*Your submission for this assignment **must include your full name** and your nine-digit **student number as a comment** at the top of the **source file you submit**. All source code files must be written using the **Python 3 programming language** and must run on the course's **official virtual machine**.*

***Submissions that crash** (i.e., terminate with an error) on execution will **receive a mark of 0**.*

Officially, the Due Date for this Assignment is:

***Friday, November 12th, 2021**, at **11:59pm EST**.*

Late Submissions are Accepted Without Penalty Until Sunday, November 14th, by 11:59pm EST.
Submissions received after that will not be accepted and will receive a mark of 0.

One of the motivations behind efforts made to divide complex programs into separate functions is modularity. This organization of functionality results in code that is easier to understand and reuse. For this assignment, you will be designing and implementing functions to draw the features for a cartoon face, and the instructor will then make calls to randomly selected submissions to see what faces are produced…

In order to complete this task, you will need to:

- assume that you are drawing your cartoon features onto a circle of radius 100
- design a set of cartoon eyes, a cartoon mouth, and a cartoon hat[1]
- separately test each function using code you write but do not include with your submission

Your submission for this assignment:

- must be a source code file with filename 'comp1405_f21_#########_assignment_08.py'
- must use the pygame.draw functions (and not, for instance, the Surface "blit" method)
- must implement the functions "draw_eyes", "draw_mouth", and "draw_hat" specified below

The function **"draw_eyes"** takes **no arguments** and has **no returns**. This function draws cartoon eyes on the surface named "win_sfc" and as though they were being drawn on a circle centered at position (150, 250). The eyes must be drawn using at least the colours black and white, and the outline should be black.
The function **"draw_hat"** takes **no arguments** and **returns exactly one string**. This function draws a cartoon hat (your choice of style) on the surface named "win_sfc" and as though it was being drawn on a circle centered at (250, 350). The function must use at least three colours, the outline should be black, and the string returned should desribe the "type" of hat in lowercase letters (e.g., "top hat", "baseball cap", etc.)
The function **"draw_mouth"** takes **two integer arguments** - x and y - and has **no returns**. This function draws a cartoon mouth on the surface named "win_sfc" and as though it was being drawn on a circle that was centered at the position (x, y). The mouth must be drawn using at least the colours black, white, and red, and the outline should be black.

[1] n.b., overly simplistic designs (e.g., two circles for eyes, a single arc for a mouth, etc.) will incur significant penalties