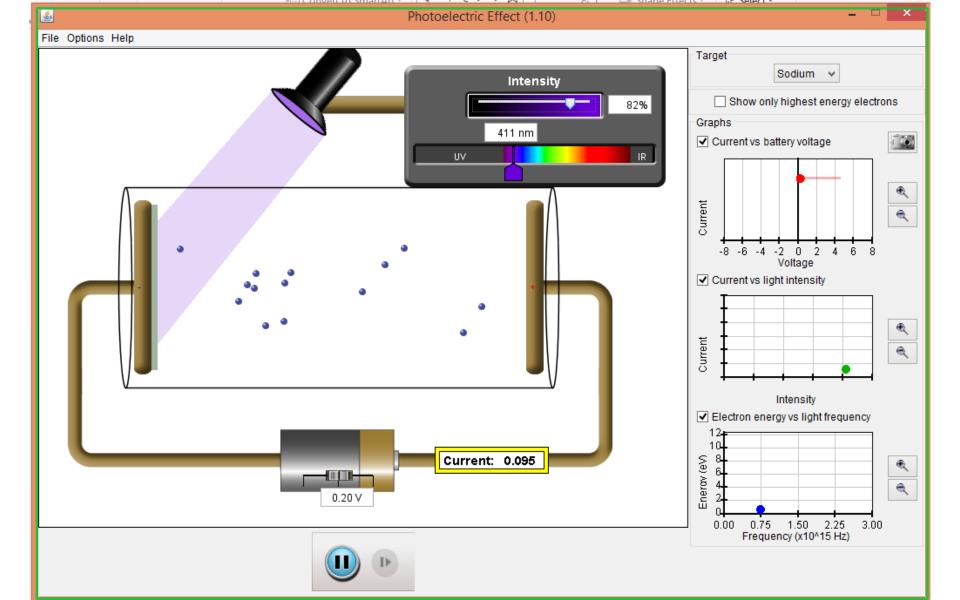
 4. Leave voltage at zero. Set intensity to 100%. Adjust the color of light and find the greatest wavelength that will cause *any* electrons to be emitted. Record this wavelength and determine the frequency of the light.



6. Try adjusting the voltage of the battery and observe the graphs and the effect on the animated electrons. In what direction does the electric field point in order to slow down the electrons?
7. For a given wavelength of light adjust the voltage of the battery until it is just enough to cause the current to be zero

5. Now set the color to purple and the intensity to 100%.

as shown in the rectangle by the battery. Electrons should be emitted but reverse direction just before reaching the other electrode. How can the kinetic energy of the electrons be determined based on the voltage? Experiment with this set of ideas and observe the energy vs. frequency graph – you may need to zoom in or out.



control for V_s by typing a value into the box instead of using the slider. Look carefully at the current vs. battery voltage graph and zoom in on it until you can inspect the resulting "step-like function" created by applying a full range of positive and negative voltages. Can you explain this graph? Why the plateaus? What explains the slope of the transition?

10. Try a different metal than sodium – how does this change

things? What stays the same?

Finally, use the controls and collect several data points for

stopping potential and wavelength and show the veracity

of $eV_s = hf - \phi$. Also solve for h, Planck's constant based

on your virtual measurements. Note: you get more precise