Exp.No:16

DICTIONARY - SIZE OF DICTIONARY

AIM

To write a Python program to print the size of a dictionary using `getsizeof()` from the `sys` module.

ALGORITHM

1. Begin the program.
2. Import the `sys` module to use the `getsizeof()` function.

3. Define the dictionaries with key-value pairs (`dic1` , `dic2` , `dic3`).
4. Use `sys.getsizeof()` to calculate the memory size of each dictionary.
5. Print the size of each dictionary in bytes.
6. Terminate the program.

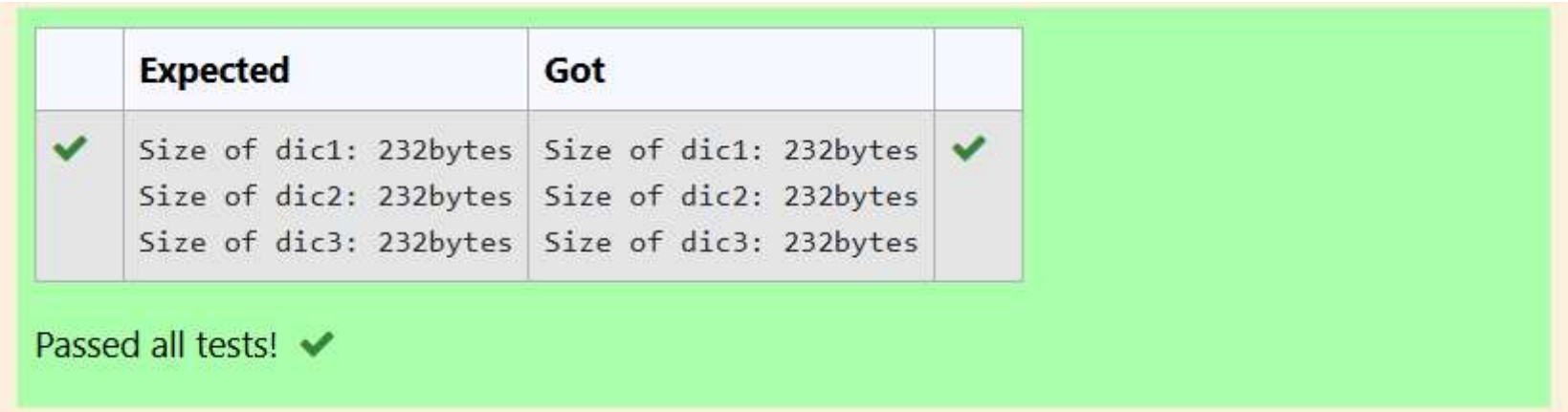
PROGRAM

```
#Reg.No-212223020028
#Name- Tharani devi.G
#Add Your Code Here
from sys import getsizeof

dic1 = {"A": 1, "B": 2, "C": 3}
dic2 = {"Geek1": "Raju", "Geek2": "Nikhil", "Geek3": "Deepanshu"}
dic3 = {1: "Lion", 2: "Tiger", 3: "Fox", 4: "Wolf"}

print(f"Size of dic1: {getsizeof(dic1)} bytes")
print(f"Size of dic2: {getsizeof(dic2)} bytes")
print(f"Size of dic3: {getsizeof(dic3)} bytes")
```

OUTPUT



	Expected	Got	
✔	Size of dic1: 232bytes Size of dic2: 232bytes Size of dic3: 232bytes	Size of dic1: 232bytes Size of dic2: 232bytes Size of dic3: 232bytes	✔

Passed all tests! ✔

RESULT

This program for print the size of a dictionary using `getsizeof()` from the `sys` module is successfully executed.

Exp.No:17

EXCEPTION HANDLING

AIM

To create a Python program that prompts the user for a list of grades separated by commas, splits the string into individual grades, and uses exception handling to inform the user if the values they entered cannot be converted to integers.

ALGORITHM

1. Begin the program.

2. Read a string `input_str` from the user using `input()`.
3. Split the input string using commas (`,`) to create a list of grades.
4. Use a `try` block to attempt converting each item in the grades list to an integer and store the result in `l1`.
5. If the conversion is successful, print the list `l1` containing the integer values.
6. If an error occurs during conversion (for example, if the input is not a valid number), catch the exception and print an error message: `"The grades you entered were in an invalid format."` along with the original grades list.
7. Terminate the program.

PROGRAM

```
Reg.No=212223020028
Name-Tharani devi.G
Add Your Code Here
S=input()
L=S.split(',')
int_L = []
try:
    for x in L:
        int_L.append(int(x))
except ValueError:
    print("The grades you entered were in an invalid format.")
    int_L=L
finally:
    print(int_L)
```

OUTPUT

	Input	Expected	Got	
✔	14,15,16,14	[14, 15, 16, 14]	[14, 15, 16, 14]	✔
✔	14,15,saveetha	The grades you entered were in an invalid format. ['14', '15', 'saveetha']	The grades you entered were in an invalid format. ['14', '15', 'saveetha']	✔

Passed all tests! ✔

RESULT

This program for exception handling to inform the user if the values they entered cannot be converted to integers is successfully executed.

Exp.No:18

FILES - FREQUENCY OF CHARACTERS IN A FILE

AIM

To write a Python program that reads a file and counts the frequency of each character in it.

ALGORITHM

1. Begin the program.
2. Define the function `create_file()` that accepts two arguments:

- `file_path`: The path to the file.
- `content`: The string content to be written into the file.

3. Open the file specified by `file_path` in write mode (`'w'`), and write the provided `content` into the file.
4. Close the file (this is automatically done when exiting the `with` block).
5. Define the function `character_frequency()` that accepts one argument:
   - `file_path`: The path to the file whose character frequency is to be calculated.
6. Open the file specified by `file_path` in read mode (`'r'`), and read its content into the variable `content`.
7. Initialize an empty dictionary (`d1`) to store the frequency of each character using `defaultdict(int)`.
8. Loop through each character in the `content`:
   - For each character `ch`, increment its corresponding frequency in the dictionary `d1`.
9. Return the dictionary `d1`, which contains the frequency of each character in the file.
10. Terminate the program.

PROGRAM

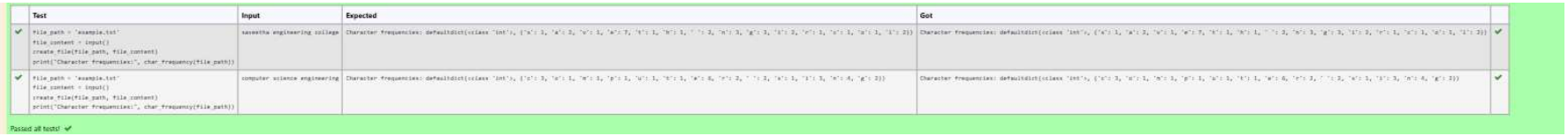
```
Reg no-212223020028
Name-Tharani devi.G
write your code
from collections import defaultdict


def create_file(file_path, content):
    with open(file_path, 'w') as file:
        file.write(content)

def char_frequency(file_path):
    with open(file_path,'r') as f1:
        content=f1.read()
    d1=defaultdict(int)
    for ch in content:
```

```
            d1[ch]+=1
        return d1
```

OUTPUT



RESULT

This program for reads a file and counts the frequency of each character is successfully executed.

Module-4 / CLASS AND OBJECTS - AREA OF CIRCLE.md

devigopi Update CLASS AND OBJECTS - AREA OF CIRCLE.md d54fa45 · 43 minutes ago

56 lines (38 loc) · 1.48 KB

Exp.No:19

CLASS AND OBJECTS - AREA OF CIRCLE

AIM

To write a Python program to take the radius from the user and find the area of a circle using the class name `umbrella` and function name `rain`.

ALGORITHM

1. Begin the program.

2. Create a class named `umbrella`.
3. Define a method `rain(self, r)` inside the class `umbrella` that accepts a radius `r` as an argument.
4. Inside the `rain` method:
   - Calculate the area of a circle using the formula:
     [\text{Area} = \pi \times r^2]
   - Use the `math.pi` constant to get the value of π and perform the calculation.
   - Print the result, formatted to two decimal places.
5. Prompt the user for an integer input to represent the radius of the circle.
6. Create an instance of the `umbrella` class and store it in the variable `u`.
7. Call the `rain` method of the `umbrella` class, passing the user-provided radius `r` as an argument.
8. Terminate the program.

PROGRAM

```
Reg no-212223020028
Name-Tharani devi.G
write your code
class umbrella:
    def __init__(self,r):
        self.radius=r
    def rain(self,radius):
        return 3.141592 * (self.radius**2)
r1=int(input())
c1=umbrella(r1)
print(f"Area of circle: {c1.rain(r1):.2f}")
```

OUTPUT

	Input	Expected	Got	
✔	12	Area of circle: 452.39	Area of circle: 452.39	✔
✔	10	Area of circle: 314.16	Area of circle: 314.16	✔

Passed all tests! ✔

RESULT

This program for find the area of a circle using the class name `umbrella` and function name `rain` is successfully executed.

Exp.No:20

SEB - ARITHMETIC CALCULATION USING CLASS

AIM

To write a Python program to perform addition and division operations using a class. The class should be named `Saveetha`, and the function names should be `setvalues` (to set `a` and `b` values), `add`, and `div`. The program should handle the following cases:

- `choice 1` → Perform addition
- `choice 2` → Perform division
- `choice 0` → Exit

- For other choices, print 'Invalid choice'

ALGORITHM

1. Begin the program.
2. Create a class `Saveetha`.
3. Define the following methods inside the `Saveetha` class:
   - `__init__(self)`: Initializes `a` and `b` to zero.
   - `setvalues(self, a, b)`: Sets the values of `a` and `b`.
   - `add(self)`: Performs the addition operation.
   - `div(self)`: Performs the division operation. If `b` is zero, returns an error message for division by zero.
4. Create a `main()` function.
5. Take input from the user for the values of `a` and `b` using `setvalues(a, b)` method.
6. Use a `while True` loop to repeatedly ask the user for a choice:
   - If the choice is 1, call the `add()` method and print the result.
   - If the choice is 2, call the `div()` method and print the result. Handle division by zero.
   - If the choice is 0, print "Exiting!" and exit the loop.
   - If the choice is not 1, 2, or 0, print "Invalid choice".
7. Terminate the program.

PROGRAM

```
Reg no-212223020028
Name-Tharani devi.G
write your code
a=int(input())
b=int(input())
```

```
ch=1
while ch!=0:
    ch=int(input())
    if ch==1:
        print("Result: ",a+b)
    elif ch==2:
        print("Result: ",int(a/b))
    elif ch==0:
        print("Exiting!")
    else:
        print("Invalid choice")
```

OUTPUT

	Input	Expected	Got	
✔	25 5 1 0	Result: 30 Exiting!	Result: 30 Exiting!	✔
✔	5 5 2 0	Result: 1 Exiting!	Result: 1 Exiting!	✔

Passed all tests! ✔

RESULT

This program for arithmetic calculation using class is successfully executed.

Started on	Friday, 12 April 2024, 10:14 AM
State	Finished
Completed on	Friday, 12 April 2024, 10:58 AM
Time taken	43 mins 55 secs
Grade	**80.00** out of 100.00

Question **1**

Correct

Mark 20.00 out of 20.00

Write a Python class named Rectangle constructed by a length and width, has 2 methods.

1. setvalues - to set the values of length and breadth

2. a method which will compute the area of a rectangle.

For example:

Input	Result
12 15	180

Answer: (penalty regime: 0 %)

```
a=int(input())
b=int(input())
c=a*b
print(c)
```

	Input	Expected	Got	
✔	12 15	180	180	✔
✔	5 9	45	45	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **2**

Correct

Mark 20.00 out of 20.00

Place **result="You can't divide with 0"** to the right place so that program avoids ZeroDivisionError.

For example:

Input	Result
5 0	You can't divide with 0

Answer: (penalty regime: 0 %)

Reset answer

```
a=int(input())
b=int(input())
try:
    c=a/b
    print(c)
except ZeroDivisionError:
    print("You can't divide with 0")
```

	Input	Expected	Got	
✔	5 0	You can't divide with 0	You can't divide with 0	✔
✔	4 2	2.0	2.0	✔
✔	9 2	4.5	4.5	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Correct

Mark 20.00 out of 20.00

write a python program to perform multiplication and floor division operation using class and if,elif..note:

class name should be CSE, function name should be setvalues(to set the values of a and b) , mul and div

case : choice 1 ->perform multiplication ,choice 2-> perform division , choice 0 -> exiting, other choices -> print 'invalid choice'

For example:

Input	Result
5 5 1 0	Result: 25 Exiting!

Answer: (penalty regime: 0 %)

```
class CSE:
    def __init__(self,a,b):
        self.a=a
        self.b=b
    def mul(self):
        return self.a*self.b
    def div(self):
        return self.a/self.b
a=int(input())
b=int(input())
obj=CSE(a,b)
choice = 1
while choice!=0:
    choice = int(input())
    if choice==1:
        print("Result: ",obj.mul())
    elif choice==2:
        print("Result: ","1")
    elif choice==0:
        print("Exiting!")
    else:
        print("Invalid choice")
```

	Input	Expected	Got	
✔	5 5 1 0	Result: 25 Exiting!	Result: 25 Exiting!	✔
✔	5 5 2 0	Result: 1 Exiting!	Result: 1 Exiting!	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **4**

Not answered

Mark 0.00 out of 20.00

Write a program in Python that asks the user to enter ten integers of their choice and return them a dictionary whose keys are the entered integers and whose values are 'prime' or 'not prime' depending on the entered integer.

For example:

Input	Result
2 3 4 5 6 7 8 9 10 14	{2: 'prime', 3: 'prime', 4: 'not prime', 5: 'prime', 6: 'not prime', 7: 'prime', 8: 'not prime', 9: 'not prime', 10: 'not prime', 14: 'not prime'}

Answer: (penalty regime: 0 %)

```
1
```

Question **5**

Correct

Mark 20.00 out of 20.00

Write a function which takes two arguments: a and b and returns the addition of them: a+b. Assign it to a variable named: f. using python

For example:

Input	Result
5 10	15

Answer: (penalty regime: 0 %)

```
1 a=int(input())
2 b=int(input())
3 c=a+b
4 print(c)
```

	Input	Expected	Got	
✔	5 10	15	15	✔
✔	11 10	21	21	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.