Zaid Shariff

2282437

Python Assignment 4

(1) Demonstrate the following Python Set operations with suitable examples:

(a) Creating a Set

```
fruits={"Apple","Mango","Banana","Apple","Papaya","Green Apple","Date"}
```

(b) Modifying a Set [add(), discard(), remove(), pop(), clear()]

```
fruits.add("Orange")
fruits
```

```
{'Apple', 'Banana', 'Date', 'Green Apple', 'Mango', 'Orange', 'Papaya'}
```

```
fruits.remove("Banana")
fruits
```

```
{'Apple', 'Date', 'Green Apple', 'Mango', 'Orange', 'Papaya'}
```

```
pop_frt=fruits.pop()
print(pop_frt+" was removed from the set.",fruits)
```

```
Papaya was removed from the set. {'Orange', 'Green Apple', 'Apple', 'Date'}
```

```
fruits.discard("Papaya")
fruits
```

```
{'Apple', 'Date', 'Green Apple', 'Orange'}
```

```
fruits.clear()
fruits
```

```
set()
```

(c) Set operations [union, intersection, difference, symmetric difference]

```
petrol_cars={"Maruti","BMW","Toyota","Hyundai","RollsRoyce","Renault"}
electric_cars={"Toyota","Hyundai","BMW","Tesla","Reva"}
```

```
all_cars=petrol_cars | electric_cars
all_cars
```

```
{'BMW',
 'Hyundai',
 'Maruti',
 'Renault',
 'Reva',
 'RollsRoyce',
 'Tesla',
 'Toyota'}
```

```
petrolANDelectric= petrol_cars & electric_cars
petrolANDelectric
```

```
{'BMW', 'Hyundai', 'Toyota'}
```

```
onlyPetrol=petrol_cars - electric_cars
onlyPetrol
```

```
{'Maruti', 'Renault', 'RollsRoyce'}
```

```
onlyelectric= electric_cars - petrol_cars
onlyelectric
```

```
{'Reva', 'Tesla'}
```

```
eitherPetORElec=petrol_cars ^ electric_cars
eitherPetORElec
```

```
{'Maruti', 'Renault', 'Reva', 'RollsRoyce', 'Tesla'}
```

(2) Demonstrate the implementation of exception handling mechanism in Python. (Use try, except, finally and raise statement)

```
try:
  x=5/0
  print(x)
except Exception as e:
  print("A "+str(e)+" error occured")
```

```
A division by zero error occured
```

```
animals=["cat","dog","monkey","cow"]
i = input("Enter a name: ")
i=i.lower()
if i in animals:
  raise Exception("Entered name is invalid. Please try again.")
```

```
Enter a name: DOG
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
<ipython-input-38-2fa57e111246> in <module>()
      3 i=i.lower()
      4 if i in animals:
----> 5   raise Exception("Entered name is invalid. Please try again.")

Exception: Entered name is invalid. Please try again.
```

```
l=input("Enter a list of number seperated by comma: ").split(",")
try:
  e=[]
  for i in l:
    if int(i)%2==0:
      e.append(i)
  if e==[]:
    raise Exception("None of the numbers entered were even.")
finally:
  print("The even numbers entered were ",e)
  l.clear()
  print(l)
```

```
Enter a list of number seperated by comma: 1,3,7,9,4
The even numbers entered were  ['4']
[]
```

(3) Demonstrate the implementation of the following object oriented concepts in Python using any scenario of your choice:

(a) Creating a class with member variables and member methods (c) Constructors

```
class person:
  def __init__(self,name,gender,age):
    self.name=name
    self.gender=gender
    self.age=age

  def sleeps(self):
    print(f"{self.name} sleeps")

  def eats(self):
      print(f"{self.name}eats")

  hands=2
  legs=2
  hair="black"
```

(b) Creating an object and accessing the class variables and methods

```
person_1=person("Raj","male",31)
person_1.sleeps()
print(person_1.age)
print(person_1.hair)
```

```
Raj sleeps
31
black
```

(d) Inheritance

```
class student(person):
  def __init__(self, name, gender, age, school):
        self.school = school
        person.__init__(self,name,gender,age)
  def studies(self):
    print(f"{self.name} studies.")

  major="Computer Science"
```

```
stud_1=student("Zuhair","male",23,"CU")
stud_1.studies()
print(f"{stud_1.name} is the name")
print(f"{stud_1.name}'s major is {stud_1.major}")
```

```
Zuhair studies.
Zuhair is the name
Zuhair's major is Computer Science
```