MyRepos\KI-Kurs-Mystro\Exercise_28062024.py

```
# Simple Dice Game

"""
01 --> Create a Base Class Player:
  ○ Attributes:
    class  --> ■ name: Name of the player (string)
               ■ score: Player's score (integer, default is 0)
  ○ Methods:
    def  ■ __init__(self, name): Constructor to initialize the name and score.
    def  ■ roll_dice(self): Method to simulate rolling a dice (random number between 1 and 6) and update the score.
    def  ■ get_score(self): Method to return the player's current score.

"""

import random

class Player:
    def __init__(self, name) -> None:
        self.name = name
        self.score = 0

    def roll_dice(self):
        roll = random.randint(1,6)
        self.score += roll
        return roll

    def get_score(self):
        #print(f"My Score : {self.score}")
        return self.score


"""
02 class  --> Create a Subclass ComputerPlayer that Inherits from Player:
    def __init__()  and  super().
    ○ No additional attributes or methods needed for simplicity.
"""
class ComputerPlayer(Player):  # subclass due to Player Inherits
    def __init__(self, name = "computer" ):  #  hat kein Einfluss, muss man den Namen beim Spielen geben
        super().__init__(name) # was macht genau dieses Satz


"""
03 Create a Game Class:
    class with  ○ Attributes:
          ■ player: An instance of the Player class.
          ■ computer: An instance of the ComputerPlayer class.
    ○ Methods:
        def --> ■  __init__(self, player_name): Constructor to initialize the player nd computer player.
        def --> ■ play_round(self): Method to play a round where both the player and computer roll the dice.
        def --> ■ display_scores(self): Method to display the current scores of both players.
        def --> ■ determine_winner(self): Method to determine the winner based on the scores
"""
class Game:
    def __init__(self, player_name) -> None:
```

```
55          self.player = Player(player_name)
56          #self.plyer2 = Player(player_name)  # muss ich medinsten 2 Players definieren
57          self.computer = ComputerPlayer()   # cann ich auch der Computer anderes definieren
58
59
60      def play_round(self):
61          player_roll = self.player.roll_dice()
62          computer_roll = self.computer.roll_dice()
63          #print(f"{self.player.name} :  {player_roll}")
64          #print(f"{self.computer.name} :  {computer_roll}")
65
66
67      def display_score(self):
68          print(f"{self.player.name}  score  : {self.player.get_score()}")
69          print(f"{self.computer.name}  score  : {self.computer.get_score()}")
70          return
71
72      def determine_winner(self):
73          if self.player.get_score() == self.computer.get_score(): return print(f"No winner")
74          if self.player.get_score() > self.computer.get_score():
75              #winner = self.player.get_score()
76              return (f"winner is {self.player.name}")
77          return (f"winner is {self.computer.name}")
78
79  def main():
80      #  play the Game:
81      p1=Game("abdou")
82      for i in range(15):
83          p1.play_round()
84      print(f"we are schon played  =  {i}  round" )
85      print(f"{p1.display_score()}")
86      print(f"{p1.determine_winner()}")
87
88  if __name__ == "__main__":
89      main()
90
91
92
93
```