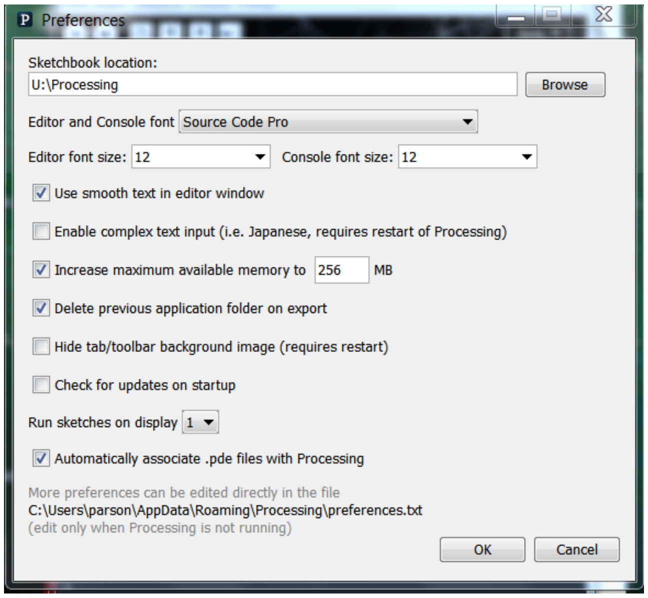
CSC 120 Introduction to Creative Graphical Coding, Fall 2017 (Handout for fall2020 CSC 220)

Dr. Dale E. Parson, Assignment 3, Implementing and testing a class that models your avatar.
This assignment is due via **D2L Dropbox** **<u>Assignment 3 avatarClass</u>** by **11:59 PM on Friday April 6**. When using Processing on the Kutztown campus Windows computers, make sure to start out **<u>every time</u>** by setting your Processing Preferences -> Sketchbook Location to U:\Processing. The U:\ drive is a networked drive that will save your work and make it accessible across campus. If you save it to your desktop or the lab PC you are using, you will lose your work when you log out. You must save it to the U:\ drive. If you do not have a folder called Processing under U:\, you must create one using the Windows Explorer. Processing Preferences is under the File menu on Windows.


Preferences

Sketchbook location: U:\Processing [Browse]

Editor and Console font: Source Code Pro

Editor font size: 12 Console font size: 12

[✓] Use smooth text in editor window

[ ] Enable complex text input (i.e. Japanese, requires restart of Processing)

[✓] Increase maximum available memory to 256 MB

[✓] Delete previous application folder on export

[ ] Hide tab/toolbar background image (requires restart)

[ ] Check for updates on startup

Run sketches on display 1

[✓] Automatically associate .pde files with Processing

More preferences can be edited directly in the file
C:\Users\parson\AppData\Roaming\Processing\preferences.txt
(edit only when Processing is not running)

[OK] [Cancel]

Use your working solution to assignment 2 as a starting point. If it has deficiencies, you must fix them before starting the following steps. We will have an in-class lab session. Come with your work-in-progress so you can ask questions and fix bugs. Don't wait until class to start.
I will post my solution to assignment 3 by end of March 24.

```
/* Purpose: Animate an avatar using Processing. Write a class & an array.
/* This is an example of my solution to assignment 3.
/* Steps in going from assignment 2 to assignment 3 example:
/*    1. Create "class Avatar" around definition of function avatar(...).
/*       Increase indent for the function. It is now inside the class.
/*       Change the function's name from "avatar" to "display".
/*    2. Remove function parameters avx, avy, avscale, and any other
/*       parameters to this function from its parameter list, and
/*       make them into data fields (variables) inside the class.
/*    3. Create another function "void move()" inside the Avatar
/*       class, and put all of the code to update the Avatar's location
/*       and other Avatar data fields into move(). Make sure to change
/*       the variable names in the move() code to match the Avatar data
/*       fields, e.g., avx, avy, etc. I needed to create two Avatar fields
/*       "avxspeed"  and "wiggleAmountspeed" to take the place of global
/*       variables avatarXspeed and legXspeed. Increase indent as needed.
/*    4. Create a constructor function Avatar(...) inside the class, with
/*       one parameter for every data field in the class. Name the parameters
/*      something slightly different from the fields. Use assignment statements
/*      to assign the values from the parameters into the fields.
/*      The class definition is now complete. Next, we must use it.
/*    5. Remove all *global* variables associated with Avatar state.
/*      Mine are avatarX, avatarXspeed, legX, & legXspeed.
/*      I will deduct points for use of any global variables with your Avatar functions.
/*      A global variable is a variable declared outside of the class and its functions.
/*    6. Create two new global variables "avatar1" and "avatar2" of class Avatar.
/*    7. In the setup() function, call the constructor for Avatar twice, once to
/*      assign an object reference into avatar1, and the second into avatar2.
/*      Use constructor arguments to give the starting location, speed, and other
/*      Avatar state for these objects. The constructor parameters show what arguments
/*      are needed for the constructor. I pulled some of my values from this code
/*      in assignment 2:
/*        avatar(avatarX, height/2, 1.0, legX); // initial value of avatarX and legX were 0.
/*        avatar(2 *width/3, height/5, 0.5, 0); // second call, without movement
/*    8. Inside the draw() function, replace calls to the avatar() function of
/*      assignment 2 (which drew the avatar) with calls to the display() and move()
/*      functions of objects avatar1 and avatar2. I replaced this code:
/*        avatar(avatarX, height/2, 1.0, legX); // initial value of avatarX and legX were 0.
/*        avatar(2 *width/3, height/5, 0.5, 0); // second call, without movement
/*      with this code:
/*        // Display, then move the avatars:
/*        avatar1.display();
/*        avatar2.display();
/*        avatar1.move();
/*        avatar2.move();
/*     9. Create a global array variable "crowd" that holds 2 or more Avatar objects.
/*       Construct two Avatar objects and place them into this array in setup().
/*       You can use the same parameters or different parameters as used for avatar1 and avatar2.
/*       We will cover arrays the week of October 16.
/*     10. Comment out the avatar1 & avatar2 display() and move() code of step 8
/*        (do not delete it), and add a loop (for or while loop) to display()
/*        and move() all of the Avatar objects in array crowd in its place.
/*     11. Update documentation comments at the top of the sketch, also for
/*        class Avatar and each of its fields and functions, and for any other code
/*        that has changed in this assignment 3 code.
/*      EACH OF THE ABOVE IS WORTH 9 POINTS IN ASSIGNMENT 3. YOU GET 1 POINT FOR
TURNING IT IN.
/*     LATE PENALTY IS 10% PER DAY AS USUAL. TURN IT IN VIA D2L ASSIGNMENT 3.
/* See http://faculty.kutztown.edu/parson/fall2017/avatarClass2017.txt
/* For assn2 see http://faculty.kutztown.edu/parson/fall2017/avatarFunction2017b.txt
 ********************************************************/
```

Drop avatarClass.pde into D2L Assignment Dropbox Assignment 3 avatarClass by 11:59 PM on Friday April 6.