**Q1. what is statically types and dynamically types programming language**

**Answer:** A statically typed programming language is a language in which the type of a value is known at compile time, and does not change at runtime. This means that type errors can be caught and reported before the program is executed. Examples of statically typed languages include C, C++, and Java.

A dynamically typed programming language, on the other hand, is a language in which the type of a value is determined at runtime, and can change during the execution of a program. Examples of dynamically typed languages include Python, Ruby, and JavaScript.

In dynamically typed languages, type errors are usually reported at runtime, which can make debugging more challenging compared to statically typed languages. However, dynamically typed languages often provide greater flexibility and more concise code, and are often favoured by developers for their ease of use and rapid development times.

**Q2. what is variable in java**

**Answer:** A variable in Java is a named location in memory that is used to store a value. In Java, all variables must have a specific data type, which determines the size and layout of the variable's memory, as well as the set of operations that can be performed on it.

There are several types of variables in Java, including:

Primitive variables: These are the basic data types in Java, such as int, float, and Boolean. Primitive variables store the actual values, not references to objects.

Reference variables: These are variables that store references to objects, rather than the objects themselves. In Java, all objects are accessed via reference variables.

Instance variables: These are variables that are declared in a class, outside of any method. They belong to the object and have a default value of null, 0 or false depending on the data type.

Static variables: These are variables that are shared by all instances of a class. They belong to the class and have only one copy for all instances.

Local variables: These are variables that are declared inside a method or a block of code. They only have scope within the method or block and are not accessible from outside.

Variables must be declared before they can be used in a Java program, and must be assigned a value before they can be used.

**Q3. how to assign value to a variable?**

**Answer:** In Java, you can assign a value to a variable by using the assignment operator (=). The syntax for assigning a value to a variable is as follows:

variable\_name = value;

**Q4. what are primitive data types in java?**

**Answer:** In Java, primitive data types are the most basic data types that can be used to store individual values in a program. There are eight primitive data types in Java:

byte: This is an 8-bit signed integer type that can store values from -128 to 127.

short: This is a 16-bit signed integer type that can store values from -32,768 to 32,767.

int: This is a 32-bit signed integer type that can store values from

-2,147,483,648 to 2,147,483,647.

long: This is a 64-bit signed integer type that can store values from

-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.

float: This is a 32-bit floating-point type that can store decimal values with 7 decimal digits of precision.

double: This is a 64-bit floating-point type that can store decimal values with 15 decimal digits of precision.

char: This is a 16-bit Unicode character type that can store a single character.

boolean: This is a type that can store the values true or false.

These data types are stored directly in memory and are the most efficient types to use when storing simple values in a program. The size and range of values that each primitive data type can store is defined by the Java language specification.

**Q5. what are identifiers in java?**

**Answer:** All Java **variables** must be **identified** with **unique names**.

These unique names are called **identifiers**.

Identifiers can be short names (like x and y) or more descriptive names (age, sum, total Volume).

The general rules for naming variables are:

* Names can contain letters, digits, underscores, and dollar signs
* Names must begin with a letter
* Names should start with a lowercase letter and it cannot contain whitespace
* Names can also begin with $ and \_ (but we will not use it in this tutorial)
* Names are case sensitive ("myVar" and "myvar" are different variables)
* Reserved words (like Java keywords, such as int or boolean ) cannot be used as names

**Q6. List the operators in java?**

**Answer:** In Java, there are several types of operators that perform different operations on variables and values. The following is a list of operators in Java:

Arithmetic Operators: +, -, \*, /, % (modulo), ++ (increment), -- (decrement)

Comparison Operators: == (equal to),!= (not equal to), > (greater than), < (less than), >= (greater than or equal to), <= (less than or equal to)

Logical Operators: && (and), || (or), ! (not)

Bitwise Operators: & (and), | (or), ^ (exclusive or), ~ (complement), << (left shift), >> (right shift), >>> (unsigned right shift)

Assignment Operators: =, +=, -=, \*=, /=, %=, &=, |=, ^=, <<=, >>=, >>>=

Conditional Operators:? : (ternary operator)

Type Comparison Operator: instance of

Each of these operators performs a different operation, and the result of the operation can be stored in a variable, used in an expression, or used as part of a larger operation. It's important to understand the behaviour and precedence of each operator to write correct and efficient code in Java.

**Q7. Explain about increment and decrement operators and give an example?**

**Answer:** Increment and decrement operators are unary operators. We can only apply these operators on a single operand, hence these operators are called as unary operators.

the increment operator is an operator which is used to increase the value of a variable by 1, on which it is applied.

Again, these increment operators are two types:

* Pre increment (++x)
* Post increment (x++)

The Decrement operator is an operator which is used to decrease the value of the variable by 1, on which it is applied.

Like increment operators, decrement operators are also 2 types,

* Pre decrement (- -x)
* Post decrement (x- -)