

EXPLORER

OPEN EDITORS

WELCOME

percipio33_properties.py Perc...

percipio29_class_initialization....

PYTHON

Automate-Boring-Stuff

my_code

Percipio_Python3-Course

01_Start

02_Data-Sequence Types

03_Collections-Mapping-Loopi...

04_Modules-Functions

05_Classes

percipio27_classes_and_types...

percipio28_class_defination.py

percipio29_class_initialization...

percipio30_class_instance_me..

percipio31_static_methods.py

percipio32_inheritance.py

percipio33_properties.py

percipio34_properties_with_in..

percipio35_operator_overloa...

06_Working-with-Files

07_Comprehensions

08_Iterables-and-Generators

09_Exceptions

Python Projects_2014

CMD_Python_Set-Path.txt

Python_Basics.txt

Python_Clear-Window-Command...

python_exercises_00.py

python_exercises_01.py

Python_Tutorial_Running-Scripts...

Python_Tutorials.md

Welcome

percipio33_properties.py

percipio29_class_initialization.py x

```
1 # percipio29_class_initialization.py
2 # Percipio video: Classes; Class Initialization
3 # Demonstrate how Class Initialization happens
4 # This shows how the '__init__' method works
5 # The special 'init' method allows initialization of different instances of a class with different kinds
  of instance attribute values
6 #
7 nl = '\n'
8 import locale # module used so the proper format will be applied to the currency
9 import sys # module to indicate which platform this script is executing on
10
11 class Base_Model(): # class defination named Base_Model with colon to delimit code block belonging to this
  class
12     # Represent base model of a car
13     trim = 'normal' # class attribute as a string
14     engine_liters = 1.5 # class attribute as a floating point value
15     # These 2 attributes will belong to each instance of the class
16 #
17     def __init__(self, price, transmission='automatic', color='white'): # with the special init method,
  when a new instance is created. it refers to itself by the first parameter, will require a price
  parameter, & will use default parameters for Transmission and color if none is specified
18         self.price = price # required parameter
19         self.transmission = transmission # default parameter which can be overridden
20         self.color = color # default parameter which can be overridden
21         ''' As the new instance is created below, this special init method is automatically executed,
22         with the passed price value gets assigned to the instance 'self.price' attribute,
23         with the passed transmission value gets assigned to the instance 'self.transmission' attribute;
24         with none passed, default value is assigned,
25         with the passed color value gets assigned to the instance 'self.color' attribute; with none passed,
26         default value is assigned, '''
27 #
28     def info(self): # coop.info method
29         if sys.platform.startswith('win'): # checks if sys.platform starts with the string 'win' (win =
30         32bit or 64bit Windows machine)
31             locale.setlocale(locale.LC_ALL, 'us') # windows locale setting is the value to set all the
32             locale settings is the string 'us'
33         else: #
34             locale.setlocale(locale.LC_ALL, 'en_US.utf8') # locale settings on non-Windows platforms, the
35             value to set all the locale settings is the string 'en_US.utf8'
36         print('The price of %s was %s.' % (self, locale.currency(self.price))) #
37         (self, locale.currency(self.price))) #
38 #
39     def __str__(self): # The string (__str__) that is substituted for 'self' will be the 'str' method.
40         return 'a %s base model with %s transmission' % (self.color, self.transmission) #
41 #
42 # 1st created instance with 1 required parameter, price
43 coop = Base_Model(25000) # a new instance created based on the class: price=25000
```

Ln 24, Col 37 Spaces: 4 UTF-8 CRLF Python



EXPLORER

OPEN EDITORS

- Welcome
- percipio33_properties.py Perc...
- percipio29_class_initialization...

PYTHON

- Automate-Boring-Stuff
- my_code
- Percipio_Python3-Course
 - 01_Start
 - 02_Data-Sequence Types
 - 03_Collections-Mapping-Loopi...
 - 04_Modules-Functions
 - 05_Classes
 - percipio27_classes_and_types...
 - percipio28_class_defination.py
 - percipio29_class_initialization...
 - percipio30_class_instance_me...
 - percipio31_static_methods.py
 - percipio32_inheritance.py
 - percipio33_properties.py
 - percipio34_properties_with_in...
 - percipio35_operator_overloa...
 - 06_Working-with-Files
 - 07_Comprehensions
 - 08_Iterables-and-Generators
 - 09_Exceptions
- Python Projects_2014
- CMD_Python_Set-Path.txt
- Python_Basics.txt
- Python_Clear-Window-Command...
- python_exercises_00.py
- python_exercises_01.py
- Python_Tutorial_Running-Scripts...
- Python_Tutorials.md

Welcome percipio33_properties.py percipio29_class_initialization.py x

```
36 #
37 # 1st created instance with 1 required parameter, price
38 coop = Base_Model(25000) # a new instance created based on the class; price=25000,
    transmission=default=auto, color=default=white
39 coop.info() # info method of the created instance is called which will display the above attributes
40 #
41 # 2nd created instance with 1 required parameter, price, and color parameter change
42 coop_blue = Base_Model(price=25500, color='blue') # parameters are passed by keyword for this instance
    creation
43 coop_blue.info() #
44 #
45 # 3rd created instance with 1 required parameter, price, transmission parameter change, and color
    parameter change
46 coop_manual_red = Base_Model(26000, 'manual', 'red') # arguements are passed by postion (caution for
    accuracy)
47 coop_manual_red.info() #
48 #
49 # 4th created instance with 1 required parameter, price, transmission parameter change, and color
    parameter change
50 coop_manual_green = Base_Model(color='green', transmission='manual', price=26300) # best & most readable
    way of calling the Base_Model class constructors is use keyword arguments for everything, where order no
    longer matters.
51 coop_manual_green.info() #
52 #
53 # class attributes which are shaerd with all instances, not belonging to 'self'
54 print('coop.trim = ', coop.trim)
55 print('coop.engine_liters = ', coop.engine_liters)
56 # class instance attributes which are instances belonging to 'self'
57 print('coop.price = ', coop.price)
58 print('coop.transmission = ', coop.transmission)
59 print('coop.color = ', coop.color)
60 # RESULTS:
61 '''
62 The price of a white base model with automatic transmission was $25000.00.
63 The price of a blue base model with automatic transmission was $25500.00.
64 The price of a red base model with manual transmission was $26000.00.
65 The price of a green base model with manual transmission was $26300.00.
66 coop.trim = normal
67 coop.engine_liters = 1.5
68 coop.price = 25000
69 coop.transmission = automatic
70 coop.color = white
71 '''
72
```

