

```
1  ### 1. Read a file and store it in the variable file_object
2  file_object = open('data/pi_digits.txt')
3  content = file_object.read()
4  file_object.close()
5
6  ## print the content (String) of the file object
7  print(content)
8
9  ## Note: you can also use absolute paths
10 ## Note: if you want to use backslash, use '\\' in your path
11
12 ### 2. Read a file and store it in the variable file_object
13 file_object = open('data/pi_digits.txt')
14 for line in file_object:
15     print('X', line.rstrip())
16
17 file_object.close()
18
19
20 ### 3. Create a list with lines items
21 file_object = open('data/pi_digits.txt')
22 lines_list = file_object.readlines()
23
24 ## print the whole list
25 print(lines_list)
26
27 file_object.close()
28
29 ## 4. Working with the content
30 file_object = open('data/pi_million_digits.txt')
31 lines_list = file_object.readlines()
32
33 pi_string = ''
34 for item in lines_list:
35     pi_string += item.strip()
36
37
38 ### Change the file name to data/pi_million_digits.txt
39 ### Text if your birthday is contained in the first million digits of pi
40
41 birthday = '290978'
42
43 if birthday in pi_string:
44     print("My birthday is in the first million digits of pi!")
45 else:
46     print("My birthday is not in the first million digits!")
47
48 file_object.close()
```

```
1 # 1. Exceptions to handle errors
2
3 num_agents = 0
4 total_payoff = 22
5
6 avg_payoff = total_payoff/num_agents # (doesn't work)
7 print(avg_payoff) # (doesn't work)
8
9 try:
10     avg_payoff = total_payoff/num_agents
11 except ZeroDivisionError:
12     print("Sorry, there is no agents out there.")
13     avg_payoff = -1
14 else:
15     print(f"The average payoff is {avg_payoff}.")
16
17
18 # 2. Exceptions to handle FileNotFoundError
19
20 filename = 'I_do_not_exist.txt'
21
22 file_object = open('data/'+filename, 'r') # (doesn't work)
23
24 try:
25     file_object = open('data/'+filename, 'r')
26 except FileNotFoundError:
27     print(f"Sorry, the file {filename} does not exist.")
28
29
```

```
1  ### 1. Writing to a file
2  ### there is also read mode: r; append mode a
3  ### try the read mode, try the append mode. What happens?
4
5  filename = 'programming.txt'
6
7  file_object = open('data/'+filename, 'w')
8  file_object.write("I love programming. ")
9  file_object.write("I love creating databases.") # \n
10
11 file_object.close()
12
13 ### 2. Creating a user information file
14 import random
15
16 filename = 'user_info.txt'
17 file_object = open('data/'+filename, 'w')
18
19 names = ['maria', 'marco', 'alberto', 'heino', 'lena', 'carolina', 'leo', 'bi
20 hobbies = ['painting', 'swimming', 'reading', 'programming', 'running', 'bi
21
22 num_sentences = 200
23 index_sentences = 0
24
25 while index_sentences < num_sentences:
26
27     # pick a random name from names
28     name_index = random.randint(0, len(names)-1)
29     my_name = names[name_index]
30
31     # pick a random sub-list of hobbies
32     num_hobbies = random.randint(2,5)
33     my_hobby_list = random.sample(hobbies,num_hobbies)
34     my_hobby_string = ''
35     for entry in my_hobby_list:
36         my_hobby_string += (entry+", ")
37     my_hobby_string = my_hobby_string[:-2]
38
39     # pick a random age value
40     my_age = random.randint(10,100)
41     sentence = f"{my_name.title()} is {my_age} years old "
42     sentence += f"and has the hobbies {my_hobby_string}."
43
44     file_object.write(sentence+"\n")
45
46     index_sentences += 1
47
48 file_object.close()
```

```
1  ### 1. Access and analyze user info
2  file_object = open('data/user_info.txt')
3
4  # initialize some count variables
5  acc_age_prog = 0
6  num_prog = 0
7  acc_age_prog_fem = 0
8  num_prog_fem = 0
9
10 for line in file_object:
11
12     line_list = line.split()
13
14     #print(line_list)
15
16     for index in range(0, len(line_list)):
17         entry = line_list[index]
18         if entry[-1] == ',' or entry[-1] == '.':
19             line_list[index] = entry[:-1]
20
21     #print(line_list)
22
23     my_age = int(line_list[2])
24
25     if 'programming' in line_list:
26         acc_age_prog += my_age
27         num_prog += 1
28
29     if line_list[0][-1] == 'a':
30         my_gender = 'f'
31     else:
32         my_gender = 'm'
33
34     if 'programming' in line_list and my_gender == 'f':
35         acc_age_prog_fem += my_age
36         num_prog_fem += 1
37
38
39 print("Average programmer's age:", acc_age_prog*1.0/num_prog)
40 print("Average female programmer's age:", acc_age_prog_fem*1.0/num_prog_fem)
41
42
43 file_object.close()
44
```

```
1 # import matplotlib
2 import matplotlib.pyplot as plt
3
4 # my data
5 labels = ['Mercedes', 'Audi', 'Toyota', 'Subaru', 'Porsche', 'Ford' ]
6 frequencies = [20, 14, 12, 6, 4, 4]
7
8 # draw a pie chart
9 fig1, ax1 = plt.subplots()
10 ax1.pie(frequencies, labels=labels, autopct='%1.1f%%', startangle=90)
11 ax1.set_title("Relative car frequencies of my car stock")
12
13 plt.show()
14
15 # more data
16 frequencies_opponent = [17, 17, 18, 2, 1, 5]
17
18 # draw a bar diagram
19 fig2, ax2 = plt.subplots()
20
21 ax2.bar(labels, frequencies, width=-0.4, color='g', align='edge')
22 ax2.bar(labels, frequencies_opponent, width=0.4, color='y', align='edge')
23
24 ax2.set_xlabel('Car types')
25 ax2.set_ylabel('Absolute frequency')
26 ax2.set_title("Frequencies of car types of my stock (green) and opponent's (yellow)")
27
28 plt.show()
```