

Conclusion of Lecture 2: Java Basics 1

Key Points Covered:

1. Structure of a Java Program

- Every Java program must have at least one class.
- The class containing the main method is the entry point.
- The program file must have a .java extension.

2. Main Method

- Execution begins from the main method.
- Syntax: `public static void main(String[] args)`

3. Output Statements

- `System.out.println()`: prints and moves to the next line.
- `System.out.print()`: prints and stays on the same line.
- Escape sequences: `\n`, `\t`, `\"`, `\\`, etc.

4. Identifiers

- Used to name variables, methods, classes, etc.
- Rules: start with letter, `_`, or `$`; can include digits; no spaces; case-sensitive.

5. Variables and Data Types

- Primitive types: `int`, `double`, `char`, `boolean`, etc.
- Reference type: `String`
- Declaration: `type variableName;`
- Initialization: `variableName = value;` or `type variableName = value;`

6. Constants

- Declared with `final` keyword.

- **Example: final double PI = 3.14159;**

7. String Concatenation

- **Using + operator to combine strings and other data types.**

8. Type Casting

- **Implicit: automatic (e.g., int to double).**
- **Explicit: manual (e.g., double to int – truncates decimal part).**

9. Formatted Output with printf

- **Uses format specifiers: %d, %f, %s, etc.**
- **Example: System.out.printf("Count: %d, Amount: %.2f", count, amount);**

10. Comments and Whitespace

- **Single-line: //**
- **Multi-line: /* ... */**
- **Whitespace improves readability but doesn't affect execution.**

11. Common Mistakes

- **Type mismatch, undeclared variables, misspelling, re-declaration, uninitialized use.**

? Questions and Problems with Answers

1. Basic Syntax and Structure

Q1: What is the output of the following program?

```
public class Test {  
  
    public static void main(String[] args) {
```

```
System.out.print("Hello");  
  
System.out.println("World");  
  
}  
  
}
```

A1:
HelloWorld (on one line)

Q2: Identify the error in the code:

```
public class Test {  
  
    public static void main(String[] args) {  
  
        System.out.println("Hello World")  
  
    }  
  
}
```

A2:
Missing semicolon at the end of the println statement.

2. Variables and Data Types

Q3: Declare and initialize a variable of type int with value 10.

A3:

```
int num = 10;
```

Q4: Which of the following are valid identifiers?

- a) 2var**
- b) _name**
- c) class**
- d) \$salary**

A4:

Valid: b, d

Invalid: a (starts with digit), c (reserved word)

Q5: What is the output?

```
int a = 5;
```

```
double b = a;
```

```
System.out.println(b);
```

A5:

5.0 (implicit casting from int to double)

3. Constants

Q6: Declare a constant named MAX_VALUE of type int with value 100.

A6:

```
final int MAX_VALUE = 100;
```

4. Escape Sequences

Q7: Write a print statement that outputs:

text

Name: "Alice"

Age: 20

A7:

```
System.out.println("Name: \"Alice\"\nAge: 20");
```

5. Type Casting

Q8: What is the value of result?

```
double x = 7.8;
```

```
int y = (int) x;
```

A8:

7 (decimal truncated)

Q9: What is wrong with this code?

```
int z = 9 / 2.0;
```

A9:
Type mismatch: 9 / 2.0 is a double, cannot be assigned to an int without casting.

6. String Concatenation

Q10: What is the output?

```
System.out.println("Result: " + 4 + 5);
```

A10:
Result: 45 (string concatenation, not addition)

Q11: How to get output Result: 9?

```
System.out.println("Result: " + (4 + 5));
```

7. printf Formatting

Q12: Write a printf statement to output:

text

Price: \$19.99

Quantity: 5

A12:

```
System.out.printf("Price: $%.2f\nQuantity: %d", 19.99, 5);
```

8. Common Errors

Q13: Find the error:

```
int number;
```

```
System.out.println(number);
```

A13:

Variable number is not initialized.

Q14: Find the error:

```
double value = 10.5;
```

```
int num = value;
```

A14:

Cannot implicitly convert double to int. Requires explicit casting:

```
int num = (int) value;
```

9. Trace the Output

Q15: What is the output?

```
public class Trace {  
  
    public static void main(String[] args) {  
  
        int a = 2;  
  
        int b = a * a * a;  
  
        System.out.println("Cube: " + b);  
  
    }  
  
}
```

A15:
Cube: 8

10. Advanced printf

Q16: What does this print?

```
System.out.printf("%10s %5.2f", "Total:", 123.4567);
```

A16:
Total: 123.46
(10-width string, 5-width float with 2 decimal places)

11. Boolean and Char

Q17: Declare a char variable that holds the letter 'A'.

A17:

```
char letter = 'A';
```

Q18: Declare a boolean variable named isValid and set it to true.

A18:

```
boolean isValid = true;
```

12. Mixed Operations

Q19: What is the output?

```
System.out.println(5 + 3 + "Java" + 2 + 1);
```

A19:

8Java21

13. Memory Tracing

Q20: Trace the values of x and y:

```
int x = 5;
```

```
int y = x++;
```

```
System.out.println("x: " + x + ", y: " + y);
```

A20:

x: 6, y: 5

14. Constant Usage

Q21: Why use constants? Give an example.

A21:

Constants make code more readable and prevent accidental changes.

Example:

```
final double TAX_RATE = 0.15;
```

15. Invalid Declarations

Q22: Which declaration is invalid?

- a) int a;**
- b) double 2b;**
- c) String name;**
- d) char c;**

A22:

b) double 2b; – identifier cannot start with a digit