Class. e.g: class test():

Function. e.g: def show\_question(abc):

An example of class:

class AnonymousSurvey():

       """Collect anonymous answers to a survey question."""

➊     def \_\_init\_\_(self, question):

           """Store a question, and prepare to store responses."""

           self.question = question

           self.responses = []

➋     def show\_question(self):

           """Show the survey question."""

           print(question)

➌     def store\_response(self, new\_response):

           """Store a single response to the survey."""

           self.responses.append(new\_response)

➍     def show\_results(self):

"""Show all the responses that have been given."""

           print("Survey results:")

           for response in responses:

               print('- ' + response)

This class starts with a survey question that you provide ➊ and includes an empty list to store responses. The class has methods to print the survey question➋, add a new response to the response list ➌, and print all the responses stored in the list ➍. To create an instance from this class, all you have to provide is a question. Once you have an instance representing a particular survey, you display the survey question with show\_question(), store a response usingstore\_response(), and show results with show\_results().

To show that the AnonymousSurvey class works, let’s write a program that uses the class:

language\_survey.py

from survey import AnonymousSurvey  
  
# Define a question, and make a survey.  
question = "What language did you first learn to speak?"  
my\_survey = AnonymousSurvey(question)  
  
# Show the question, and store responses to the question.  
my\_survey.show\_question()  
print("Enter 'q' at any time to quit.\n")  
while True:  
    response = input("Language: ")  
    if response == 'q':  
        break  
    my\_survey.store\_response(response)  
  
# Show the survey results.  
print("\nThank you to everyone who participated in the survey!")  
my\_survey.show\_results()

As you use class like above, it’s more convenient than using separate functions. In case, you use separate functions, you have to pass an argument every time you use the function. e.g: need to pass question argument to show\_question, store\_response….

You should use class when there are many functions which are related to each other. Also by using class, you can use its OOP characteristics.