Python Assignment book by Naveen

Basic Questions and Programs

1. What is Python.
2. What is Programming language.
3. What is software and how many types.
4. What is python script.
5. What kind of applications we can develop using python.
6. How to develop standalone application in python.
7. How to develop distributed applications in python
8. Is python a case sensitive language
9. where we can use python.
10. WAP to Display Welcome To Sathya Technologies
11. WAP to accept the Student Name and Display a message Welcome Student Name
12. WAP to accept two numbers and perform addition, Subtraction, Multiplication and Division
13. WAP to find the type Of a Variable
14. WAP to accept student first name and last name and Display the student full Name
15. WAP to accept a str value and convert the string value to int and print int value.
16. WAP to accept a double value and convert the double value to str and print string value

17. WAP to accept a str value and convert the str value to double and print double value.

18. WAP to accept student no, student name, marks1, marks2, marks3 from command prompt and calculate total marks, average marks and print.

19. WAP to input two numbers and swap those values.

20. WAP to input Employee id_no, name, Basic salary, HRA, DA and Find out the Gross Salary.

Questions on Operators and Programs

1. How many types of operators in python.
2. What is the difference between logical and 'and' bitwise and.
3. What is the difference between logical or and bitwise or.
4. What are identity Operators.
5. Output of print(9//2)
6. What is the type a when a = 1,00,000
7. What is the type of 'inf'?
   a) Boolean
   b) Integer
   c) Float
   d) Complex
8. What does ~~~~~~5 evaluate to?
   a) +5
   b) -11
   c) +11
   d) -5
9. What is the purpose of not in operator.
10. What is the purpose pass statement in python.
11. Which function overloads the >> operator.

12. Output of format(10/3) and what is the return type.
13. Output of "Sathya" > "sathya".
14. Output of 10 and 10.
15. Output of 10 and 0.
16. Output of 0 and 10.
17. Output of 0 and 0.
18. Output of 0 or 10.
19. Which is the correct operator for power(x^y)?
    a) X^y
    b) X**y
    c) X^^y
    d) None of the mentioned
20. Which one of these is floor division?
    a) /
    b) //
    c) %
    d) None of the mentioned
21. What is the order of precedence in python?
    i) Parentheses ii) Exponential iii) Multiplication iv) Division
    v) Addition vi) Subtraction
    a) i,ii,iii,iv,v,vi
    b) ii,i,iii,iv,v,vi
    c) ii,i,iv,iii,v,vi
    d) i,ii,iii,iv,vi,v
22. Mathematical operations can be performed on a string. State whether true or false.
    a) True
    b) False
23. Operators with the same precedence are evaluated in which manner?
    a) Left to Right

b) Right to Left

c) Can't say

24. What is the output of this expression, 3*1**3?

a) 27

b) 9

c) 3

d) 1

25. Which one of the following have the same precedence?

a) Addition and Subtraction

b) Multiplication and Division

c) Both Addition and Subtraction AND Multiplication and Division

d) None of the mentioned

26. The expression Int(x) implies that the variable x is converted to integer. State whether true or false.

a) True

b) False

27. Which one of the following have the highest precedence in the expression?

a) Exponential

b) Addition

c) Multiplication

d) Parentheses

28. Which of the following is invalid?

a) _a = 1

b) __a = 1

c) __str__ = 1

d) none of the mentioned

29. Which of the following is an invalid variable?

a) my_string_1

b) 1st_string

c) foo

d) _

30. Which of the following is not a keyword?

a) eval

b) assert

c) nonlocal

d) pass

31. Which of the following is an invalid statement?

a) abc = 1,000,000

b) a b c = 1000 2000 3000

c) a,b,c = 1000, 2000, 3000

d) a_b_c = 1,000,000

32. Which of the following cannot be a variable?

a) __init__

b) in

c) it

d) on

Questions on Flow Control

(if, if-else, if-elif-else, nested-if, for and while)

1. WAP input two numbers and find out the biggest number.
2. WAP input three numbers and find out the biggest number.
3. WAP input any number and check it is positive or negative.
4. Given an int n, return the absolute difference between n and 21, except return double the absolute difference if n is over 21.
   diff21(19) → 2
   diff21(10) → 11
   diff21(21) → 0
5. WAP input any positive number and check it is even or odd

6. Write a program input customer name and slab type(i/c/r) calculate units consumed
   Conditions: If slab type is industry then unit rate is 5.25/-
   If slab type is commercial then unit rate is 4.00/-
   If slab type is residence then unit rate is 3.08/-
   Calculate total bill.
7. Write a program input Employee name, salary, Designation(m/a/c) .
   If designation is manager then bonus is 20% on his salary
   If designation is analyst then bonus is 10% on his salary
   If designation is clerk then bonus is 5% on his salary.
   Calculate Total salary.
8. Given 2 ints, a and b, return True if one if them is 10 or if their sum is 10.
   makes10(9, 10) → True
   makes10(9, 9) → False
   makes10(1, 9) → True
9. Given 2 int values, return True if one is negative and one is positive. Except if the parameter "negative" is True, then return True only if both are negative
   pos_neg(1, -1, False) → True
   pos_neg(-1, 1, False) → True
   pos_neg(-4, -5, True) → True
10. We have a loud talking parrot. The "hour" parameter is the current hour time in the range 0..23. We are in trouble if the parrot is talking and the hour is before 7 or after 20. Return True if we are in trouble.
    parrot_trouble(True, 6) → True
    parrot_trouble(True, 7) → False
    parrot_trouble(False, 6) → False
11. Write a program input Total Marks.

If tm>360 then print 'First class'

If Tm>=300 and tm<360 then print 'second class'

If tm<300 then print 'third class'

12. We have two monkeys, a and b, and the parameters a_smile and b_smile indicate if each is smiling. We are in trouble if they are both smiling or if neither of them is smiling. Return True if we are in trouble

monkey_trouble(True, True) → True

monkey_trouble(False, False) → True

monkey_trouble(True, False) → False

13. Given two int values, return their sum. Unless the two values are the same, then return double their sum.

sum_double(1, 2) → 3

sum_double(3, 2) → 5

sum_double(2, 2) → 8

14. The parameter weekday is True if it is a weekday, and the parameter vacation is True if we are on vacation. We sleep in if it is not a weekday or we're on vacation. Return True if we sleep in.

sleep_in(False, False) → True

sleep_in(True, False) → False

sleep_in(False, True) → True

Loop's Questions

15. Print Sathya Technologies for 10 times
16. Print 1 to 10 numbers
17. Print even numbers from 1 to 10
18. Print odd numbers from 1 to 10

19. Print sum of all the numbers 1 to 10
20. Print sum of all even numbers from 1 to 10
21. Print sum of all odd numbers from 1 to 10
22. Find the factorial of a given number
23. WAP to Check the number is prime or not
24. Print the number from 10 to 1
25. Input any 10 numbers find out no of even numbers and no of odd numbers
26. Input any 10 numbers find out no of positive numbers and no of negative numbers
27. Input any 10 numbers find the first biggest and second biggest number
28. Input any 10 find number of prime numbers.
29. Input any 4 digit number and reverse it
30. Input any 4 digit number and find out the sum of the digits
31. Input any 4 digit number and find out the sum of first and last digit
32. Input any 4 digit number and find out the sum of middle digits
33. Input any 4 digit number and find out the difference of all digits
34. Print the Multiplication table for any given Number.
35. What is the output of the following

```
string = "my name is Naveen"
for i in string:
    print (i, end=", ")
```

36. What is the output of the following

```
for i in range(10):
    if i == 5:
```

```
                break
        else:
                print(i)
else:
        print("Here")
```

37. What is the output of the following

```
i = 0
while i < 3:
        print(i,end="")
        i += 1
else:
        print(0,end="")
```

38. Given a non-empty string like "Code" return a string like "CCoCodCode".

string_splosion('Code') → 'CCoCodCode'
string_splosion('abc') → 'aababc'
string_splosion('ab') → 'aab'

39. Given a string, return a new string made of every other char starting with the first, so "Hello" yields "Hlo".

string_bits('Hello') → 'Hlo'
string_bits('Hi') → 'H'
string_bits('Heeololeo') → 'Hello'

40. Given a string and a non-negative int n, we'll say that the front of the string is the first 3 chars, or whatever is there if the string is less than length 3. Return n copies of the front.

front_times('Chocolate', 2) → 'ChoCho'
front_times('Chocolate', 3) → 'ChoChoCho'
front_times('Abc', 3) → 'AbcAbcAbc'

41. Given a string and a non-negative int n, return a larger string that is n copies of the original string.

    string_times('Hi', 2) → 'HiHi'

    string_times('Hi', 3) → 'HiHiHi'

    string_times('Hi', 1) → 'Hi'

42. Input any number check it is strong or not

    What is strong number: a number is called strong number if sum of the factorial of its digits is equal to number itself .

    Ex:145=1! + 4! + 5!=1+24+120

43. Input any number check it is palindrome

    What is palindrome: A word, phrase or sequence that reads the same backward as forward

    Ex:121,242,madam,

44. Input any number check armstrong or not

    What is Armstrong number: it is the sum of the cubes of its digits is equal to the number itself

    Ex:153=1^3+5^3+3^3=1+125+27=153

45. Input any number perfect number or not.

    What is perfect number it is a positive integer i.e equal to the sum of its proper positive divisors

    Ex : 6 Divisors of 6 are 1,2,3 1+2+3=6

Datatypes in Python

List

1) What is list
2) What is the purpose of list
3) Real time example on a list
4) What type of elements we can store in list

5) How to access list elements

6) What is forward and backward index

7) List allow duplicate objects

8) How to find length of a list

9) What operators we can use on list

10) Can we apply "+" operator on list and int

11) Can we apply "+" operator on list and list

12) If we apply "+" operator on list and list what will happen

13) What happens if we apply * operator on list

14)

L1=5	L1=[1,2,3]
L1*3	l1*3
Output ?	output ?

15) If we give out of index value in list, what happens

16) What is slice and define the Syntax

17) If we give invalid index in slicing expression what happens

18) What is the use of "in" operator in list

19) How to add an element at end of a list

20) How to insert an element in specific location in a list

21) How to arrange list element in ascending order

22) How to arrange list elements in descending order

23) How to remove an element in list

24) Difference between remove method , del statement and pop method

25) How to find lowest and highest values in a list

26) How to find sum of all elements in a list

27) How to compare two lists

28) How to add an element to empty list like

29) Can we copy one list to another list? If yes How.

30) What is two dimensional lists? When to use it

31) How to create two dimensional lists

32) Can we convert an integer object to list

33) What type of objects we can convert into list

34) Difference between append( ) and extend( )

35) How to clone or copy of a list

36) How to count occurrences of a list

37) How to split a list into evenly sized chunks

38) How to get an intersection of two list

39) How to remove duplicate elements in a list

40) How to create flat lists our of lists

Tuple

1) What is tuple
2) What is the purpose of tuple and when to use with a real time example
3) How to declare tuple
4) When to use tuple
5) Can we modify tuple data
6) Can we declare empty tuple
7) Can we declare a tuple with single element, if yes how, if no why
8) How to access tuples
9) What is packing and unpacking
10) Different types of methods in tuples
11) What is the use of count( ) method
12) What is the use of index( ) method
13) If I want to modify tuple, what to do
14) How to convert tuple to list
15) How to convert list to tuple

16)

X=(10)	x=(10,)
Print(X*3)	print(x*3)
Output : ?	output : ?

SET

1) What is set
2) Can we store duplicate elements in sets
3) Can we modify set items
4) How to create sets
5) Can we create an empty sets by using { }
6) How to create empty sets
7) Can we access set elements by using index
8) Can we access set elements by using slice
9) How to access set elements
10) If we store duplicate elements in set what happens
11) How to add an element to set
12) Can we add new element in specific position
13) How to remove an element in set
14) How to work with remove( )
15) How to work with pop(
16) How to work with discard( )
17) What is the difference between discard( ) and remove( )
18) How to update set
19) What is the difference between add( ) and update( )
20) How to delete all the elements in a set

21) How to get common elements in two sets

22) What is meaning of " & " operator in sets

23) How to get all the elements in two sets

24) What is meaning of " | " operator in sets

25) How many ways to get difference in sets and explain

26) What is symmetric difference

27) Can we use for loop on sets to print set values

28) print(set("sathya")) output?

29) for x in set(" sathya "):

print(x,end=" ")

30) Use of forzenset

31) How to convert any iterable object as set

Dictionary

1) What is Dictionary
2) When to use dictionary
3) How to store elements in dictionary
4) Can we declare duplicates elements as a keys in dictionary
5) Can we declare duplicate element as a value in dictionary
6) Can we declare empty set
7) Can we change dictionary

8) While performing modification , If key is not available then what happens

9) Can we change keys in dictionary

10) How to delete elements in dictionary

11) While deleting element in dictionary, if the key is not available what happens

12) How to avoid Exception

13) How to check length of a dictionary

14) Can we declare dictionary values as list

```
d1={   "Kohli"    :  [14,85,149],
       "Dhoni"    :  [120,67,50],
       "Rohit"    :  [60,30,8],
       "Rahul"    :  [110,90,56],
       "Rahane" : [60,30,80]     }
```

15) Can we apply loop on dictionary? If yes how?

16) How to delete all the elements in dictionary

17) What is the use of get method

18) How to get all the keys in a dictionary

19) How to get deleted item in dictionary

String

1) What is String

2) How to Access individual characters in a string

3) Given me a real time example for processing individual characters in strings

4) What's the index of first character in a string

5) If a string has 10 characters, then what's the index of last character

6) If we try to access a character which is out of range what happens

7) How can you find length of a string

8) qualification=" B.Tech" qualification[0]='M' What is the output

9) What is the meaning of immutable

10) Can we apply slicing on strings

11) If we give invalid index in slicing, what happens

12) How to check whether given string contains any symbols or not

13) How to check whether given string contains only alphabet or not

14) How to check whether given string contains only digits or not

15) How to check our string is in lower case or not

16) How to check our string is in upper case or not

17) How to check our string contains only whitespaces

18) How to convert our string in to lower case

19) How to delete spaces that are available in left side of a string

20) How to delete spaces that are available in right side of a string

21) How to delete spaces that are available in both left side and right side of a string

22) How to change existing string with a new string

23) How to find a string ends with specific string

24) How to check whether a given string starts with a specific string or not

Logical Questions on Datatypes

1. Given 2 arrays of ints, a and b, return True if they have the same first element or they have the same last element. Both arrays will be length 1 or more.

common_end([1, 2, 3], [7, 3]) → True
common_end([1, 2, 3], [7, 3, 2]) → False
common_end([1, 2, 3], [1, 3]) → True

2. Given an array of ints length 3, return the sum of all the elements.

3. Given an array of ints length 3, return a new array with the elements in reverse order, so {1, 2, 3} becomes {3, 2, 1}.

reverse3([1, 2, 3]) → [3, 2, 1]
reverse3([5, 11, 9]) → [9, 11, 5]
reverse3([7, 0, 0]) → [0, 0, 7]

4. Given an array of ints length 3, figure out which is larger, the first or last element in the array, and set all the other elements to be that value. Return the changed array.

max_end3([1, 2, 3]) → [3, 3, 3]
max_end3([11, 5, 9]) → [11, 11, 11]
max_end3([2, 11, 3]) → [3, 3, 3]

5. Given an array of ints, return the sum of the first 2 elements in the array. If the array length is less than 2, just sum up the elements that exist, returning 0 if the array is length 0.

sum2([1, 2, 3]) → 3
sum2([1, 1]) → 2
sum2([1, 1, 1, 1]) → 2

6. Given an array of ints length 3, return an array with the elements "rotated left" so {1, 2, 3} yields {2, 3, 1}.

rotate_left3([1, 2, 3]) → [2, 3, 1]
rotate_left3([5, 11, 9]) → [11, 9, 5]
rotate_left3([7, 0, 0]) → [0, 0, 7]

7. Given an int array length 2, return True if it contains a 2 or a 3.

has23([2, 5]) → True
has23([4, 3]) → True
has23([4, 5]) → False

8. Given an array of ints, return a new array length 2 containing the first and last elements from the original array. The original array will be length 1 or more.

make_ends([1, 2, 3]) → [1, 3]
make_ends([1, 2, 3, 4]) → [1, 4]
make_ends([7, 4, 6, 2]) → [7, 2]

9. Return the sum of the numbers in the array, returning 0 for an empty array. Except the number 13 is very unlucky, so it does not count and numbers that come immediately after a 13 also do not count

sum13([1, 2, 2, 1]) → 6
sum13([1, 1]) → 2
sum13([1, 2, 2, 1, 13]) → 6

10. Given an array length 1 or more of ints, return the difference between the largest and smallest values in the array. Note: the built-in min(v1, v2) and max(v1, v2) functions return the smaller or larger of two values.

big_diff([10, 3, 5, 6]) → 7
big_diff([7, 2, 10, 9]) → 8
big_diff([2, 10, 7, 2]) → 8

11. Return the number of even ints in the given array. Note: the % "mod" operator computes the remainder, e.g. 5 % 2 is 1.

count_evens([2, 1, 2, 3, 4]) → 3
count_evens([2, 2, 0]) → 3
count_evens([1, 3, 5]) → 0

12. Given 2 int arrays, a and b, each length 3, return a new array length 2 containing their middle elements.

middle_way([1, 2, 3], [4, 5, 6]) → [2, 5]
middle_way([7, 7, 7], [3, 8, 0]) → [7, 8]
middle_way([5, 2, 9], [1, 4, 5]) → [2, 4]

13. Given an array of ints, return True if the array is length 1 or more, and the first element and the last element are equal.

same_first_last([1, 2, 3]) → False
same_first_last([1, 2, 3, 1]) → True
same_first_last([1, 2, 1]) → True

14. Return True if the string "cat" and "dog" appear the same number of times in the given string.

cat_dog('catdog') → True
cat_dog('catcat') → False
cat_dog('1cat1cadodog') → True

15. Return the number of times that the string "code" appears anywhere in the given string, except we'll accept any letter for the 'd', so "cope" and "cooe" count.

count_code('aaacodebbb') → 1
count_code('codexxcode') → 2
count_code('cozexxcope') → 2

16. Given two strings, return True if either of the strings appears at the very end of the other string, ignoring upper/lower case differences (in other words, the computation should not be "case sensitive"). Note: s.lower() returns the lowercase version of a string.

end_other('Hiabc', 'abc') → True
end_other('AbC', 'HiaBc') → True
end_other('abc', 'abXabc') → True

17. Return the number of times that the string "hi" appears anywhere in the given string

count_hi('abc hi ho') → 1
count_hi('ABChi hi') → 2
count_hi('hihi') → 2

18. Given a string, return a string where for every char in the original, there are two chars

double_char('The') → 'TThhee'
double_char('AAbb') → 'AAAAbbbb'
double_char('Hi-There') → 'HHii--TThheerree'

19. Given an array of ints, return True if 6 appears as either the first or last element in the array. The array will be length 1 or more.

first_last6([1, 2, 6]) → True
first_last6([6, 1, 2, 3]) → True
first_last6([13, 6, 1, 2, 3]) → False

1. What is the difference between list and tuples?
2. What is the difference between deep and shallow copy?
3. How can the ternary operators be used in python?
4. How is memory managed in Python?
5. Explain Inheritance in Python with an example.
6. Explain what Flask is and its benefits?
7. What is the usage of help() and dir() function in Python?

8. What is dictionary in Python?

9. What is monkey patching in Python?

10. What does this mean: *args, **kwargs? And why would we use it?

11. Write a one-liner that will count the number of capital letters in a file. Your code should work even if the file is too big to fit in memory.

12. What are negative indexes and why are they used?

13. How can you randomize the items of a list in place in Python?

14. What is the process of compilation and linking in python?

15. How can you generate random numbers in Python?

16. What is the difference between range & xrange?

17. What is pickling and unpickling?

Functions in Python

1) What is function
2) Advantage of function
3) How functions make development faster
4) How to define and call a function
5) What's the difference between define and calling a function
6) Can we call a function inside of another function
7) How many times we can call a function

8) What is local variable

9) What is scope of local variable

10) Can we declare local variables with same name in different functions

11) How many ways we can define functions

12) What is arguments in functions

13) What is parameter

14) Can we pass multiple arguments

15) The modifications occur in parameters does it reflect to arguments

16) What is named or keyword arguments

17) Can we mix non-keyword arguments with keyword arguments

18) What is a global variable

19) What is global constant

20) What is the purpose of return statement in a function

21) What is Boolean function

22) Can we return multiple values

23) What is lambda function

24) How to create Anonymous functions

25) How many arguments can we pass to lambda functions

26) Can we declare lambda functions with default arguments

27) What's the use of map in lambda

28) What's the use of filter function in lambda

29) What is use of reduce function

30) What is recursive function

Module

1) What is module
2) Can we call global variables or methods of one program to another program
3) How to access modules
4) How to call a function in side of another module
5) Can i change module names in our program
6) Can we import only required function in another module
7) Can we import two or more functions from module and how
8) I have a module, which contains variable ph, and two functions email( ) , ph( ). Now i want to import ph variable. How to do it.
9) How to know what are the functions and variables available in a module
10) Can we import more than one module at a time
11) Can we import required functions from more than one module

Packages:

1) What is package
2) To create a package, which file is compulsory
3) we need to write any logic for _ _ init_ _ .py file
4) How to access modules inside of a packages
5) Can we create package in side another package
6) we need to create _ _ init_ _.py file inside sub packages
7) How to call sub-package related modules

OOPS

1. What is OOPS
2. How many types of principles in OOPS and what are they.
3. What is OOPL
4. What is the difference between OOPS and OOPL
5. Is Python supports OOPS
6. Is OOPS Principles fallows any order and what is that.
7. What is Abstraction
8. What is Encapsulation
9. How to achieve encapsulation in Python?
10. What is Inheritance
11. What is Polymorphism

12. What is class?
13. How to define a class?
14. What is Object?
15. What is the use of the Object?
16. How to create Object?
17. How to destroy or delete an Object from memory?
18. Where Object gets memory?
19. Where class gets memory?
20. What is instance variable?
21. What is static variable?
22. Where Instance variables are created?
23. Where static variables are created?
24. What is the difference between instance and static variables?
25. When static variables get memory?
26. When instance variables get memory?
27. Where static variables get memory?
28. Where instance variables get memory?
29. What are the ways of accessing static variables?
30. What are the ways of accessing instance variables?
31. What happens if I use all variables as instance in the class?

32. What happens if I use all variables as static in the class?
33. What is a method?
34. How to define a method in python?
35. How to use/call a method?
36. What is the use of method parameters?
37. What are the types of methods in python
38. When should I define an Instance method?
39. When should I define a static method?
40. What are the ways of accessing an instance method?
41. What are the ways of accessing a static method?
42. What are the variables can instance method access
43. What are the variables can static method access
44. Can we create a variable inside the method?
45. Can we print a variable without setting a value? If not why?
46. Who will provide the initial/default value to local variable?
47. Can we initialize the instance or static variables through method?
48. Can we assign the instance or static variables through method?
49. Who will provide the initial/default value to instance variable?

50. Can a method call another method?
51. Can I call a static method from instance method?
52. Can I call an instance method from instance method?
53. Can I call an instance method from static method?
54. Can I call a static method from static method?
55. What is constructor?
56. What are the rules in Defining a constructor?
57. Where/When Constructor is used?
58. Can I write a class without defining at least constructor?
59. Can I create an object if a class is having zero constructors?
60. What are the types of constructor?
61. What is the parameterized constructor?
62. When should we use parameterized constructor?
63. How to call a constructor?
64. Can we use a constructor to initialize static variables?
65. Can we call a constructor from a method(static/instance)?
66. Can we call a method from constructor?
67. Can constructors access static variables?
68. Can constructors access instance variables?
69. Can I declare constructor as static
70. Can I write a method with class name?

71. What happens if I define a method with class name?
72. What is Inheritance?
73. What is the use of Inheritance?
74. Can static members participate in inheritance?
75. Can instance members participate in inheritance?
76. Can we inherit a constructor?
77. What are the types of Inheritance?
78. What happens if we create an Object to sub class/any class?
79. How to initialize the super class instance variables?
80. How to call a super class constructor?
81. Who will call upper class constructor by default?
82. How to call one constructor from another constructor of same class?
83. What is the use of calling one constructor from another of same class?
84. What is Overloading?
85. What is method overloading?
86. What is constructor overloading?
87. What is the use of overloading?
88. Can we overload a constructor in different class?
89. Can we overload an instance method in different class?

90. Can we overload a constructor in same class?
91. Can we overload an instance method in same class?
92. Can we overload variables?
93. Can we overload static method?
94. Can we overload instance method?
95. What is overriding?
96. What is method overriding?
97. What is constructor overriding?
98. Is overriding done without inheritance?
99. What is the use of overriding?
100. Dose overriding depends on parameter?
101. Dose overriding depends on method name?
102. Can we overriding methods in same class?
103. Can we override methods in different class?
104. Can we override variables?
105. How to stop method overriding?