Q1

1. it is a floor division operator...

No

floor division removes and decimels, and shows an intger

2. The output of run the following call ```print((1.1 + 2.2) == 3.3) ``` is True

No

programming langues are not that accurate with numbers, there should be an extra number at the end, though there is a workaround using libraries that are specialized with number and float precision

3. Thw following statement

```
sum = 1 + 2 + 3 + 4 + \
      5 + 6
```

is equivalent to ? (Yes)

```
sum = 1 + 2 + 3 + 4 + 5 + 6
```

and is equivalent to ? (No, Python is sensitive to indentations)

```
sum = 1 + 2 + 3 + 4 +
      5 + 6
```

4. The “and” in python does an OR logical operation?

No

similar to && in other languages

5. You can use ```eval``` function to...

```
No
 the eval function evaluate a string as a Python code, and return the result as a string
```

6. The ```in``` operator is used to check for a value exists...

```
Yes
```

7. The operator preccedence in Python is different than other lanagues like C++

```
Yes
```

8. The ```is``` operator in Python checks if two object has the same id

```
Yes
```

9. The ```not``` operator in Python performs a logical negation

```
Yes
```

10. To generate a number you need to do this

```
import random
random.random()
```
This return a floating point number in the range of [0, 1]

```
Yes
```

11. In order to convert a number into a string use the ```str()`` funciton...

```
Yes
```

12. Importing everything with the asterisk(*) is good, but can lead to duplicate definitions

```
Yes
```

13. In Python the ```from ... import ...``` statement is used to import specific parts...

```
Yes
```

14. In Python a module is the way to structure a program...

```
Yes
```

Q2

1. What is the value of x if ```x = int(44.55 + 2/2)```

```
44
```

2. What is the value of the following expression ```2 + 4.00, 2 ** 4.0```

```
6.0, 16.0
```

3. Which of the following is truncation operator

    1. /
    2. %
    3. //

    ```
    3
    ```

4. What are the values of the following expressions

    ````
    x = 2 ** ( 3 ** 2 )
    y = ( 2 ** 3 ) ** 2
    z = 2 ** 3 ** 2
    ````

    1. 512 64 512
    2. 64 512 512
    3. 512 51 64

```
1
```

5. What is the value of the following expression ```8 / 4 / 2, 8 / (4 / 2)```

```
(1.0, 4.0)
```

6. What is the value of the folowing expression ```float(22 // 3 + 3 / 3)```

```
8.0
```

7. Which of the following operators has its associativity from right to left?

1. +
2. //
3. %
4. **

```
4
```

8. Which operator has higher precedence in the following list

1. %
2. &
3. **
4. >

```
1
```

9. What is the output of the following code?

```
var = “James” * 2 * 3
print(var)
```

```
JamesJamesJamesJamesJamesJames
```

10. The value of expression ```4 + 3 % 5```

```
7
```

11. Display the sum of 5 + 10 using two variables x and y

```
x = 5
y = 10
print( x ... y )
```

```
+
```

12. multiply 10 with 5 and print the result

```
print( 10 ... 5 )
```

```
*
```

13. Divide 10 by 2 and print the result

```
print( 10 ... 2 )
```

```
/
```

14. Use the correct membership operator to check if “apple” is present in the fruits

```
fruits = [“apple”, “banana”]
if “apple” ... in fruits:
    print(“yes, apple is a fruits”)  # no way
```

```
in
```

15. The output of the following program is?

```
x = 5
y = 0
result = x / y
```

```
print(result)
```

```
Error: division by zero
```

16. Use the correct comparison operator to check if 5 is **not equal** to 10

```
if 5 ... 10:
    print("5 and 10 is not equal") # I think it is
```

```
!=, is not

though is not will get you a SyntaxWarrning, because it
weird to use it with ints literals
```

17. Use the correct logical operator to check if at least one of two statement is True

```
if 5 == 10 ... 4 == 4:
    print("At least one of the statement is true")
```

```
or
```

18. What will be the output of the following code?

```
x = 10
y = 5
print(x > y)
```

```
True
```

19. Which of the following is used to check if two objects refer to the same memoery location

1. is
2. ==
3. equals()
4. same()

```
is
```

20. What is the output of the following code?

```
valueOne = 5 ** 2
valueTwo = 5 ** 3

print(valueOne, end='     ')
print(valueTwo)
```

```
25 125
```

21. Evalue the expression given bellow if A = 16 and B = 15

```
print(A % B // A)
```

```
0
```

22. What is the output of the following code?

```
x = 36 / 4 * ( 3 + 2 ) * 4 + 2
print(x)
```

```
182.0
```

23. What is the output of the following code?

```
print( 2 * 3 ** 3 * 4 )
```

```
864
```

24. What is the output of the following code?

```
x = 6
y = 2
```

```
print( x ** y, end = ‘       ’)
print( x // y )
```

```
36.3
```

25. What is the output of the following code?

```
print( 2 % 6 )
```

```
2
```

26. What is the average value of the code that is executed below

```
grade1 = 80
grade2 = 90

average = ( grade1 + grade2 ) / 2
```

```
85.0
```

27. In Python we do not specify types...

```
I have no idea what you want??
```

28. What is the output of

```
print( 2 ** 3 ** 2 )
```

```
512
```

29. Which of the following operators has the highest precedence?

1. not
2. &
3. *
4. +

```
3
```

30. What is the output of the following assignment operator?

```
y = 10
x = y += 2

print(x)
```

```
SyntaxError
```

31. What is the output of the following code?

```
x = 100
y = 50

print(x and y)
```

```
50
```

32. What is the value of the following Python expression

```
print(36 / 4)
```

```
9.0
```

33. What is the output of the expression

```
print(-18 // 4)
```

```
-5
```

34. What is the output of the following code?

```
print(10 – 4 * 2 )
```

```
2
```

35. What is the output of the following math function?

```
import math
print(math.ceil(252.4))
print(math.floor(252.4))
```

```
253
252
```

36. What is the correct synatx of printing all variables and function of a module

```
import mymodule

print(....)
```

```
dir(mymodule)
```

37. What is the correct syntax of importing only the person1 dictionary of the mymodule module?

```
... mymodule .... person1
```

```
from mymodule import person1
```

38. Which of the following properly expresses the precedence of the operators....

```
5 * > 10 and 4 + 6 == 11
```

```
((5 * 3) > 10) and ((4 + 6) == 11)
```

39. The output of the following python statement

```
print(chr(ord('A')))
```

```
A
```

40. What will be displated by

```
print(ord('b') - ord('a'))
```

```
1
```

41. What is the output of

```
print(abs(-45.300))
```

```
45.3
```

42. What is the output of the following isinstance() function

```
from numbers import Number
from decimal import Decimal
from fractions import Fraction

print(isinstance(2.0, Number))
```

print(isinstance(Decimal('2.0'), Number)
print(isinstance(Fraction(2, 1), Number)
print(isistance("2", Number)
```

```
True False True True
```

43. What is the correct syntax to import a module named "mymodule"?

```
... mymodule
```

```
import
```

44. If you want to refer to a module by another name ...

```
import mymodule ... mx
```

```
as
```

45. In Python, to import a specific part...

```
from math import pi
```

Q3
Evaluate the following expressions in Python

1. 25 / 3

```
8.333
```
```

2. 20 – 12 / 4 * 24

```
14.0
```

3. 32 % 7

```
4
```

4. 3 – 5 % 7

```
-2
```

5. 18.0 / 4

```
4.5
```

6. 28 – 5 / 2.0

```
25.5
```

7. 17 + 5 % 2 – 3

```
15
```

8. 15.0 + 3.0 * 2.0 / 5.0

```
16.2
```

Q4

Write equivalent compund statements of the following simple statements if possible

1. x = 2 * x

```
x *= 2
```

2. x = x + y – 2

```
x += y - 2
```

3. sum = sum + num

```
sum += num
```

4. z = z * x + 2 * z

5. y = y (x + 5)

```
y /= x + 5
```

Q5

Write the following compund statements as equivalent simple statements

1. x += 5 – z

```
x = x + 5 – z
```

2. y *= 2 * x + 5 - z

```
y = y * 2 * x + 5 – z
```

3. w += 2 * z + 4

```
w = w + 2 * z + 4
```

4. x -= z + y – t

```
x = x – z + y – t
```

5. sum += num

```
sum = sum + num
```

Q6

Which of the following assignments are valid or not and why?

1. num 1 = 35

```
Valid
```

2. num 2 += 4

```
Valid
```

3. newNum = num1 – num2

```
Valid
```

4. num1 = 5; num2 = 2 + num1; num1 = num2 / 3

```
Valid
```

5. num1 * num2 = newNum

```
not valid
     it's reversed
```

6. x = 12 * num1 – 15.3

```
Valid
```

7. num1 * 2 = newNum + num2

```
not valid
     you can't assign a result
```

8. x / y = x * y

```
not valid
     you can't assign a result
```

9. num2 = num 1 % 2.0

```
Valid
```

10. newNum = static_cast<int> % 5

```
not valid
     who let c++ in?
```

11. x = x + y – 5

```
Valid
```

12. newNum = num1 + int(4.6/2)

```
Valid
```

Q7
Identify errors in the following Python code snapshot and then correct them

1.

```
x = 10
y = 5
result = x – y
print(“The result is: “ + result)
```

wrong

wrap result with str

```
str(result)
```

2.

```
x = 5
y = “2”
result = x + y
print(result)
```

wrong

wrap y with int

```
result = x + int(y)
```

3.

```
num = 10
print(“The sqaure root of num is: “ + math.sqrt(num))
```

wrong

wrap the sqrt with str

```
print(“The sqaure root of num is: “ +
str(math.sqrt(num)))
```

Q8
What is the output of the following program

1.

```
....
```

```
5.0
3.1415…
144.59155
1.04..
0.90..
0.877..
0.234..
24
```

2.

```
....
```

```
0 → 5 random int
0 → 1 random float
0 → 100 random float
random item from the list
```

3.

```
.....
```

```
--------------------------
     Format the output of ...
--------------------------

Current date and time: 2025-04-14 06:00:46.083582
Formatted date and time: 2025-04-14 06:01:48


--------------------------
   Display the time in 12h..
```

```
-------------------------
Formatted date and time: 2025-04-14 13:01:48


-------------------------
 Customzide the date and time…
-------------------------

Formatted date and time: Monday, April 14 2025 06:04:48 AM

Formatted date and time: Mon, Apr 14 25 06:05:31 AM
```

4.

```
‴

    ….
‴

    d1 =  14/04/2025
    d2 =  April 14 2025
    d3 =  04/14/25
    d4 =  Apr-14-2025
```

Q9

Do these programming excerises with Python langauage

1. Write a program to perform different arithmetic operations on numbers in Python

```
 ```
def arithmetic_ops(a, b):
    print("Addition:", a + b)
    print("Subtraction:", a - b)
    print("Multiplication:", a * b)
    if b != 0:
        print("Division:", a / b)
        print("Modulus:", a % b)
        print("Floor Division:", a // b)
    else:
        print("Division: Error - Division by zero")
    print("Exponentiation:", a ** b)
 ```
```

2. Write a program to crate concatenate and print a string and accessing sub-string from given string

```
str1 = "Hello, "
str2 = "World!"
concat_str = str1 + str2
print("Concatenated String:", concat_str)


substring = concat_str[7:12]
print("Substring (extracted 'World'):", substring)
```

3. try a print the day, month, year in the form “Today is 2/2/2016”

```
I did not understand
```

4. Make a python program to find out what versionb of Python you are using

```
import sys
print("Python version:", sys.version)
```

5. write a program to get three value from the user and compute the average

```
num1 = float(input("Enter first number: "))
num2 = float(input("Enter second number: "))
num3 = float(input("Enter third number: "))

average = (num1 + num2 + num3) / 3
print("The average is:", average)
```

6. write a program that takes 2 numbers as command line aarguments and make a simple calculator

```
import sys

if len(sys.argv) != 3:
    print("Usage: python calculator.py <num1> <num2>")
```

```
    sys.exit(1)

try:
    num1 = float(sys.argv[1])
    num2 = float(sys.argv[2])
except ValueError:
    print("Please provide valid numbers.")
    sys.exit(1)

print("Addition:", num1 + num2)
print("Subtraction:", num1 - num2)
print("Multiplication:", num1 * num2)
if num2 != 0:
    print("Division:", num1 / num2)
else:
    print("Division: Error - Division by zero")
```

7. write a python program to define a module and import specifc function in thtat module to another program

module.py
```
def hey(name):
    return f"Hello, {name}!"
```

main.py
```
from module import hey

name = input("Enter your name: ")
print(greeting(name))
```

8. import the math module and call sin function

```
import math

angle_radians = math.pi / 2
print("The sine of 90 degrees is:", math.sin(angle_radians))
```

9. write a python script to print the currect date in the following format “Sun May 29 02:26:23 IST 2017”

```
import datetime
import time

tz = datetime.timezone(datetime.timedelta(hours=5, minutes=30))
now = datetime.datetime.now(tz)
formatted_date = now.strftime("%a %b %d %H:%M:%S %Z %Y")
print("Current date and time with IST timezone:", formatted_date)
```

10. write a python program to add some days to your present date and print the dater added

```
import datetime

days_to_add = int(input("Enter number of days to add: "))
today = datetime.date.today()
future_date = today + datetime.timedelta(days=days_to_add)
print("Date after adding", days_to_add, "days:", future_date)
```

11. Write programs to read in a value of x then calculate the following expressions

1. $x - 2 / x^2 + 1$
2. 2^{x+1}
3. x^{-2x}

```
    x = float(input("Enter a value for x: "))

    expr1 = (x - 2) / (x**2 + 1)

    expr2 = (2 * x) + 1

    expr3 = x - (2 ** x)
```

```
   print(f"\nResults:")
   print(f"1. (x - 2) / (x^2 + 1)  = {expr1}")
   print(f"2. (2 * x) + 1          = {expr2}")
   print(f"3. x - 2^x              = {expr3}")
```

12. Create a program which converts temperature from Celcius to Fehernhite

```
c = float(input("Celsius: "))
f = (c * 9 / 5) + 32
print("Fahrenheit:", f)
```

13. Write a program to compute distance between two points taking input from the user. The Pythagorean theorem is the basis for computing distance between two points. Let (x1, y1) and (x2, y2) be th co-ordinates of points on xy-plane. From Pythagorean theorem, the distance between two points is calculated using the formulae: math.sqrt((x2 – x1) ** 2 + (y2 – y1) ** 2)

```
import math

def calculate_distance(x1, y1, x2, y2):
    return math.sqrt((x2 - x1)**2 + (y2 - y1)**2)

x1 = float(input("Enter x1: "))
y1 = float(input("Enter y1: "))
x2 = float(input("Enter x2: "))
y2 = float(input("Enter y2: "))

distance = calculate_distance(x1, y1, x2, y2)
print(f"The distance between ({x1}, {y1}) and ({x2}, {y2}) is:
{distance:.4f}")
```

14. Write a program to convert U.S dollars to indian rupees

```
us_bucks = float(input("Enter in U.S dollars: "))
```

```
print(f"{us_bucks} in rupees is: {us_bucks * 74.5}")
```

15. Write a program to convert bits to Megabytes, Gigabytes, and Terabytes

```
bits = 10 * 100

bytes_ = bits / 8

megabytes = bytes_ / (1024 ** 2)
gigabytes = bytes_ / (1024 ** 3)
terabytes = bytes_ / (1024 ** 4)
```

16. Write a progam to find the square root of a number

```
import math

num = 5

print(f"sqaure root: {math.sqrt(num)}")
```

17. Write a program to calculate area and primeter of the sqaure

```
side = float(input("Enter the length of the side of the square:
"))

area = side * side
perimeter = 4 * side

print("Area of the square:", area)
print("Perimeter of the square:", perimeter)q
```

18. Write a program to swap the value of two variables

```
```

```
x = 1
y = 2

x = y + x
y = x - y
x = x - y
```

19. Write a Python program to convert kilometer to miles

```
kilometers = float(input("Enter distance in kilometers: "))

miles = kilometers * 0.621371

print("Distance in miles:", miles)
```