**Python Programming Unit 2 Notes and MCQs**

**1. Python Operators**

**Operators in Python**

* Operators perform operations on variables and values.
* Types of operators:
  + Arithmetic
  + Assignment
  + Comparison
  + Logical
  + Bitwise
  + Identity
  + Membership

**Arithmetic Operators**

* Used for mathematical calculations.
* Example:
  + + (Addition) → 10 + 3 = 13
  + - (Subtraction) → 10 - 3 = 7
  + \* (Multiplication) → 10 \* 3 = 30
  + / (Division) → 10 / 3 = 3.33
  + // (Floor Division) → 10 // 3 = 3
  + % (Modulus) → 10 % 3 = 1
  + \*\* (Exponentiation) → 10 \*\* 3 = 1000

**Assignment Operators**

* Used to assign values to variables.
* Example:
  + = (Assign) → C = A + B
  + += (Add and assign) → A += B
  + -= (Subtract and assign) → A -= B

**Comparison Operators**

* Used for comparing values.
* Example:
  + > (Greater than) → A > B
  + < (Less than) → A < B
  + == (Equal to) → A == B

**Logical Operators**

* Used for logical conditions.
* Example:
  + and → True if both conditions are True
  + or → True if one condition is True
  + not → Reverses the result

**Bitwise Operators**

* Operate on binary numbers.
* Example:
  + & (AND)
  + | (OR)
  + ~ (NOT)

**Identity and Membership Operators**

* is and is not for object identity.
* in and not in for checking membership.

**2. Conditional and Looping Statements**

**If Statements**

* if → Executes if the condition is True.
* if-else → Executes one block if True, another if False.
* elif → Multiple conditions.

**Loops in Python**

* **For loop** → Iterates over a sequence.
* **While loop** → Runs as long as the condition is True.
* **Break, Continue, Pass**:
  + break → Stops loop execution.
  + continue → Skips current iteration.
  + pass → Placeholder for future code.

**3. Functions in Python**

* Defined using def keyword.
* Types of functions:
  + Built-in functions (e.g., print(), len())
  + User-defined functions
* Arguments:
  + Positional, Keyword, Default, Variable-length

**4. MCQs**

**Q11. Which keyword is used to define a function in Python?**

* a) func
* b) def
* c) define
* d) function

**Answer: b) def**

**Q12. What is the output of bool(0)?**

* a) True
* b) False
* c) None
* d) Error

**Answer: b) False**

**Q13. What will print("Hello" \* 3) output?**

* a) HelloHelloHello
* b) Error
* c) Hello 3 times
* d) None

**Answer: a) HelloHelloHello**

**Q14. What is the correct way to open a file in Python for reading?**

* a) open('file.txt', 'r')
* b) open('file.txt', 'w')
* c) open('file.txt', 'rb')
* d) open('file.txt', 'wb')

**Answer: a) open('file.txt', 'r')**

**Q15. What does continue do in a loop?**

* a) Stops the loop execution
* b) Skips the current iteration
* c) Exits the program
* d) Jumps to another function

**Answer: b) Skips the current iteration**

**Q16. What will be the output of print(type([]))?**

* a) list
* b) tuple
* c) dict
* d) set

**Answer: a) list**

**Q17. What is the result of 3 \*\* 2?**

* a) 6
* b) 9
* c) 3
* d) 8

**Answer: b) 9**

**Q18. Which of the following is NOT a valid Python data type?**

* a) List
* b) Dictionary
* c) Enumeration
* d) Tuple

**Answer: c) Enumeration**

**Q19. How do you start a single-line comment in Python?**

* a) //
* b)
* c) #
* d) /\* \*/

**Answer: c) #**

**Q20. What function is used to take input from the user in Python?**

* a) read()
* b) input()
* c) scan()
* d) enter()

**Answer: b) input()**