Java 1.8 or above (preferred id java 17) and windows

Programming Language (What, Why)

1. JAVA
2. PHP
3. C
4. C++
5. .Net

JAVA

1. What is java
2. Java SDK (>= 1.17) (jdk+jre)
   a. Jdk - > javac Library.java
   b. Jre -> java Library
3. Compile -> Converts to Machine Lang (binary file/byte code) javac
4. Execute -> Exec binary code => java

OOPS Principles

1. OOPS Features
   a. Encapsulation
      i. Binding all the members into a single unit called as class
         - Ex: Capsule
         - In Java: - Class
   b. Inheritance
      i. Inherit the property from parent to child
      ii. Code re-usability
   c. Polymorphism
      i. Compile Time: Overloading
      ii. Run time: Over-ridding
   d. Abstraction
      i. Hiding the complexity of program
         - Ex: Mobile app

1. Class
    a. A blueprint, Skeleton that wraps it’s data member into a single unit (encapsulation)
2. Machine independent
    a. Byte code(.class file)
3. Platform independent (OS independent)
    a. JVM – Java virtual machine

Assignment

- Write a program to take three input during runtime and print all in same line and each input in separate line.
- i/p -> Welcome to Class
- o/p ->
    - WelcometoClass
    - Welcome
    - To
    - Class

4. Variables
    a. Instance Variables (global)
        i. Can be accessed by an Object
    b. Class Variables (static)(global)
        i. Can be accessed by an object or Class name
    c. Local Variables
        i. Always declared inside a method or block of code
        ii. Have only default access modifier or final as non-access modifier.
        iii. Default values are not assigned.
5. Data Types (primitives)
    a. byte - 8bits
        i. - default 0
    b. short – 16
        i. - default 0
    c. int - 32
        i. - default 0
    d. long -64
        i. - default 0
    e. float -32

- i. - default 0.0
- f. double -64
  - i. - default 0.0
- g. char 16
  - i. - default ‘’
- h. Boolean
  - i. - default false

6. Type casting
   - a. Implicit - automatically done
   - b. Explicit – have to do the casting

Program:

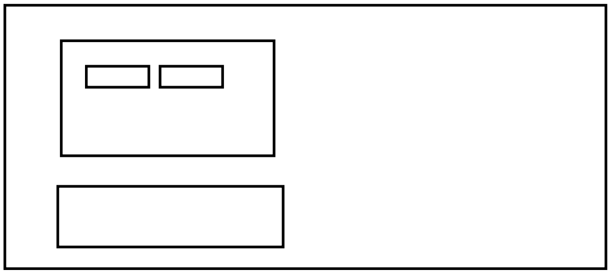
- Adding two numbers with different numeric and decimal data types

7. Operators
   - a. Types of Operator
     - i. Arithmetic Operators
       - (+,-,*,/,%,++,-- , %= , += , -= , *= ,/=)
         - a. ++a => pre-Increment
         - b. A++ => post-Increment
         - c. Write prog for all arithmetic operator
     - ii. Relational Operators
       - (== , != , >,<,>=,<=)
     - iii. Logical Operators
       - (& , | , ||, &&)
       - Practice with **“|” , “||”**
     - iv. Conditional Operators
       - Ternary operator
         - a. Condition? true: false
         - b. Exp1?exp2:exp3
           - i. If exp1 returns true then exp2 get executed else exp3.
       - Assignment operator (=)
   - b. Operator precedence
     - i. a++ , a--
     - ii. ++a , --a ,!

iii. * / % (in revision class)
iv. && (Logical AND) , || (Logical OR)
v. ? : (Ternary operator)
vi. Write a program to check this. (++ ,--) (increment &decrement)

8. Arrays: Fixed length for similar kind of objects/types.
    a. Write prog for int type and use operators with 1-D array data.
    b. Write prog for int type and use operators with 2-D array data.

❖ Define
❖ Initialize
    ○ Index starts with 0
    ○ Counting starts with 1.
    ○ Length is used to retrieve the size of an array
    ○ Size= length
❖ 2-D Array
    ○



9.
10.
11.
12.
13.
14. Control Statement (always last statement in that block)
    a. break;
    b. continue;
    c. return
15. Loops (iterations)
    a. If/else
    b. If else if .. ladder
    c. Nested if else.
    d. Switch case
        i. Check for expression type in switch.
    e. While while(condition) { // block of code }

f. Do while do { // block of code } while (condition)
g. For
h. Foreach