adhish-203-lab3

August 12, 2023

Adhish Bahl 2347203 1MCA B Python Lab 3

0.1 Lab 3

Question: Write a function in Python with a string such that it accepts a parameter- “stringsplit”. This encoded string will contain your name, domain name and register number. You can separate the values in the string by any number of underscores. [The string should not contain any other underscore symbols in your name, domain name and register number]. The function should return a Python dictionary with your name, domain name and register number._ For example, if the input would be ” Aaron_**Googleplaystore**___2347201”. Then the function should return the output as follows: { “name”: ” Aaron “,”Domain_name“:” Googleplaystore “,”Regno“:”2347201” }

[2]:
```
import re
def takeInput():
    confirm = 0
    while confirm != 1:
        stringMain = input("\n\nEnter the string with your name, domain and reg␣
 ↪no seperated with '_': ")
        confirm = int(input(f"\n{stringMain} is the string you want to enter?
 ↪\n1. Confirm\n2. Enter String Again\n----------"))
    splitAndDictionary(stringMain)

def splitAndDictionary(mainString):
    tempList = re.split("_+", mainString)

    # tempList2 =  ["", "", ""]
    # index = 0
    # for i in range(len(mainString)):
    #     if mainString[i] != "_":
    #         tempList2[index] = str(tempList2[index]) + str(mainString[i])
    #     elif mainString[i] == "_":
    #         index = index + 1
    # print(tempList2)

    finalDict = {"Name": tempList[0], "Domain": tempList[1], "Registration No.":
 ↪ tempList[2]}
    print("\nInitial String: ", mainString)
    print(f"\n\n{finalDict}")
```

```
takeInput()
```

```
Initial String:  Adhish__Sneakers____2347203


{'Name': 'Adhish', 'Domain': 'Sneakers', 'Registration No.': '2347203'}
```

0.2 Lab 4

Question: Write a Python program to implement the object-oriented concepts of multiple, Multilevel and Hierarchical Inheritances using your domain applications.

[3]:

```
class Sneaker:
  def __init__(self, brand, model, color, size, price):
    self.brand = brand
    self.model = model
    self.color = color
    self.size = size
    self.price = price

  def __str__(self):
    return (f"\nBrand: {self.brand} \nModel: {self.model} \nColor: {self.color}␣
 ↪\nSize: {self.size} \nPrice:  {self.price}")


class LimitedEdition(Sneaker):
  def __init__(self, brand, model, color, size, price, release_date, quantity):
    super().__init__(brand, model, color, size, price)
    self.release_date = release_date
    self.quantity = quantity

  def __str__(self):
    return super().__str__() + f"\nReleased Date: {self.release_date}␣
 ↪\nQuantity: {self.quantity} pairs available"


class PreOwned(Sneaker):
  def __init__(self, brand, model, color, size, price, condition, seller):
    super().__init__(brand, model, color, size, price)
    self.condition = condition
    self.seller = seller

  def __str__(self):
    return super().__str__() + f"\nCondition: {self.condition} \nSold by: {self.
 ↪seller}"
```

```
class Customiser:
  def __init__(self, name, company, rating):
    self.name = name
    self.company = company
    self.rating = rating

  def __str__(self):
    return (f"\nCustomised By: {self.name} \nComapny: {self.company} \nRating:␣
  ↪{self.rating}")



class Customized(PreOwned, Customiser):
  def __init__(self, brand, model, color, size, price, condition, seller, name,␣
  ↪company, rating, design):
    Sneaker.__init__(self, brand, model, color, size, price)
    PreOwned.__init__(self, brand, model, color, size, price, condition, seller)
    Customiser.__init__(self, name, company, rating)
    self.design = design

  def __str__(self):
    return super().__str__() + f"\nCustomised By: {self.name}\nCompanyName:␣
  ↪{self.company}\nCompany Rating: {self.rating}\nCustom design: {self.design}"

s1 = Sneaker("Nike", "Air Force 1", "White", 10.5, 9000)
s2 = LimitedEdition("Adidas", "Yeezy Boost 350 V2", "Black/Red", 9.5, 220,␣
  ↪"2023-02-11", 500000)
s3 = PreOwned("Converse", "Chuck Taylor All Star", "Navy", 8.5, 15000, "Good",␣
  ↪"Alice")
s4 = Customized("Vans", "Old Skool", "Black/White", 11.5 ,80000 , "New", "Bob",␣
  ↪"Alice", "AliceCustoms", 4.5, "Flames")

print(s1)
print(s2)
print(s3)
print(s4)
```

```
Brand: Nike
Model: Air Force 1
Color: White
Size: 10.5
Price:  9000

Brand: Adidas
```

```
Model: Yeezy Boost 350 V2
Color: Black/Red
Size: 9.5
Price:  220
Released Date: 2023-02-11
Quantity: 500000 pairs available

Brand: Converse
Model: Chuck Taylor All Star
Color: Navy
Size: 8.5
Price:  15000
Condition: Good
Sold by: Alice

Brand: Vans
Model: Old Skool
Color: Black/White
Size: 11.5
Price:  80000
Condition: New
Sold by: Bob
Customised By: Alice
CompanyName: AliceCustoms
Company Rating: 4.5
Custom design: Flames
```