

```
In [1]: class Restaurant:
    name = "Parsakhani"
    def __init__(self, name, age):
        self.name = name
        self.age = age
    def sit(self):
        print(f"({self.name}) is now sitting.")
    def roll_overs(self):
        print(f"({self.name}) rolled over!")

my_dog = Dog('Willie', 6)

my_dog.sit()
my_dog.roll_overs()

print(f"my dog's name is {my_dog.name}.")
print(f"my dog is {my_dog.age} years old.")

In [1]: class User:
    def __init__(self, f_name, l_name, date_birth, qualification):

In [2]: class Car:
    def __init__(self, make, model, year, odometer):
        self.make = make
        self.model = model
        self.year = year
        self.odometer = odometer
        #setting a default value for an attribute.
        self.speed = 0
        self.spldlimt=0
        self.spldlimt1=0
        self.shlpsped=40
        self.incrctm=0

    def get_descriptive_name(self):
        long_name = f'({self.year}) {self.make} ({self.model})'
        return long_name.title()

    def odometer_reading(self):
        print(f"Odometer reading is {self.odometer}")

# modifying an attribute value directly
def __speed_limit(self):
    print(f"the speed limit at the school area {self.speed}")

def school_spldlimt(self):
    print(f"the school speed limit is {self.shlpsped}")

def user_input(self):
    spldlimt1 = int(input("Enter the school area speed limit here to know is it correctly define d or not?"))
    print(spldlimt1)
    self.spldlimt= spldlimt1
    print(f"Decleared speed limit is {self.shlpsped}")
    if spldlimt1<= self.shlpsped:
        print(f"the entered {spldlimt1}, and this speed limit is not exceeds the defined speed 1 limit ")
    else:
        print(f"You entered {spldlimt1}, and this speed limit exceeds the original limit ")

    def increment_spldlimt(self,incrctm):
        self.speed +=incrctm

    def capacity_fill_gas(self):
        print("The gas tank capacity is: 40 Gallons")

my_new_car = Car('audi', 'a4', 2019,20)

print(my_new_car.get_descriptive_name())
print(my_new_car.odometer_reading())

sp = my_new_car.speed_limit = 30
spl1 = my_new_car.school_spldlimt = 25
my_new_car.user_input()
print("Before increment {my_new_car.speed}")
increment = my_new_car.increment_spldlimt(10)
print(f"the speed limit {my_new_car.spldlimt}")
my_new_car.capacity_fill_gas()

2019 Audi A4
Odometer reading is 20
None
Enter the school area speed limit here to know is it correctly defined or not:89
Decleared speed limit is 40
You entered 89, and this speed limit exceeds the original limit
Before increment 0
After increment 10
The gas tank capacity is: 40 Gallons

In [6]: class Restant:
    def __init__(self,number_of_table,number_served):
        self.number_served = number_served
        self.number_of_table = number_of_table
        self.additionallard = 0

    def reset(self):
        print(f"the number of people served {self.number_served}")

    def number_of_table(self):
        print(f"the number of table ordered {self.number_of_table}")

    def add_numerserved(self,additionalard):
        self.number_served += additionalard
        print(f"after adding the number of people is {self.number_served}")

restntnr = Restant(303,23)
print(restntnr.number_served)
print(restntnr.number_of_table)
restntnr.numrcpl_served()
restntnr.number_of_table()

restntnr.add_numerserved(303)
print(f"after rollng the people is {restntnr.number_served}")

23
303
The number of people served 23
The number of table ordered 303
After adding the number of people is 526
After adding the people is 526

In [34]: class UserDet:
    def __init__(self, login_attempts):
        self.login_attempts = login_attempts
        incrctmpt=0
        decatempt=0

    def user_login(self):
        print(login_attempts)

    def increment_login_attempts(self,incrctmpt):
        self.login_attempts += incrctmpt
        print(f"after increment the login attempts are : {self.login_attempts}")

    def reset_login_attempts(self,decatempt):
        self.login_attempts = decatempt
        self.my_ticket = my_ticket
        print(f"after decrement the login attempt is : {self.login_attempts}")

log = UserDet(10)
log.increment_login_attempts(1)
log.reset_login_attempts(0)

log = UserDet(20)
log.increment_login_attempts(9)
log.reset_login_attempts(0)

After increment the login attempts are : 21
After decrement the login attempt is : 0
After increment the login attempts are : 29
After decrement the login attempt is : 0

In [13]: #USERINTERFACE
class ElectricCar(Car):
    def __init__(self, make, model, year, odometer):
        super().__init__(make, model, year, odometer)
    def get_range(self):
        ent_battery_size = int(input("Entered the battery size"))
        print(ent_battery_size)
        self.battery_size = ent_battery_size
        if ent_battery_size >=100:
            print(f"({ent_battery_size})The range is = 250")
        else:
            print(f"({ent_battery_size})The range is = 150")

Tesla = ElectricCar('Tesla','Hyprid',2019,23)
print(Tesla.get_descriptive_name())

Tesla.get_range()

2019 Tesla Hybrid
Entered the battery size:89
89
89so, the range is = 150

In [15]: class ElectricCar(Car):
    def __init__(self, make, model, year, odometer):
        super().__init__(make, model, year, odometer)
        self.battery_size = 75
        self.battery_charge = 'Battery'

    def user_input(self):
        print(self.spldlimt1)
        print(self.shlpsped)

    def car_battery_size(self):
        print(self.battery_size)

#FOR A BATTERY A METHOD FROM THE PARENT CLASS.
def capacity_fill_battery_size(self):
    print("This Tesla car is hybrid,so no need any gas to fill now")

Tesla = ElectricCar('Tesla','Hyprid',2019,23)
print(Tesla.get_descriptive_name())

Tesla.car_battery_size()
Tesla.user_input()
Tesla.capacity_fill_gas()

Tesla.batterycar.battery_req()
Tesla.batterycar.get_range()

2019 Tesla Hybrid
75
0
Odometer reading is 20
This Tesla car is hybrid,so no need any gas to fill now
Battery charge required for the electric car and the battery size is 75
Entered the battery size is : 89
89 so, the range is = 150

In [14]: class Battery:
    def __init__(self,battery_charge=75, updbattery_size=100):
        self.battery_charge = battery_charge
        self.battery_size = 0
        self.ent_battery_size= 0
        self.updbattery_size= 0

    def battery_req(self):
        print(f"battery charge required for the electric car and the battery size is {self.battery_c
harge}")

    def get_range(self):
        ent_battery_size = int(input("Entered the battery size : "))
        print(ent_battery_size)
        self.battery_size = ent_battery_size
        if ent_battery_size >=100:
            print(f"({ent_battery_size}) so, the range is = 250")
        else:
            print(f"({ent_battery_size}) so, the range is = 150")

    def updbattery_size(self):
        upg_battery_size = int(input("Entered the battery size is : "))
        print(upg_battery_size)
        self.updbattery_size = upg_battery_size
        if upg_battery_size >= 100:
            print("This battery size will suitable for the electri car")
        else:
            print("It is not suitable")

bat = Battery()
print(bat.battery_req())

Battery charge required for the electric car and the battery size is 75
None

In [13]: class IceCreamStand(Restant):
    def __init__(self,flavors,number_of_table, number_served):
        super().__init__(number_of_table,number_served)
        self.flavors = Flavors

    def name_flavors(self):
        print(f"the name of the flavor is : {self.flavors}")

    def numofpl_served(self):
        print(f"the number of people served {self.number_served}")

    def number_of_table(self):
        print(f"the number of table ordered {self.number_of_table}")

ice = IceCreamStand('Vennila',89,22)
print(ice.numofpl_served())
print(ice.number_of_table())
print(ice.name_flavors())

The number of people served 22
None
The number of table ordered 89
None
The name of the flavor is : Vennila
None

In [39]: #9.7
class Admin(UserDet):
    def __init__(self,privileges):
        self.privileges = privileges
        login_attempts = 100
        super().__init__(login_attempts)
        # self.privileges=privileges()
        incrctmpt=0
        decatempt=0

    def show_privileges(self):
        print("Can add post")
        print(f"Can delete post {self.privileges}")
        print(f"Can ban post {self.privileges}")
        print(self.privileges)

    def user_login(self):
        print(f"Number of login time is {self.login_attempts}")

    def increment_login_attempts(self,incrctmpt):
        self.login_attempts += incrctmpt
        print(f"after increment the login attempts are : {self.login_attempts}")

    def reset_login_attempts(self,decatempt):
        self.login_attempts = decatempt
        print(f"after decrement the login attempt is : 0")

admin1 = Admin('Administrator')
admin1 = Admin('I can add post')
admin1.show_privileges()
admin1.reset_login_attempts()
admin1.increment_login_attempts(10)

#admin1.privileges.password_change()

I can add post
I can delete post
Can add post
Can delete post - Administrator
Can ban post - Administrator
- Administrator
Number of login time is 100
After increment the login attempts are : 110

In [40]: #9.8
class Privileges(Admin):
    def __init__(self,pw):
        self.pw=pw
        # super().__init__(privileges)

    def password_change(self):
        # self.password = UserDet()
        print(password_change(self))
        print(f"Hi {self.pw}")

privs = Privileges('jk')
#privs.show_privileges()
#privs.user_login()
#privs.password_change()

jk
jk

In [18]: from car import ElectricCar,Car
Tesla = ElectricCar('Tesla','Hyprid',2019,23)
print(Tesla.get_descriptive_name())

my_new_car = Car('audi1','a4', 2019,20)
print(my_new_car.get_descriptive_name())

2019 Audi A4
Odometer reading is 20
None
Enter the school area speed limit here to know is it correctly defined or not:8
Decleared speed limit is 40
The entered 8, and this speed limit is not exceeds the defined speed limit
Before increment 0
After increment 10
The gas tank capacity is: 40 Gallons

-----
NameError: name 'ElectricCar' is not defined Traceback (most recent call last)
----> 1 from car import ElectricCar,Car
2 Tesla = ElectricCar('Tesla','Hyprid',2019,23)
3 print(Tesla.get_descriptive_name())
4 my_new_car = Car('audi1','a4', 2019,20)
5 print(my_new_car.get_descriptive_name())

~\car.py in <module>
85
86
--> 90 Tesla = ElectricCar('Tesla','Hyprid',2019,23)
91 print(Tesla.get_descriptive_name())
92

~\car.py in __init__(self, make, model, year, odometer)
74 super().__init__(make, model, year, odometer)
75 self.battery_size = 75
--> 76 self.batterycar = Battery()
77
78 def user_input(self):

NameError: name 'Battery' is not defined

In [11]: import car
capcar = car.Car('audi1', 'a4', 2019,20)
print(capcar.get_descriptive_name())

Tesla = car.ElectricCar('Tesla','Hyprid',2019,23)
print(Tesla.get_descriptive_name())

2019 Audi A4
Odometer reading is 20
None
Enter the school area speed limit here to know is it correctly defined or not:7
Decleared speed limit is 40
The entered 7, and this speed limit is not exceeds the defined speed limit
Before increment 0
After increment 10
The gas tank capacity is: 40 Gallons

-----
NameError: name 'Battery' is not defined Traceback (most recent call last)
----> 1 from car import ElectricCar,Car
2 Tesla = ElectricCar('Tesla','Hyprid',2019,23)
3 print(Tesla.get_descriptive_name())
4 my_new_car = Car('audi1','a4', 2019,20)
5 print(my_new_car.get_descriptive_name())

~\car.py in <module>
85
86
--> 90 Tesla = ElectricCar('Tesla','Hyprid',2019,23)
91 print(Tesla.get_descriptive_name())
92

~\car.py in __init__(self, make, model, year, odometer)
74 super().__init__(make, model, year, odometer)
75 self.battery_size = 75
--> 76 self.batterycar = Battery()
77
78 def user_input(self):

NameError: name 'Battery' is not defined

In [18]: from car import Car
from ElectricCar import ElectricCar
from ElectricCar import Battery

capcar = Car('audi1', 'a4', 2019,20)
print(capcar.get_descriptive_name())

Tesla = ElectricCar('Tesla','Hyprid',2019,23)
print(Tesla.get_descriptive_name())

bat1 = Battery()
print(bat1.battery_req())

2019 Audi A4
Odometer reading is 20
None
Enter the school area speed limit here to know is it correctly defined or not:9
Decleared speed limit is 40
The entered 9, and this speed limit is not exceeds the defined speed limit
Before increment 0
After increment 10
The gas tank capacity is: 40 Gallons

-----
NameError: name 'ElectricCar' is not defined Traceback (most recent call last)
----> 1 from car import Car
2 from ElectricCar import ElectricCar
3 from ElectricCar import Battery
4
5 capcar = Car('audi1', 'a4', 2019,20)

~\car.py in <module>
85
86
--> 90 Tesla = ElectricCar('Tesla','Hyprid',2019,23)
91 print(Tesla.get_descriptive_name())
92

~\car.py in __init__(self, make, model, year, odometer)
74 super().__init__(make, model, year, odometer)
75 self.battery_size = 75
--> 76 self.batterycar = Battery()
77
78 def user_input(self):

NameError: name 'Battery' is not defined

In [20]: from User_infor import UserDet
log1 = UserDet(20)
log1.increment_login_attempts(9)
log1.reset_login_attempts(0)
log1.reset_login_attempts(0)

After increment the login attempts are : 29
After decrement the login attempt is : 0

In [22]: from User_infor import UserDet
log1 = UserDet(20)
log1.increment_login_attempts(9)
log1.reset_login_attempts(0)
admin1 = Admin('Administrator')
admin1 = Admin('I can add post')
admin1 = Admin('I can delete post')
admin1.show_privileges()
print(admin1.privileges)
print(admin12.privileges)
admin1.show_privileges()

After increment the login attempts are : 29
After decrement the login attempt is : 0
I can add post
I can delete post
Can delete post - Administrator
Can ban post - Administrator
- Administrator

In [39]: from random import randint
randint(1,5)
randint(2,3)

Out[39]: 3

In [40]: from random import choice
players = ['kavi','loga','uma','krishna','adhi','1','2','3','4','5']
cho = choice(players)
print(cho)

kaniha

In [14]: from random import randint
class Die:
    def __init__(self,rl=0,sides=6,sides1=6,sides10=10,sides20=20):
        self.sides = sides
        self.rl = rl
        self.sides1 = sides1
        self.sides10 = sides10
        self.sides20 = sides20

    def roll_die(self,sides):
        dice = randint(1,sides)
        print(dice)

    def troll(self,rl):
        while self.rl<=10:
            dice1 = randint(1,self.sides1)
            # print(dice1)
            print(f"The number {rl} roll and the value in the die is: {dice1}")
            rl+=1

    def twent_sided_die(self):
        self.rl = 1
        while self.rl<= 10:
            times20 = randint(1,self.sides20)
            print(f"The number {self.rl} roll and the value in the die is: {times20}")
            self.rl+=1

    def twent_sided_die(self):
        self.rl = 1
        while self.rl<= 10:
            times20 = randint(1,self.sides20)
            print(f"The number {self.rl} roll and the value in the die is: {times20}")
            self.rl+=1

die6 = Die(6)
res = die6.roll_die(6)
roll = die6.roll_die(6)
die6.roll_die(6)

die6.troll(1)

die6.twent_sided_die()

die6.twent_sided_die()

1
5
The number 1 roll and the value in the die is: 6
The number 2 roll and the value in the die is: 6
The number 3 roll and the value in the die is: 2
The number 4 roll and the value in the die is: 3
The number 5 roll and the value in the die is: 3
The number 6 roll and the value in the die is: 1
The number 7 roll and the value in the die is: 1
The number 8 roll and the value in the die is: 1
The number 9 roll and the value in the die is: 2
The number 10 roll and the value in the die is: 10
The number 1 roll and the value in the die is: 8
The number 2 roll and the value in the die is: 6
The number 3 roll and the value in the die is: 2
The number 4 roll and the value in the die is: 2
The number 5 roll and the value in the die is: 8
The number 6 roll and the value in the die is: 8
The number 7 roll and the value in the die is: 9
The number 8 roll and the value in the die is: 9
The number 9 roll and the value in the die is: 6
The number 10 roll and the value in the die is: 12
The number 1 roll and the value in the die is: 8
The number 2 roll and the value in the die is: 10
The number 3 roll and the value in the die is: 20
The number 4 roll and the value in the die is: 10
The number 5 roll and the value in the die is: 9
The number 6 roll and the value in the die is: 7
The number 7 roll and the value in the die is: 12
The number 8 roll and the value in the die is: 9
The number 9 roll and the value in the die is: 10
The number 10 roll and the value in the die is: 14

In [66]: from random import choice
class Lottery:
    #this class used 2 methods for choosing the values from the list and tuple using choice() method.
    #it is imported random library.

    def __init__(self,names,my_ticket,lotto_tuple,my_ticket1):
        self.names = names
        self.my_ticket = my_ticket
        self.lotto_tuple = lotto_tuple
        self.my_ticket1 = my_ticket1

    def list_list(self):
        self.names = ('kavi','loga','uma','krishna','adhi','1','2','3','4','5')
        self = choice(self.names)
        reit1 = choice(self.names)
        reit2 = choice(self.names)
        reit3 = choice(self.names)
        self.my_ticket = reit1 + reit1 + reit2 + reit3
        print(self.my_ticket)

        print(f"Any ticket matching for the following numbers and names wins a prize {reit1,reit2,reit3}")
        # print(self.my_ticket1)

#TUPL VALUE IS IMMUTABLE IT MEANS IF CAN'T CHANGE ONCE WE DECLARED IN TUPLE BUT LIST CAN CHANGE.

    def lotto_tuple(self):
        self.lotto_tuple = ('kavi','loga','uma','krishna','adhi','1','2','3','4','5')

        reit4 = choice(self.lotto_tuple)
        print(f"result {reit4}")

        for lottp1 in choice(self.lotto_tuple):
            if lottp1 == reit4:
                print(f"Equal 1st value {lottp1}")
            else:
                print(reit4)

        reit5 = choice(self.lotto_tuple)
        print(f"result {reit5}")

        reit6 = choice(self.lotto_tuple)
        print(f"result {reit6}")

        for lottp2 in choice(self.lotto_tuple):
            if lottp2 == reit6:
                print(f"Equal 2nd value {lottp2}")
            else:
                print(reit6)

        reit7 = choice(self.lotto_tuple)
        print(f"result {reit7}")

        for lottp3 in choice(self.lotto_tuple):
            if lottp3 == reit7:
                print(f"Equal 3rd value {lottp3}")
            else:
                print(reit7)

        self.my_ticket1 = reit4 + reit5 + reit6 + reit7
        print(f"the choice of value from the tuple {self.my_ticket1}")

        for lottp4 in self.lotto_tuple:
            if lottp4 == self.my_ticket1:
                print(f"Equal 4th value {lottp4}")
            else:
                print(reit4)

        reit8 = choice(self.lotto_tuple)
        print(f"result {reit8}")

        reit9 = choice(self.lotto_tuple)
        print(f"result {reit9}")

        for lottp1 in choice(self.lotto_tuple):
            if lottp1 == reit9:
                print(f"Equal 1st value {lottp1}")
            else:
                print(reit9)

        reit10 = choice(self.lotto_tuple)
        print(f"result {reit10}")

        reit11 = choice(self.lotto_tuple)
        print(f"result {reit11}")

        reit12 = choice(self.lotto_tuple)
        print(f"result {reit12}")

        reit13 = choice(self.lotto_tuple)
        print(f"result {reit13}")

        reit14 = choice(self.lotto_tuple)
        print(f"result {reit14}")

        reit15 = choice(self.lotto_tuple)
        print(f"result {reit15}")

        reit16 = choice(self.lotto_tuple)
        print(f"result {reit16}")

        reit17 = choice(self.lotto_tuple)
        print(f"result {reit17}")

        reit18 = choice(self.lotto_tuple)
        print(f"result {reit18}")

        reit19 = choice(self.lotto_tuple)
        print(f"result {reit19}")

        reit20 = choice(self.lotto_tuple)
        print(f"result {reit20}")

        reit21 = choice(self.lotto_tuple)
        print(f"result {reit21}")

        reit22 = choice(self.lotto_tuple)
        print(f"result {reit22}")

        reit23 = choice(self.lotto_tuple)
        print(f"result {reit23}")

        reit24 = choice(self.lotto_tuple)
        print(f"result {reit24}")

        reit25 = choice(self.lotto_tuple)
        print(f"result {reit25}")

        reit26 = choice(self.lotto_tuple)
        print(f"result {reit26}")

        reit27 = choice(self.lotto_tuple)
        print(f"result {reit27}")

        reit28 = choice(self.lotto_tuple)
        print(f"result {reit28}")

        reit29 = choice(self.lotto_tuple)
        print(f"result {reit29}")

        reit30 = choice(self.lotto_tuple)
        print(f"result {reit30}")

        reit31 = choice(self.lotto_tuple)
        print(f"result {reit31}")

        reit32 = choice(self.lotto_tuple)
        print(f"result {reit32}")

        reit33 = choice(self.lotto_tuple)
        print(f"result {reit33}")

        reit34 = choice(self.lotto_tuple)
        print(f"result {reit34}")

        reit35 = choice(self.lotto_tuple)
        print(f"result {reit35}")

        reit36 = choice(self.lotto_tuple)
        print(f"result {reit36}")

        reit37 = choice(self.lotto_tuple)
        print(f"result {reit37}")

        reit38 = choice(self.lotto_tuple)
        print(f"result {reit38}")

        reit39 = choice(self.lotto_tuple)
        print(f"result {reit39}")

        reit40 = choice(self.lotto_tuple)
        print(f"result {reit40}")

        reit41 = choice(self.lotto_tuple)
        print(f"result {reit41}")

        reit42 = choice(self.lotto_tuple)
        print(f"result {reit42}")

        reit43 = choice(self.lotto_tuple)
        print(f"result {reit43}")

        reit44 = choice(self.lotto_tuple)
        print(f"result {reit44}")

        reit45 = choice(self.lotto_tuple)
        print(f"result {reit45}")

        reit46 = choice(self.lotto_tuple)
        print(f"result {reit46}")

        reit47 = choice(self.lotto_tuple)
        print(f"result {reit47}")

        reit48 = choice(self.lotto_tuple)
        print(f"result {reit48}")

        reit49 = choice(self.lotto_tuple)
        print(f"result {reit49}")

        reit50 = choice(self.lotto_tuple)
        print(f"result {reit50}")

        reit51 = choice(self.lotto_tuple)
        print(f"result {reit51}")

        reit52 = choice(self.lotto_tuple)
        print(f"result {reit52}")

        reit53 = choice(self.lotto_tuple)
        print(f"result {reit53}")

        reit54 = choice(self.lotto_tuple)
        print(f"result {reit54}")

        reit55 = choice(self.lotto_tuple)
        print(f"result {reit55}")

        reit56 = choice(self.lotto_tuple)
        print(f"result {reit56}")

        reit57 = choice(self.lotto_tuple)
        print(f"result {reit57}")

        reit58 = choice(self.lotto_tuple)
        print(f"result {reit58}")

        reit59 = choice(self.lotto_tuple)
        print(f"result {reit59}")

        reit60 = choice(self.lotto_tuple)
        print(f"result {reit60}")

        reit61 = choice(self.lotto_tuple)
        print(f"result {reit61}")

        reit62 = choice(self.lotto_tuple)
        print(f"result {reit62}")

        reit63 = choice(self.lotto_tuple)
        print(f"result {reit63}")

        reit64 = choice(self.lotto_tuple)
        print(f"result {reit64}")

        reit65 = choice(self.lotto_tuple)
        print(f"result {reit65}")

        reit66 = choice(self.lotto_tuple)
        print(f"result {reit66}")

        reit67 = choice(self.lotto_tuple)
        print(f"result {reit67}")

        reit68 = choice(self.lotto_tuple)
        print(f"result {reit68}")

        reit69 = choice(self.lotto_tuple)
        print(f"result {reit69}")

        reit70 = choice(self.lotto_tuple)
        print(f"result {reit70}")

        reit71 = choice(self.lotto_tuple)
        print(f"result {reit71}")

        reit72 = choice(self.lotto_tuple)
        print(f"result {reit72}")

        reit73 = choice(self.lotto_tuple)
        print(f"result {reit73}")

        reit74 = choice(self.lotto_tuple)
        print(f"result {reit74}")

        reit75 = choice(self.lotto_tuple)
        print(f"result {reit75}")

        reit76 = choice(self.lotto_tuple)
        print(f"result {reit76}")

        reit77 = choice(self.lotto_tuple)
        print(f"result {reit77}")

        reit78 = choice(self.lotto_tuple)
        print(f"result {reit78}")

        reit79 = choice(self.lotto_tuple)
        print(f"result {reit79}")

        reit80 = choice(self.lotto_tuple)
        print(f"result {reit80}")

        reit81 = choice(self.lotto_tuple)
        print(f"result {reit81}")

        reit82 = choice(self.lotto_tuple)
        print(f"result {reit82}")

        reit83 = choice(self.lotto_tuple)
        print(f"result {reit83}")

        reit84 = choice(self.lotto_tuple)
        print(f"result {reit84}")

        reit85 = choice(self.lotto_tuple)
        print(f"result {reit85}")

        reit86 = choice(self.lotto_tuple)
        print(f"result {reit86}")

        reit87 = choice(self.lotto_tuple)
        print(f"result {reit87}")

        reit88 = choice(self.lotto_tuple)
        print(f"result {reit88}")

        reit89 = choice(self.lotto_tuple)
        print(f"result {reit89}")

        reit90 = choice(self.lotto_tuple)
        print(f"result {reit90}")

        reit91 = choice(self.lotto_tuple)
        print(f"result {reit91}")

        reit92 = choice(self.lotto_tuple)
        print(f"result {reit92}")

        reit93 = choice(self.lotto_tuple)
        print(f"result {reit93}")

        reit94 = choice(self.lotto_tuple)
        print(f"result {reit94}")

        reit95 = choice(self.lotto_tuple)
        print(f"result {reit95}")

        reit96 = choice(self.lotto_tuple)
        print(f"result {reit96}")

        reit97 = choice(self.lotto_tuple)
        print(f"result {reit97}")

        reit98 = choice(self.lotto_tuple)
        print(f"result {reit98}")

        reit99 = choice(self.lotto_tuple)
        print(f"result {reit99}")

        reit100 = choice(self.lotto_tuple)
        print(f"result {reit100}")

        reit101 = choice(self.lotto_tuple)
        print(f"result {reit101}")

        reit102 = choice(self.lotto_tuple)
        print(f"result {reit102}")

        reit103 = choice(self.lotto_tuple)
        print(f"result {reit103}")

        reit104 = choice(self.lotto_tuple)
        print(f"result {reit104}")

        reit105 = choice(self.lotto_tuple)
        print(f"result {reit105}")

        reit106 = choice(self.lotto_tuple)
        print(f"result {reit106}")

        reit107 = choice(self.lotto_tuple)
        print(f"result {reit107}")

        reit108 = choice(self.lotto_tuple)
        print(f"result {reit108}")

        reit109 = choice(self.lotto_tuple)
        print(f"result {reit109}")

        reit110 = choice(self.lotto_tuple)
        print(f"result {reit110}")

        reit111 = choice(self.lotto_tuple)
        print(f"result {reit111}")

        reit112 = choice(self.lotto_tuple)
        print(f"result {reit112}")

        reit113 = choice(self.lotto_tuple)
        print(f"result {reit113}")

        reit114 = choice(self.lotto_tuple)
        print(f"result {reit114}")

        reit115 = choice(self.lotto_tuple)
        print(f"result {reit115}")

        reit116 = choice(self.lotto_tuple)
        print(f"result {reit116}")

        reit117 = choice(self.lotto_tuple)
        print(f"result {reit117}")

        reit118 = choice(self.lotto_tuple)
        print(f"result {reit118}")

        reit119 = choice(self.lotto_tuple)
        print(f"result {reit119}")

        reit120 = choice(self.lotto_tuple)
        print(f"result {reit120}")

        reit121 = choice(self.lotto_tuple)
        print(f"result {reit121}")

        reit122 = choice(self.lotto_tuple)
        print(f"result {reit122}")

        reit123 = choice(self.lotto_tuple)
        print(f"result {reit123}")

        reit124 = choice(self.lotto_tuple)
        print(f"result {reit124}")

        reit125 = choice(self.lotto_tuple)
        print(f"result {reit125}")

        reit126 = choice(self.lotto_tuple)
        print(f"result {reit126}")

        reit127 = choice(self.lotto_tuple)
        print(f"result {reit127}")

        reit128 = choice(self.lotto_tuple)
        print(f"result {reit128}")

        reit129 = choice(self.lotto_tuple)
        print(f"result {reit129}")

        reit130 = choice(self.lotto_tuple)
        print(f"result {reit130}")

        reit131 = choice(self.lotto_tuple)
        print(f"result {reit131}")

        reit132 = choice(self.lotto_tuple)
        print(f"result {reit132}")

        reit133 = choice(self.lotto_tuple)
        print(f"result {reit133}")

        reit134 = choice(self.lotto_tuple)
        print(f"result {reit134}")

        reit135 = choice(self.lotto_tuple)
        print(f"result {reit135}")

        reit136 = choice(self.lotto_tuple)
        print(f"result {reit136}")

        reit137 = choice(self.lotto_tuple)
        print(f"result {reit137}")

        reit138 = choice(self.lotto_tuple)
        print(f"result {reit138}")

        reit139 = choice(self.lotto_tuple)
        print(f"result {reit139}")

        reit140 = choice(self.lotto_tuple)
        print(f"result {reit140}")

        reit141 = choice(self.lotto_tuple)
        print(f"result {reit141}")

        reit142 = choice(self.lotto_tuple)
        print(f"result {reit142}")

        reit143 = choice(self.lotto_tuple)
        print(f"result {reit143}")

        reit144 = choice(self.lotto_tuple)
        print(f"result {reit144}")

        reit145 = choice(self.lotto_tuple)
        print(f"result {reit145}")

        reit146 = choice(self.lotto_tuple)
        print(f"result {reit146}")

        reit147 = choice(self.lotto_tuple)
        print(f"result {reit147}")

        reit148 = choice(self.lotto_tuple)
        print(f"result {reit148}")

        reit149 = choice(self.lotto_tuple)
        print(f"result {reit149}")

        reit150 = choice(self.lotto_tuple)
        print(f"result {reit150}")

        reit151 = choice(self.lotto_tuple)
        print(f"result {reit151}")

        reit152 = choice(self.lotto_tuple)
        print(f"result {reit152}")

        reit153 = choice(self.lotto_tuple)
        print(f"result {reit153}")

        reit154 = choice(self.lotto_tuple)
        print(f"result {reit154}")

        reit155 = choice(self.lotto_tuple)
        print(f"result {reit155}")

        reit156 = choice(self.lotto_tuple)
        print(f"result {reit156}")

        reit157 = choice(self.lotto_tuple)
        print(f"result {reit157}")

        reit158 = choice(self.lotto_tuple)
        print(f"result {reit158}")

        reit159 = choice(self.lotto_tuple)
        print(f"result {reit159}")

        reit160 = choice(self.lotto_tuple)
        print(f"result {reit160}")

        reit161 = choice(self.lotto_tuple)
        print(f"result {reit161}")

        reit162 = choice(self.lotto_tuple)
        print(f"result {reit162}")

        reit163 = choice(self.lotto_tuple)
        print(f"result {reit163}")

        reit164 = choice(self.lotto_tuple)
        print(f"result {reit164}")

        reit165 = choice(self.lotto_tuple)
        print(f"result {reit165}")

        reit166 = choice(self.lotto_tuple)
        print(f"result {reit166}")

        reit167 = choice(self.lotto_tuple)
        print(f"result {reit167}")

        reit168 = choice(self.lotto_tuple)
        print(f"result {reit168}")

        reit169 = choice(self.lotto_tuple)
        print(f"result {reit169}")

        reit170 = choice(self.lotto_tuple)
        print(f"result {reit170}")

        reit171 = choice(self.lotto_tuple)
        print(f"result {reit171}")

        reit172 = choice(self.lotto_tuple)
        print(f"result {reit172}")

        reit173 = choice(self.lotto_tuple)
        print(f"result {reit173}")

        reit174 = choice(self.lotto_tuple)
        print(f"result {reit174}")

        reit175 = choice(self.lotto_tuple)
        print(f"result {reit175}")

        reit176 = choice(self.lotto_tuple)
        print(f"result {reit176}")

        reit177 = choice(self.lotto_tuple)
        print(f"result {reit177}")

        reit178 = choice(self.lotto_tuple)
        print(f"result {reit178}")

        reit179 = choice(self.lotto_tuple)
        print(f"result {reit179}")

        reit180 = choice(self.lotto_tuple)
        print(f"result {reit180}")

        reit181 = choice(self.lotto_tuple)
        print(f"result {reit181}")

        reit182 = choice(self.lotto_tuple)
        print(f"result {reit182}")

        reit183 = choice(self.lotto_tuple)
        print(f"result {reit183}")

        reit184 = choice(self.lotto_tuple)
        print(f"result {reit184}")

        reit185 = choice(self.lotto_tuple)
        print(f"result {reit185}")

        reit186 = choice(self.lotto_tuple)
        print(f"result {reit186}")

        reit187 = choice(self.lotto_tuple)
        print(f"result {reit187}")

        reit188 = choice(self.lotto_tuple)
        print(f"result {reit188}")

        reit189 = choice(self.lotto_tuple)
        print(f"result {reit189}")

        reit190 = choice(self.lotto_tuple)
        print(f"result {reit190}")

        reit191 = choice(self.lotto_tuple)
        print(f"result {reit191}")

        reit192 = choice(self.lotto_tuple)
        print(f"result {reit192}")

        reit193 = choice(self.lotto_tuple)
        print(f"result {reit193}")

        reit194 = choice(self.lotto_tuple)
        print(f"result {reit194}")

        reit195 = choice(self.lotto_tuple)
        print(f"result {reit195}")

        reit196 = choice(self.lotto_tuple)
        print(f"result {reit196}")

        reit197 = choice(self.lotto_tuple)
        print(f"result {reit197}")

        reit198 = choice(self.lotto_tuple)
        print(f"result {reit198}")

        reit199 = choice(self.lotto_tuple)
        print(f"result {reit199}")

        reit200 = choice(self.lotto_tuple)
        print(f"result {reit200}")

        reit201 = choice(self.lotto_tuple)
        print(f"result {reit201}")

        reit202 = choice(self.lotto_tuple)
        print(f"result {reit202}")

        reit203 = choice(self.lotto_tuple)
        print(f"result {reit203}")

        reit204 = choice(self.lotto_tuple)
        print(f"result {reit204}")

        reit205 = choice(self.lotto_tuple)
        print(f"result {reit205}")

        reit206 = choice(self.lotto_tuple)
        print(f"result {reit206}")

        reit207 = choice(self.lotto_tuple)
        print(f"result {reit207}")

        reit208 = choice(self.lotto_tuple)
        print(f"result {reit208}")

        reit209 = choice(self.lotto_tuple)
        print(f"result {reit209}")

        reit210 = choice(self.lotto_tuple)
        print(f"result {reit210}")

        reit211 = choice(self.lotto_tuple)
        print(f"result {reit211}")

        reit212 = choice(self.lotto_tuple)
        print(f"result {reit212}")

        reit213 = choice(self.lotto_tuple)
        print(f"result {reit213}")

        reit214 = choice(self.lotto_tuple)
        print(f"result {reit214}")

        reit215 = choice(self.lotto_tuple)
        print(f"result {reit215}")

        reit216 = choice(self.lotto_tuple)
        print(f"result {reit216}")

        reit217 = choice(self.lotto_tuple)
        print(f"result {reit217}")

        reit218 = choice(self.lotto_tuple)
        print(f"result {reit218}")

        reit219 = choice(self.lotto_tuple)
        print(f"result {reit219}")

        reit220 = choice(self.lotto_tuple)
        print(f"result {reit220}")

        reit221 = choice(self.lotto_tuple)
        print(f"result {reit221}")

        reit222 = choice(self.lotto_tuple)
        print(f"result {reit222}")

        reit223 = choice(self.lotto_tuple)
        print(f"result {reit223}")

        reit224 = choice(self.lotto_tuple)
        print(f"result {reit224}")

        reit225 = choice(self.lotto_tuple)
        print(f"result {reit225}")

        reit226 = choice(self.lotto_tuple)
        print(f"result {reit226}")

        reit227 = choice(self.lotto_tuple)
        print(f"result {reit227}")

        reit228 = choice(self.lotto_tuple)
        print(f"result {reit228}")

        reit229 = choice(self.lotto_tuple)
        print(f"result {reit229}")

        reit230 = choice(self.lotto_tuple)
        print(f"result {reit230}")

        reit231 = choice(self.lotto_tuple)
        print(f"result {reit231}")

        reit232 = choice(self.lotto_tuple)
        print(f"result {reit232}")

        reit233 = choice(self.lotto_tuple)
        print(f"result {reit233}")

        reit234 = choice(self.lotto_tuple)
        print(f"result {reit234}")

        reit235 = choice(self.lotto_tuple)
        print(f"result {reit235}")

        reit236 = choice(self.lotto_tuple)
        print(f"result {reit236}")

        reit237 = choice(self.lotto_tuple)
        print(f"result {reit237}")

        reit238 = choice(self.lotto_tuple)
        print(f"result {reit238}")

        reit239 = choice(self.lotto_tuple)
        print(f"result {reit239}")

        reit240 = choice(self.lotto_tuple)
        print(f"result {reit240}")

        reit241 = choice(self.lotto_tuple)
        print(f"result {reit241}")

        reit242 = choice(self.lotto_tuple)
        print(f"result {reit242}")

        reit243 = choice(self.lotto_tuple)
        print(f"result {reit243}")

        reit244 = choice(self.lotto_tuple)
        print(f"result {reit244}")

        reit245 = choice(self.lotto_tuple)
        print(f"result {reit245}")

        reit246 = choice(self.lotto_tuple)
        print(f"result {reit246}")

        reit247 = choice(self.lotto_tuple)
        print(f"result {reit247}")

        reit248 = choice(self.lotto_tuple)
        print(f"result {reit248}")

        reit249 = choice(self.lotto_tuple)
        print(f"result {reit249}")

        reit250 = choice(self.lotto_tuple)
        print(f"result {reit250}")

        reit251 = choice(self.lotto_tuple)
        print(f"result {reit251}")

        reit252 = choice(self.lotto_tuple)
        print(f"result {reit252}")

        reit253 = choice(self.lotto_tuple)
        print(f"result {reit253}")

        reit254 = choice(self.lotto_tuple)
        print(f"result {reit254}")

        reit255 = choice(self.lotto_tuple)
        print(f"result {reit255}")

        reit256 = choice(self.lotto_tuple)
        print(f"result {reit256}")

        reit257 = choice(self.lotto_tuple)
        print(f"result {reit257}")

        reit258 = choice(self.lotto_tuple)
        print(f"result {reit258}")

        reit259 = choice(self.lotto_tuple)
        print(f"result {reit259}")

        reit260 = choice(self.lotto_tuple)
        print(f"result {reit260}")

        reit261 = choice(self.lotto_tuple)
        print(f"result {reit261}")

        reit262 = choice(self.lotto_tuple)
        print(f"result {reit262}")

        reit263 = choice(self.lotto_tuple)
        print(f"result {reit263}")

        reit264 = choice(self.lotto_tuple)
        print(f"result {reit264}")

        reit
```