Lab #29

Programming and Algorithms

1. In the lecture we saw the COLOUR class:

|  |
| --- |
| class Colour:  def \_ \_init\_ \_(self, rgb\_value, name):  self.\_rgb\_value = rgb\_value  self.\_name = name  # END \_ \_init\_ \_  def set\_name(self, name):  self.\_name = name  # END set\_name  def get\_name(self):  return self.\_name  # END get\_name  # END class. |

We extended the GETTER class as follows:

|  |
| --- |
| def get\_name(self):  if self.\_name == "":  return "There is a blank value"  else:  return self.\_name  # ENDIF;  # END get\_name |

Extend the SETTER class in a similar way:

* Check if the name passed in is blank, if so raise an exception, otherwise set the name to self.name.

1. Show the outcomes of the following operations without using the PROPERTY function:

* redcolour = Colour("#FF0000", "Red")
* redcolour.name = “Red”
* redcolour.name .set\_name(“Red”)
* redcolour.name = “”
* redcolour.name .set\_name(“”)

1. Show the outcomes of the following operations using the PROPERTY function:

* redcolour = Colour("#FF0000", "Red")
* redcolour.name = “Red”
* redcolour.name .set\_name(“Red”)
* redcolour.name = “”
* redcolour.name .set\_name(“”)

|  |
| --- |
| e-mail me a completed solution to each of the above programs in a Word document, and include Labs #25-28 in this document also.  e-mail to [Damian.Gordon@dit.ie](mailto:Damian.Gordon@dit.ie) with subject heading “DT255 PaA Lab #25-29” |