RGPV PYQ

1. Data Type:

It is a way of organizing data in such a manner that you let the computer know how the programmer intends to use it.

It shows what kind of values can be stored in the variable and what actions can be carried out on those values.

Some examples of simple data types are integers, floating-point numbers, characters, strings and more complex types like array and struct.

In programming, data types serve as constraints on data thus making sure that you do meaningful operations.

Data types example in C language:

C

```c
int age = 25;          // Integer data type
float pi = 3.14;       // Floating-point data type
char grade = 'A';      // Character data type
```

2. Tokens:

Tokens are the smallest meaningful units to the compiler or interpreter in programming.

These include identifiers (i.e., variable and function names), literals, operators, punctuation symbols, and keywords—these are what make up a program in its individual parts.

In lexical analysis phase of compilation, source codes are split into tokens which help compilers view how they are arranged in the program.

Tokens example in Python language:

Python

```python
# Tokens in Python
x = 10       # 'x', '=', '10' are tokens
if x > 5:    # 'if', 'x', '>', '5', ':' are tokens
    print("Hello")  # 'print', '("Hello")', are tokens
```

3. Variables:

In a program, a variable is a named storage location containing a value.

The stored value of the variable is referenced and manipulated using the variable name itself. Through programming, variables enable data storage, retrieval and modification during

program execution.

A variable's data type defines its ability to hold different types of data.

Variables example in Java language:

`Java`

```java
int count = 5;        // 'count' is a variable of type 'int'
double price = 2.99;  // 'price' is a variable of type 'double'
String name = "John"; // 'name' is a variable of type 'String'
```

4. Operators:

A collection of keywords or symbols that can be used to operate on an operand is referred to as an operator in programming.

In programming, operands are represented by variables, expressions and literals.

They decide how they should be consolidated or modified so that a particular outcome is given.

Some operators frequently employed in programming include the arithmetic ones such as plus (+), minus (-), multiplication (*) and division (/).

Other examples are comparison (==, !=, <, >), logical (&&, ||) and assignment (=) operators.

Operators example in Python language:

Python

```
# Arithmetic operators
x = 5 + 3    # Addition
y = 7 - 2    # Subtraction
z = 4 * 6    # Multiplication
w = 8 / 2    # Division

# Comparison operators
a = (x == y)    # Equal to
b = (z > w)     # Greater than

# Logical operators
c = (a and b)   # Logical AND
d = (a or b)    # Logical OR
```

Related posts:

1. RGPV BCE PYQs
2. Describe Remote sensing
3. Explain I/O devices in detail with suitable examples
4. Explain memory and type of memory in detail.
5. Define algorithms. What is the need of algorithms ? Describe three benefits of algorithms.
6. Explain procedure-oriented programming with examples.
7. Explain data structures in detail.

8. Define objects and classes. can a class in C++ have more than one constructor with the same name? Justify your answer with suitable example.